

NIM : 2141762044

### Pertemuan 3

```

    erDiagram
        m_kategori ||--o{ m_barang : "has"
        t_stok ||--o{ m_barang : "has"
        t_perjualan_detail ||--o{ m_barang : "has"
        m_level ||--o{ m_user : "has"
        t_perjualan ||--o{ m_user : "has"

        m_kategori {
            int kategori_id PK
            string(10) kategori_kode
            string(100) kategori_nama
        }

        t_stok {
            int stok_id PK
            int barang_id FK
            int user_id FK
            datetime stok_tanggal
            int stok_jumlah
        }

        m_barang {
            int barang_id PK
            int kategori_id FK
            string(10) barang_kode
            string(100) barang_nama
            int harga_beli
            int harga_jual
        }

        t_perjualan_detail {
            int detail_id PK
            int barang_id FK
            int penjualan_id FK
            int harga
            int jumlah
        }

        m_level {
            int level_id PK
            string(10) level_kode
            string(100) level_nama
        }

        m_user {
            int user_id PK
            int level_id FK
            string(20) username
            string(100) nama
            string(255) password
        }

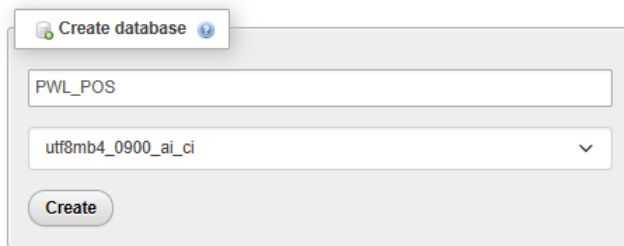
        t_perjualan {
            int penjualan_id PK
            int user_id FK
            string(50) pembeli
            string(20) penjualan_kode
            datetime penjualan_tanggal
        }
  
```

Studi kasus ini bertujuan untuk memberikan gambaran menyeluruh tentang proses pengembangan aplikasi PoS sederhana, mulai dari perencanaan hingga implementasi. Setiap langkahnya akan dijabarkan dengan rinci untuk memastikan keberhasilan aplikasi PoS yang sesuai dengan kebutuhan pengguna dan lingkungan bisnis yang diinginkan.

## A. Praktikum 1 – Pengaturan Database

### 1. Membuat database PWL\_POS

#### Databases

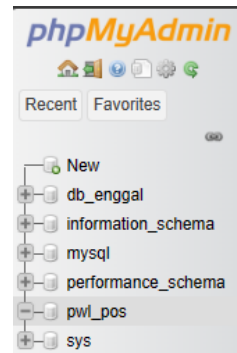


Create database

PWL\_POS

utf8mb4\_0900\_ai\_ci

Create



### 2. APP\_KEY file .env

```
.env
APP_NAME=Laravel
APP_ENV=local
APP_KEY=base64:sjhdI7NLSZgOp52j9bkgKXfNEtiFBcdsWEK/RE0HF2g=
APP_DEBUG=true
APP_URL=http://localhost

LOG_CHANNEL=stack
LOG_DEPRECATIONS_CHANNEL=null
LOG_LEVEL=debug
```

### 3. Menyesuaikan nama database di file .env

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=pwl_pos
DB_USERNAME=root
DB_PASSWORD=

BROADCAST_DRIVER=log
CACHE_DRIVER=file
FILESYSTEM_DISK=local
QUEUE_CONNECTION=sync
SESSION_DRIVER=file
SESSION_LIFETIME=120

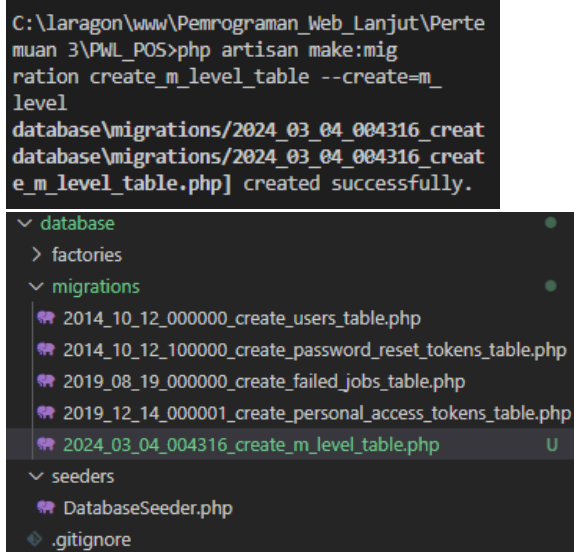
MEMCACHED_HOST=127.0.0.1
```

## B. Praktikum 2.1 – Pembuatan File Migrasi Tanpa Relasi

### Tabel m\_level

1. Membuat file migrasi m\_level

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:migration create_m_level_table --create=m_level
database\migrations/2024_03_04_004316_create_m_level_table.php created successfully.
```



2. Memodifikasi schema tabel m\_level

```
public function up(): void
{
    Schema::create('m_level', function (Blueprint $table) {
        $table->id('level_id');
        $table->string('level_kode', 10)->unique();
        $table->string('level_nama', 100);
        $table->timestamps();
    });
}
```

3. Melakukan migrasi tabel m\_level

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan migrate

INFO Preparing database.

Creating migration table ..... 160ms DONE

INFO Running migrations.

2014_10_12_000000_create_users_table ..... 81ms DONE
2014_10_12_100000_create_password_reset_tokens_table ..... 16ms DONE
2019_08_19_000000_create_failed_jobs_table ..... 42ms DONE
2019_12_14_000001_create_personal_access_tokens_table ..... 75ms DONE

C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan migrate

INFO Running migrations.

2024_03_04_004316_create_m_level_table ..... 47ms DONE
```

4. Hasil

Table	Action
<input type="checkbox"/> failed_jobs	★ Browse Structure Search Ins
<input type="checkbox"/> migrations	★ Browse Structure Search Ins
<input type="checkbox"/> m_level	★ Browse Structure Search Ins
<input type="checkbox"/> password_reset_tokens	★ Browse Structure Search Ins
<input type="checkbox"/> personal_access_tokens	★ Browse Structure Search Ins
<input type="checkbox"/> users	★ Browse Structure Search Ins
6 tables	Sum

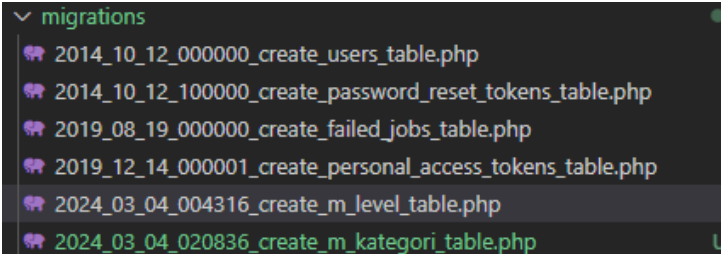
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/> 1	level_id	bigint		UNSIGNED	No	None		AUTO_I
<input type="checkbox"/> 2	level_kode	varchar(10)	utf8mb4_unicode_ci		No	None		
<input type="checkbox"/> 3	level_nama	varchar(100)	utf8mb4_unicode_ci		No	None		
<input type="checkbox"/> 4	created_at	timestamp			Yes	NULL		
<input type="checkbox"/> 5	updated_at	timestamp			Yes	NULL		

## Tabel m\_kategori

### 1. Membuat file migrasi m\_kategori

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan_3\PWL_POS>php artisan make:migration create_m_kategori_table --create=m_kategori
```

INFO Migration [C:\laragon\www\Pemrograman\_Web\_Lanjut\Pertemuan\_3\PWL\_POS\database\migrations\2024\_03\_04\_020836\_create\_m\_kategori\_table.php] created successfully.



### 2. Memodifikasi schema m\_kategori

```
public function up(): void
{
    Schema::create('m_kategori', function (Blueprint $table) {
        $table->id('kategori_id');
        $table->string('kategori_kode', 10)->unique();
        $table->string('kategori_nama', 100);
        $table->timestamps();
    });
}
```

### 3. Melakukan migrasi m\_kategori

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan_3\PWL_POS>php artisan migrate
```

INFO Running migrations.

2024\_03\_04\_020836\_create\_m\_kategori\_table ..... 154ms DONE

### 4. Hasil

The screenshot shows the Laravel Admin interface. At the top, there's a list of database tables: failed\_jobs, migrations, m\_kategori, m\_level, password\_reset\_tokens, personal\_access\_tokens, and users. Each table has icons for Browse, Structure, Search, and other actions. Below this, the 'Table structure' view for the 'm\_kategori' table is displayed. It shows a table with 5 columns: kategori\_id (bigint, UNSIGNED, AUTO\_INCREMENT), kategori\_kode (varchar(10), utf8mb4\_unicode\_ci, UNIQUE), kategori\_nama (varchar(100), utf8mb4\_unicode\_ci), created\_at (timestamp), and updated\_at (timestamp).

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	kategori_id	bigint		UNSIGNED	No	None		AUTO_INCREMENT
2	kategori_kode	varchar(10)	utf8mb4_unicode_ci		No	None		UNIQUE
3	kategori_nama	varchar(100)	utf8mb4_unicode_ci		No	None		
4	created_at	timestamp			Yes	NULL		
5	updated_at	timestamp			Yes	NULL		

## C. Praktikum 2.2 – Pembuatan File Migrasi Dengan Relasi

### Tabel m\_user

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:migration create_m_user_table --create=m_user
```

```
INFO Migration [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\database\migrations\2024_03_04_021545_create_m_user_table.php] created successfully.
```

```
migrations
2014_10_12_000000_create_users_table.php
2014_10_12_100000_create_password_reset_tokens_table.p
2019_08_19_000000_create_failed_jobs_table.php
2019_12_14_000001_create_personal_access_tokens_table.p
2024_03_04_004316_create_m_level_table.php
2024_03_04_020836_create_m_kategori_table.php
2024_03_04_021545_create_m_user_table.php U
```

```
public function up(): void
{
    Schema::create('m_user', function (Blueprint $table) {
        $table->id('user_id');
        $table->unsignedBigInteger('level_id')->index();//indexing untuk ForeignKey
        $table->string('username', 20)->unique();//unique untuk memastikan tidak ada username yang sama
        $table->string('nama', 100);
        $table->string('password');
        $table->timestamps();

        //mendefinisikan Foreign Key pada kolom level_id mengacu pada kolom level_id di tabel m_level
        $table->foreign('level_id')->references('level_id')->on('m_level');
    });
}
```

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan migrate
```

```
INFO Running migrations.
```

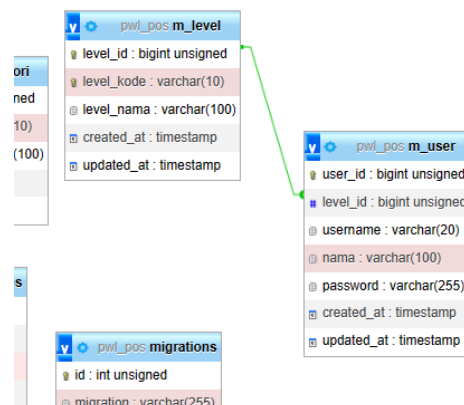
```
2024_03_04_021545_create_m_user_table ..... 197ms DONE
```

Hasil:

Table	Action
<input type="checkbox"/> failed_jobs	★ Browse Structure Search In
<input type="checkbox"/> migrations	★ Browse Structure Search In
<input type="checkbox"/> m_kategori	★ Browse Structure Search In
<input type="checkbox"/> m_level	★ Browse Structure Search In
<input type="checkbox"/> m_user	★ Browse Structure Search In
<input type="checkbox"/> password_reset_tokens	★ Browse Structure Search In
<input type="checkbox"/> personal_access_tokens	★ Browse Structure Search In
<input type="checkbox"/> users	★ Browse Structure Search In

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	user_id	bigint		UNSIGNED	No	None		AUTO_INCREMENT
2	level_id	bigint		UNSIGNED	No	None		
3	username	varchar(20)	utf8mb4_unicode_ci		No	None		
4	nama	varchar(100)	utf8mb4_unicode_ci		No	None		
5	password	varchar(255)	utf8mb4_unicode_ci		No	None		
6	created_at	timestamp			Yes	NULL		
7	updated_at	timestamp			Yes	NULL		

Tabel m\_level dan m\_user berelasi dimana level\_id pada m\_user merupakan Foreign Key



## Tabel m\_barang

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:migration create_m_barang_table --create=m_barang

INFO Migration [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\database\migrations\2024_03_04_024814_create_m_b
arang_table.php] created successfully.

2024_03_04_024814_create_m_barang_table.php

public function up(): void
{
    Schema::create('m_barang', function (Blueprint $table) {
        $table->id('barang_id');
        $table->unsignedBigInteger('kategori_id')->index();
        $table->string('barang_kode', 10)->unique();
        $table->string('barang_nama', 100);
        $table->integer('harga_beli');
        $table->integer('harga_jual');
        $table->timestamps();

        $table->foreign('kategori_id')->references('kategori_id')->on('m_kategori');
    });
}

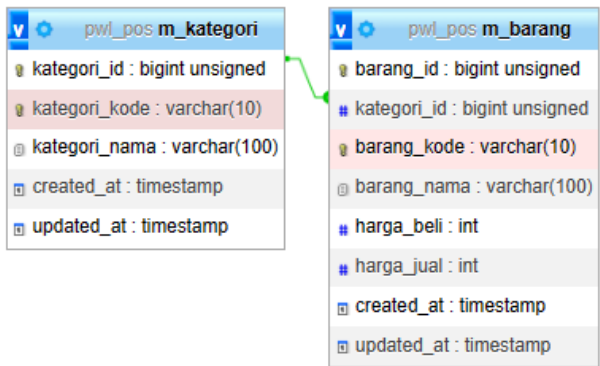
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan migrate

INFO Running migrations.

2024_03_04_024814_create_m_barang_table ..... 226ms DONE
```

Hasil:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	barang_id	bigint		UNSIGNED	No	None		AUTO_INCREMENT
2	kategori_id	bigint		UNSIGNED	No	None		
3	barang_kode	varchar(10)	utf8mb4_unicode_ci		No	None		
4	barang_nama	varchar(100)	utf8mb4_unicode_ci		No	None		
5	harga_beli	int			No	None		
6	harga_jual	int			No	None		
7	created_at	timestamp			Yes	NULL		
8	updated_at	timestamp			Yes	NULL		



## Tabel t\_penjualan

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:migration create_t_penjualan_table --create=t_penjualan
```

```
INFO Migration [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\database\migrations\2024_03_04_030654_create_t_penjualan_table.php] created successfully.
```

2024\_03\_04\_030654\_create\_t\_penjualan\_table... U

```
public function up(): void
{
    Schema::create('t_penjualan', function (Blueprint $table) {
        $table->id('penjualan_id');
        $table->unsignedBigInteger('user_id')->index();
        $table->string('pembeli', 50);
        $table->string('penjualan_kode', 20)->unique();
        $table->datetime('penjualan_tanggal');
        $table->timestamps();

        $table->foreign('user_id')->references('user_id')->on('m_user');
    });
}
```

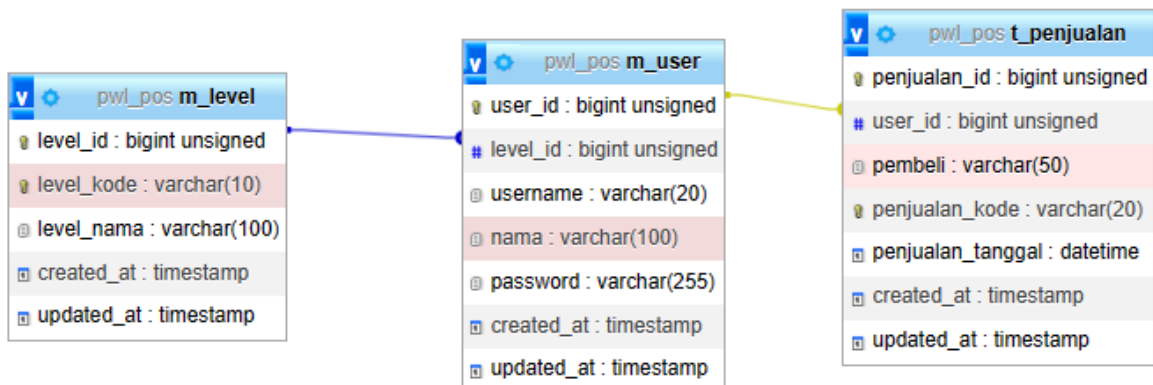
```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan migrate
```

```
INFO Running migrations.
```

```
2024_03_04_030654_create_t_penjualan_table ..... 195ms DONE
```

Hasil:

Server: localhost3306 > Database: pwl_pos > Table: t_penjualan									
Browse Structure SQL Search Insert Export Import Privileges									
Table structure Relation view									
	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/>	1	penjualan_id	bigint		UNSIGNED	No	None		AUTO_INCREMENT
<input type="checkbox"/>	2	user_id	bigint		UNSIGNED	No	None		
<input type="checkbox"/>	3	pembeli	varchar(50)	utf8mb4_unicode_ci		No	None		
<input type="checkbox"/>	4	penjualan_kode	varchar(20)	utf8mb4_unicode_ci		No	None		
<input type="checkbox"/>	5	penjualan_tanggal	datetime			No	None		
<input type="checkbox"/>	6	created_at	timestamp			Yes	NULL		
<input type="checkbox"/>	7	updated_at	timestamp			Yes	NULL		



## Tabel t\_stok

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:migration create_t_stok_table --create=t_stok
```

```
INFO Migration [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\database\migrations\2024_03_04_032149_create_t_stok_table.php] created successfully.
```

```
2024_03_04_032149_create_t_stok_table.php U
```

```
public function up(): void
{
    Schema::create('t_stok', function (Blueprint $table) {
        $table->id('stok_id');
        $table->unsignedBigInteger('barang_id')->index();
        $table->unsignedBigInteger('user_id')->index();
        $table->datetime('stok_tanggal');
        $table->integer('stok_jumlah');
        $table->timestamps();

        $table->foreign('barang_id')->references('barang_id')->on('m_barang');
        $table->foreign('user_id')->references('user_id')->on('m_user');
    });
}
```

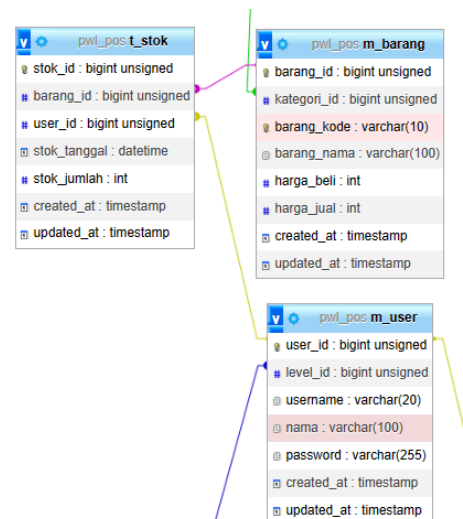
```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan migrate
```

```
INFO Running migrations.
```

```
2024_03_04_032149_create_t_stok_table ..... 213ms DONE
```

## Hasil:

Server: localhost:3306 » Database: pwf_pos » Table: t_stok								
Browse Structure SQL Search Insert Export Import								
Table structure Relation view								
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/>	1 stok_id	bigint		UNSIGNED	No	None		AUTO_INCREMENT
<input type="checkbox"/>	2 barang_id	bigint		UNSIGNED	No	None		INDEX
<input type="checkbox"/>	3 user_id	bigint		UNSIGNED	No	None		INDEX
<input type="checkbox"/>	4 stok_tanggal	datetime			No	None		
<input type="checkbox"/>	5 stok_jumlah	int			No	None		
<input type="checkbox"/>	6 created_at	timestamp			Yes	NULL		
<input type="checkbox"/>	7 updated_at	timestamp			Yes	NULL		





## Tabel t\_penjualan\_detail

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan_3\PWL_POS>php artisan make:migration create_t_penjualan_detail_table --create=t_penjualan_detail
```

```
INFO Migration [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan_3\PWL_POS\database\migrations\2024_03_04_033127_create_t_penjualan_detail_table.php] created successfully.
```

```
2024_03_04_033127_create_t_penjualan_det... U
```

```
public function up(): void
{
    Schema::create('t_penjualan_detail', function (Blueprint $table) {
        $table->id('detail_id');
        $table->unsignedBigInteger('penjualan_id')->index();
        $table->unsignedBigInteger('barang_id')->index();
        $table->integer('harga');
        $table->integer('jumlah');
        $table->timestamps();

        $table->foreign('penjualan_id')->references('penjualan_id')->on('t_penjualan');
        $table->foreign('barang_id')->references('barang_id')->on('m_barang');
    });
}
```

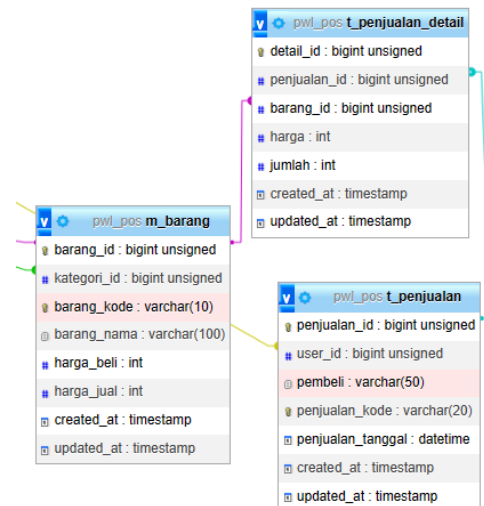
```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan_3\PWL_POS>php artisan migrate
```

```
INFO Running migrations.
```

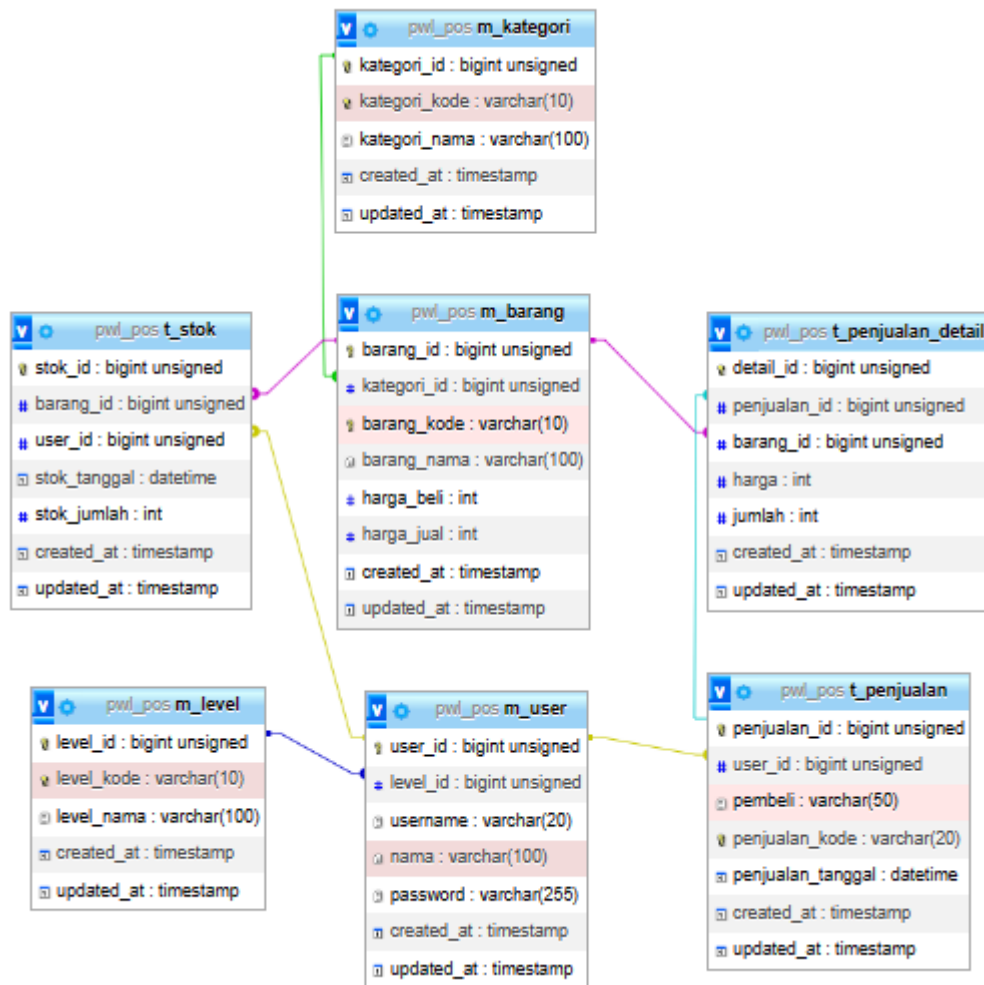
```
2024_03_04_033127_create_t_penjualan_detail_table ..... 162ms DONE
```

## Hasil:

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	detail_id	bigint		UNSIGNED	No	None		AUTO
2	penjualan_id	bigint		UNSIGNED	No	None		
3	barang_id	bigint		UNSIGNED	No	None		
4	harga	int			No	None		
5	jumlah	int			No	None		
6	created_at	timestamp			Yes	NULL		
7	updated_at	timestamp			Yes	NULL		



## Desain Database Diagram



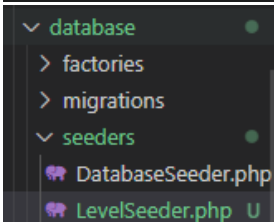
## D. Praktikum 3 – Membuat File Seeder

### Seeder m\_level

#### 1. Membuat file seeder LevelSeeder

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:seeder LevelSeeder
```

```
INFO Seeder [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\database\seeders\LevelSeeder.php] created successfully
```



#### 2. Modifikasi use Illuminate dan isi function run()

```
<?php

namespace Database\Seeders;

use Illuminate\Database\Console\Seeds\WithoutModelEvents;
use Illuminate\Database\Seeder;
use Illuminate\Support\Facades\DB;

class LevelSeeder extends Seeder
{
    /**
     * Run the database seeds.
     */
    public function run(): void
    {
        $data = [
            ['level_id' => 1, 'level_kode' => 'ADM', 'level_nama' => 'Administrator'],
            ['level_id' => 2, 'level_kode' => 'MNG', 'level_nama' => 'Manager'],
            ['level_id' => 3, 'level_kode' => 'STF', 'level_nama' => 'Staff/Kasir'],
        ];
        DB::table('m_level')->insert($data);
    }
}
```

#### 3. Menjalankan file seeder table m\_level

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan db:seed --class=LevelSeeder
```

```
INFO Seeding database.
```

#### 4. Hasil

			level_id	level_kode	level_nama	created_at	updated_at
<input type="checkbox"/>	Edit	Copy	Delete	1	ADM	Administrator	NULL
<input type="checkbox"/>	Edit	Copy	Delete	2	MNG	Manager	NULL
<input type="checkbox"/>	Edit	Copy	Delete	3	STF	Staff/Kasir	NULL

## Seeder m\_user me-refer ke m\_level

1. Membuat file seeder LevelSeeder

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:seeder UserSeeder  
INFO Seeder [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\database\seeders\UserSeeder.php] created successfully.
```

```
▼ seeders  
  DatabaseSeeder...  
  LevelSeeder.... U  
  UserSeeder.... U
```

2. Modifikasi use Illuminate dan isi function run()

```
<?php  
  
namespace Database\Seeders;  
  
use Illuminate\Database\Console\Seeds\WithoutModelEvents;  
use Illuminate\Database\Seeder;  
use Illuminate\Support\Facades\DB;  
use Illuminate\Support\Facades\Hash;  
  
public function run(): void  
{  
    $data = [  
        [  
            'user_id' => 1,  
            'level_id' => 1,  
            'username' => 'admin',  
            'nama' => 'Administrator',  
            'password' => Hash::make('12345'), //class untuk mengenkripsikan/hash password  
        ],  
        [  
            'user_id' => 2,  
            'level_id' => 2,  
            'username' => 'manager',  
            'nama' => 'Manager',  
            'password' => Hash::make('12345'),  
        ],  
        [  
            'user_id' => 3,  
            'level_id' => 3,  
            'username' => 'staff',  
            'nama' => 'Staff/Kasir',  
            'password' => Hash::make('12345'),  
        ],  
    ];  
    DB::table('m_user')->insert($data);  
}
```

3. Menjalankan file seeder table m\_level

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan db:seed --class=UserSeeder  
INFO Seeding database.
```

4. Hasil

user_id	level_id	username	nama	password	created_at	updated_at
1	1	admin	Administrator	\$2y\$12\$i1687B/eXZcfwp1CXGZuj00L.1SPzTMqQkLwBRNaNt...	NULL	NULL
2	2	manager	Manager	\$2y\$12\$aYfpZ0YppbElN6czcGmaMevLeBQDZHhuyI6HD2ezeiK...	NULL	NULL
3	3	staff	Staff/Kasir	\$2y\$12\$k/8aM4dlb9d5/d2ZpvdvVOPvbs.npzRQBx6GNor9agS...	NULL	NULL

## Seeder m\_kategori

```
public function run(): void
{
    $data = [
        [
            'kategori_id' => 1,
            'kategori_kode' => 'FSH',
            'kategori_nama' => 'Fashion',
        ],
        [
            'kategori_id' => 2,
            'kategori_kode' => 'KCT',
            'kategori_nama' => 'Kecantikan',
        ],
        [
            'kategori_id' => 3,
            'kategori_kode' => 'ELK',
            'kategori_nama' => 'Elektronik',
        ],
    ],
```

kategori_id	kategori_kode	kategori_nama	created_at	updated_at
1	FSH	Fashion	NULL	NULL
2	KCT	Kecantikan	NULL	NULL
3	ELK	Elektronik	NULL	NULL
4	SBK	Sembako	NULL	NULL
5	PRT	Perlengkapan Rumah Tangga	NULL	NULL

## Seeder m\_barang

```
public function run(): void
{
    $data = [
        [
            'barang_id' => 1,
            'kategori_id' => 1,
            'barang_kode' => 'ATG',
            'barang_nama' => 'Anting',
            'harga_beli' => 18000,
            'harga_jual' => 20000,
        ],
        [
            'barang_id' => 2,
            'kategori_id' => 1,
            'barang_kode' => 'JTG',
            'barang_nama' => 'Jam Tangan',
            'harga_beli' => 85000,
            'harga_jual' => 100000,
        ],
    ],
```

barang_id	kategori_id	barang_kode	barang_nama	harga_beli	harga_jual	created_at	updated_at
1	1	ATG	Anting	18000	20000	NULL	NULL
2	1	JTG	Jam Tangan	85000	100000	NULL	NULL
3	2	DDR	Deodoran	15000	17000	NULL	NULL
4	2	SBW	Sabun Wajah	26000	29000	NULL	NULL
5	3	KMR	Kamera	3850000	4000000	NULL	NULL
6	3	PBK	Powerbank	75000	80000	NULL	NULL
7	4	MYK	Minyak	11000	12000	NULL	NULL
8	4	BRS	Beras	140000	150000	NULL	NULL
9	5	DTR	Detergen	9000	10000	NULL	NULL
10	5	TSU	Tisu	7000	8000	NULL	NULL

## Seeder t\_stok

```
public function run(): void
{
    $data = [
        [
            'stok_id' => 1,
            'barang_id' => 1,
            'user_id' => 1,
            'stok_tanggal' => now(), //'2017-04-27 10:29:59'
            'stok_jumlah' => 50,
        ],
        [
            'stok_id' => 2,
            'barang_id' => 2,
            'user_id' => 3,
            'stok_tanggal' => now(),
            'stok_jumlah' => 15,
        ],
    ]
}
```

stok_id	barang_id	user_id	stok_tanggal	stok_jumlah	created_at	updated_at
1	1	1	2024-03-04 11:46:38	40	NULL	NULL
2	2	3	2024-03-04 11:58:04	15	NULL	NULL
3	3	3	2024-03-04 11:58:04	50	NULL	NULL
4	4	1	2024-03-04 11:58:04	50	NULL	NULL
5	5	2	2024-03-04 11:58:04	8	NULL	NULL
6	6	3	2024-03-04 11:58:04	12	NULL	NULL
7	7	1	2024-03-04 11:58:04	40	NULL	NULL
8	8	2	2024-03-04 11:58:04	20	NULL	NULL
9	9	3	2024-03-04 11:58:04	40	NULL	NULL
10	10	3	2024-03-04 11:58:04	50	NULL	NULL

### Seeder t\_penjualan

```
$data = [
    [
        'penjualan_id' => 1,
        'user_id' => 1,
        'pembeli' => 'Rihana Somy',
        'penjualan_kode' => 'P01',
        'penjualan_tanggal' => now(),
    ],
    [
        'penjualan_id' => 2,
        'user_id' => 3,
        'pembeli' => 'Panji Takir',
        'penjualan_kode' => 'P02',
        'penjualan_tanggal' => now(),
    ],
    [
        'penjualan_id' => 3,
        'user_id' => 3,
        'pembeli' => 'Eggy Rawitan',
        'penjualan_kode' => 'P03',
        'penjualan_tanggal' => now(),
    ],
    [
        'penjualan_id' => 4,
        'user_id' => 3,
        'pembeli' => 'Vania',
        'penjualan_kode' => 'P04',
        'penjualan_tanggal' => now(),
    ],
    [
        'penjualan_id' => 5,
        'user_id' => 1,
        'pembeli' => 'Lena Riswa',
        'penjualan_kode' => 'P05',
        'penjualan_tanggal' => now(),
    ],
    [
        'penjualan_id' => 6,
        'user_id' => 2,
        'pembeli' => 'Rismawati Marine',
        'penjualan_kode' => 'P06',
        'penjualan_tanggal' => now(),
    ],
    [
        'penjualan_id' => 7,
        'user_id' => 3,
        'pembeli' => 'Menik Ramin',
        'penjualan_kode' => 'P07',
        'penjualan_tanggal' => now(),
    ],
    [
        'penjualan_id' => 8,
        'user_id' => 3,
        'pembeli' => 'Sasa Takuri',
        'penjualan_kode' => 'P08',
        'penjualan_tanggal' => now(),
    ],
    [
        'penjualan_id' => 9,
        'user_id' => 1,
        'pembeli' => 'Wariam Meri',
        'penjualan_kode' => 'P09',
        'penjualan_tanggal' => now(),
    ],
    [
        'penjualan_id' => 10,
        'user_id' => 3,
        'pembeli' => 'Kolin Fawer',
        'penjualan_kode' => 'P10',
        'penjualan_tanggal' => now(),
    ],
]
```

penjualan_id	user_id	pembeli	penjualan_kode	penjualan_tanggal	created_at	updated_at
1	1	Rihana Somy	P01	2024-03-04 12:16:21	NULL	NULL
2	3	Panji Takir	P02	2024-03-04 12:16:21	NULL	NULL
3	3	Eggy Rawitan	P03	2024-03-04 12:16:21	NULL	NULL
4	3	Vania	P04	2024-03-04 12:16:21	NULL	NULL
5	1	Lena Riswa	P05	2024-03-04 12:16:21	NULL	NULL
6	2	Rismawati Marine	P06	2024-03-04 12:16:21	NULL	NULL
7	3	Menik Ramin	P07	2024-03-04 12:16:21	NULL	NULL
8	3	Sasa Takuri	P08	2024-03-04 12:16:21	NULL	NULL
9	1	Wariam Meri	P09	2024-03-04 12:16:21	NULL	NULL
10	3	Kolin Fawer	P10	2024-03-04 12:16:21	NULL	NULL

### Seeder t\_penjualan\_detail

```
$data = [
    ['detail_id' => 1, 'penjualan_id' => 1, 'barang_id' => 1, 'harga' => 20000, 'jumlah' => 2,],
    ['detail_id' => 2, 'penjualan_id' => 2, 'barang_id' => 2, 'harga' => 40000, 'jumlah' => 7,],
    ['detail_id' => 3, 'penjualan_id' => 3, 'barang_id' => 3, 'harga' => 30000, 'jumlah' => 3,],
    ['detail_id' => 4, 'penjualan_id' => 4, 'barang_id' => 4, 'harga' => 25000, 'jumlah' => 5,],
    ['detail_id' => 5, 'penjualan_id' => 5, 'barang_id' => 5, 'harga' => 20000, 'jumlah' => 3,],
    ['detail_id' => 6, 'penjualan_id' => 6, 'barang_id' => 6, 'harga' => 50000, 'jumlah' => 3,],
    ['detail_id' => 7, 'penjualan_id' => 7, 'barang_id' => 7, 'harga' => 37000, 'jumlah' => 2,],
    ['detail_id' => 8, 'penjualan_id' => 8, 'barang_id' => 8, 'harga' => 35000, 'jumlah' => 1,],
    ['detail_id' => 9, 'penjualan_id' => 9, 'barang_id' => 9, 'harga' => 31000, 'jumlah' => 7,],
    ['detail_id' => 10, 'penjualan_id' => 10, 'barang_id' => 10, 'harga' => 30000, 'jumlah' => 1,],
]
```

detail_id	penjualan_id	barang_id	harga	jumlah	created_at	updated_at
1	1	1	20000	2	NULL	NULL
2	2	2	40000	7	NULL	NULL
3	3	3	30000	3	NULL	NULL
4	4	4	25000	5	NULL	NULL
5	5	5	20000	3	NULL	NULL
6	6	6	50000	3	NULL	NULL
7	7	7	37000	2	NULL	NULL
8	8	8	35000	1	NULL	NULL
9	9	9	31000	7	NULL	NULL
10	10	10	30000	1	NULL	NULL

## E. Praktikum 4 – Implementasi DB Facade

### 1. Insert

Membuat file LevelController

```
INFO Controller [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan_3\PMW_POS\app\Http\Controllers\LevelController.php] created successfully
```

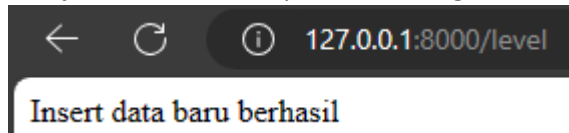
Menambahkan route

```
Route::get('/level', [LevelController::class, 'index']);
```

Memodifikasi LevelController

```
class LevelController extends Controller
{
    public function index(){
        DB::insert(
            'insert into m_level(level_kode, level_nama, created_at) values(?, ?, ?)',
            ['CUS', 'Pelanggan', now()]);
        return 'Insert data baru berhasil';
    }
}
```

Menjalankan localhost pada web dengan url /level



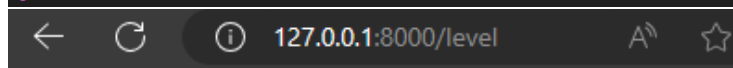
Terdapat penambahan data baru didatabase

level_id	level_kode	level_nama	created_at	updated_at
1	ADM	Administrator	NULL	NULL
2	MNG	Manager	NULL	NULL
3	STF	Staff/Kasir	NULL	NULL
4	CUS	Pelanggan	2024-03-04 14:17:12	NULL

### 2. Update

```
public function index(){
    // DB::insert(
    //     'INSERT into m_level(level_kode, level_nama, created_at) values(?, ?, ?)',
    //     ['CUS', 'Pelanggan', now()]);
    // return 'Insert data baru berhasil';

    $row = DB::update(
        'update m_level SET level_nama = ? where level_kode = ?',
        ['Customer', 'CUS']);
    return 'Update data berhasil. Jumlah data yang diupdate: ' . $row . ' baris';
}
```



level_id	level_kode	level_nama	created_at	updated_at
1	ADM	Administrator	NULL	NULL
2	MNG	Manager	NULL	NULL
3	STF	Staff/Kasir	NULL	NULL
4	CUS	Customer	2024-03-04 14:17:12	NULL



### 3. Delete

```
public function index(){
    // DB::insert(
    //     'insert into m_level(level_kode, level_nama, created_at) values(?, ?, ?)'
    //     ['CUS', 'Pelanggan', now()]);
    // return 'Insert data baru berhasil';

    // $row = DB::update(
    //     'update m_level SET level_nama = ? where level_kode = ?',
    //     ['Customer', 'CUS']);
    // return 'Update data berhasil. Jumlah data yang diupdate: ' . $row . ' baris';

    $row = DB::delete(
        'DELETE from m_level WHERE level_kode = ?',
        ['CUS']);
    return 'Delete data berhasil. Jumlah data yang dihapus: ' . $row . ' baris';
}
```

← ↻ ⓘ 127.0.0.1:8000/level A

Delete data berhasil. Jumlah data yang dihapus: 1 baris

level_id	level_kode	level_nama	created_at	updated_at
1	ADM	Administrator	NULL	NULL
2	MNG	Manager	NULL	NULL
3	STF	Staff/Kasir	NULL	NULL

### 4. Select

```
public function index(){
    // DB::insert(
    //     'insert into m_level(level_kode, level_nama, created_at) values(?, ?, ?)'
    //     ['CUS', 'Pelanggan', now()]);
    // return 'Insert data baru berhasil';

    // $row = DB::update(
    //     'UPDATE m_level SET level_nama = ? WHERE level_kode = ?',
    //     ['Customer', 'CUS']);
    // return 'Update data berhasil. Jumlah data yang diupdate: ' . $row . ' baris';

    // $row = DB::delete(
    //     'DELETE from m_level WHERE level_kode = ?',
    //     ['CUS']);
    // return 'Delete data berhasil. Jumlah data yang dihapus: ' . $row . ' baris';

    $data = DB::select(
        'SELECT * FROM m_level');
    return view('level', ['data' => $data]);
}
```

```

<!DOCTYPE html>
<html>
  <head>
    <title>Data Level Pengguna</title>
  </head>
  <body>
    <h1>Data Level Pengguna</h1>
    <table border="1" cellpadding="2" cellspacing="0">
      <tr>
        <th>ID</th>
        <th>Kode Level</th>
        <th>Nama Level</th>
      </tr>
      @foreach ($data as $d)
      <tr>
        <td>{{ $d->level_id }}</td>
        <td>{{ $d->level_kode }}</td>
        <td>{{ $d->level_nama }}</td>
      </tr>
      @endforeach
    </table>
  </body>
</html>

```

← ↻ ⓘ 127.0.0.1:8000/level

## Data Level Pengguna

ID	Kode Level	Nama Level
1	ADM	Administrator
2	MNG	Manager
3	STF	Staff/Kasir

## F. Praktikum 5 – Implementasi Query Builder

### 1. Insert

Membuat KategoriController

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan_3\PWL_POS>php artisan make:controller KategoriController
```

INFO Controller [C:\laragon\www\Pemrograman\_Web\_Lanjut\Pertemuan\_3\PWL\_POS\app\Http\Controllers\KategoriController.php] created successfully.

```
Route::get('/kategori', [KategoriController::class, 'index']);
```

```
use Illuminate\Http\Request;
use Illuminate\Support\Facades\DB;

class KategoriController extends Controller
{
    public function index()
    {
        $data = [
            'kategori_kode' => 'SNK',
            'kategori_nama' => 'Snack/Makanan Ringan',
            'created_at' => now(),
        ];
        DB::table('m_kategori')->insert($data);
        return 'Insert data baru berhasil';
    }
}
```

127.0.0.1:8000/kategori

Insert data baru berhasil

kategori_id	kategori_kode	kategori_nama	created_at	updated_at
1	FSH	Fashion	NULL	NULL
2	KCT	Kecantikan	NULL	NULL
3	ELK	Elektronik	NULL	NULL
4	SBK	Sembako	NULL	NULL
5	PRT	Perlengkapan Rumah Tangga	NULL	NULL
6	SNK	Snack/Makanan Ringan	2024-03-04 15:15:18	NULL

### 2. Update

```
$row = DB::table('m_kategori')
->WHERE('kategori_kode', 'SNK')
->UPDATE(['kategori_nama' => 'Camilan']);
return 'Update data berhasil. Jumlah data yang diupdate: ' . $row->count();
```

127.0.0.1:8000/kategori

Update data berhasil. Jumlah data yang diupdate: 1 baris

kategori_id	kategori_kode	kategori_nama	created_at	updated_at
1	FSH	Fashion	NULL	NULL
2	KCT	Kecantikan	NULL	NULL
3	ELK	Elektronik	NULL	NULL
4	SBK	Sembako	NULL	NULL
5	PRT	Perlengkapan Rumah Tangga	NULL	NULL
6	SNK	Camilan	2024-03-04 15:15:18	NULL

### 3. Delete

```
$row = DB::table('m_kategori')
->WHERE('kategori_kode', 'SNK')
->DELETE();
return 'Delete data berhasil. Jumlah data yang dihapus: ' . $row->count();
```

127.0.0.1:8000/kategori

Delete data berhasil. Jumlah data yang dihapus: 1baris

kategori_id	kategori_kode	kategori_nama	created_at	updated_at
1	FSH	Fashion	NULL	NULL
2	KCT	Kecantikan	NULL	NULL
3	ELK	Elektronik	NULL	NULL
4	SBK	Sembako	NULL	NULL
5	PRT	Perlengkapan Rumah Tangga	NULL	NULL

### 4. Select

```
$data = DB::table('m_kategori')->get();
return view('kategori', ['data' => $data]);
```

```
<!DOCTYPE html>
<html>
<head>
<title>Data Kategori Barang</title>
</head>
<body>
<h1>Data Kategori Barang</h1>
<table border="1" cellpadding="2" cellspacing="0">
<tr>
<th>ID</th>
<th>Kode Kategori</th>
<th>Nama Kategori</th>
</tr>
@foreach ($data as $d)
<tr>
<td>{{ $d->kategori_id }}</td>
<td>{{ $d->kategori_kode }}</td>
<td>{{ $d->kategori_nama }}</td>
</tr>
@endforeach
</table>
</body>
</html>
```

127.0.0.1:8000/kategori

## Data Kategori Barang

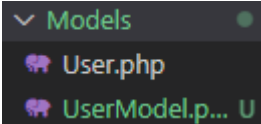
ID	Kode Kategori	Nama Kategori
1	FSH	Fashion
2	KCT	Kecantikan
3	ELK	Elektronik
4	SBK	Sembako
5	PRT	Perlengkapan Rumah Tangga

## G. Praktikum 6 – Implementasi Eloquent ORM

Membuat model UserModel

```
C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS>php artisan make:model UserModel
```

```
INFO Model [C:\laragon\www\Pemrograman_Web_Lanjut\Pertemuan 3\PWL_POS\app\Models\UserModel.php] created successfully.
```



Memodifikasi file UserModel

```
class UserModel extends Model
{
    use HasFactory;

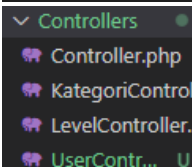
    protected $table = 'm_user';
    protected $primaryKey = 'user_id';
}
```

Memodifikasi route

```
Route::get('user', [UserController::class, 'index']);
```

Membuat file UserController

```
aman_Web_Lanjut\Pertemuan 3\PWL_POS\app\Http\Controllers\UserController.php] created successfully.
```



Memodifikasi controller

```
class UserController extends Controller
{
    public function index(){
        //coba akses model UserModel
        $user = UserModel::all(); //ambil semua data dari tabel m_user
        return view('user', ['data' => $user]);
    }
}
```

Membuat file view user

```
<html>
<head>
</head>
<body>
    <h1>Data User</h1>
    <table border="1" cellpadding="2" cellspacing="0">
        <tr>
            <th>ID</th>
            <th>Username</th>
            <th>Nama</th>
            <th>ID Level Pengguna</th>
        </tr>
        @foreach ($data as $d)
            <tr>
                <td>{{ $d->user_id }}</td>
                <td>{{ $d->username }}</td>
                <td>{{ $d->nama }}</td>
                <td>{{ $d->level_id }}</td>
            </tr>
        @endforeach
    </table>
</body>
</html>
```

Hasil

← ↻ ⓘ 127.0.0.1:8000/user A

## Data User

ID	Username	Nama	ID Level Pengguna
1	admin	Administrator	1
2	manager	Manager	2
3	staff	Staff/Kasir	3

Memodifikasi UserController untuk menambahkan data baru

```
public function index(){
    //tambah data user dengan Eloquent Model
    $data = [
        'username' => 'customer-1',
        'nama' => 'Pelanggan',
        'password' => Hash::make('12345'),
        'level_id' => 3
    ];
    UserModel::insert($data); //tambahkan data ke tabel m_user

    //coba akses model UserModel
    $user = UserModel::all(); //ambil semua data dari tabel m_user
    return view('user', ['data' => $user]);
}
```

← ↻ ⓘ 127.0.0.1:8000/user A

## Data User

ID	Username	Nama	ID Level Pengguna
1	admin	Administrator	1
2	manager	Manager	2
3	staff	Staff/Kasir	3
9	customer-1	Pelanggan	3

user_id	level_id	username	nama	password
1	1	admin	Administrator	\$2y\$12\$i1687B/e2
2	2	manager	Manager	\$2y\$12\$aYfpZ0Yp
3	3	staff	Staff/Kasir	\$2y\$12\$k/8aM4dl
9	3	customer-1	Pelanggan	\$2y\$12\$TisX50SY

## Memodifikasi kembali

```
public function index()
{
    // //tambah data user dengan Eloquent Model
    // $data = [
    //     'username' => 'customer-1',
    //     'nama' => 'Pelanggan',
    //     'password' => Hash::make('12345'),
    //     'level_id' => 3
    // ];
    // UserModel::insert($data); //tambahkan data ke tabel m_user

    //tambah data user dengan Eloquent Model
    $data = [
        'nama' => 'Pelanggan Pertama',
    ];
    UserModel::where('username', 'customer-1')->update($data); //update data user

    //coba akses model UserModel
    $user = UserModel::all(); //ambil semua data dari tabel m_user
    return view('user', ['data' => $user]);
}
```

127.0.0.1:8000/user

## Data User

ID	Username	Nama	ID Level Pengguna
1	admin	Administrator	1
2	manager	Manager	2
3	staff	Staff/Kasir	3
9	customer-1	Pelanggan Pertama	3

user_id	level_id	username	nama	password
1	1	admin	Administrator	\$2y\$12\$i1687B/e
2	2	manager	Manager	\$2y\$12\$aYfpZ0Yf
3	3	staff	Staff/Kasir	\$2y\$12\$k/8aM4dl
9	3	customer-1	Pelanggan Pertama	\$2y\$12\$TisX50Sy

## PENUTUP

1. Fungsi APP\_KEY merupakan kunci untuk mengenkripsi data sensitif, menghasilkan token autentikasi yang aman, dan memastikan integritas data aplikasi kecuali hashing/password. Penggunaan APP\_KEY dilakukan untuk keamanan data, karena jika APP\_KEY tidak dikonfigurasi maka semua sesi dan data terenkripsi tidak aman.
2. Untuk generate nilai APP\_KEY menggunakan perintah `php artisan key:generate` dimana nantinya akan menghasilkan string 32 karakter baru
3. Secara default Laravel memiliki 4 file migrasi, yaitu
  - Users: untuk menyimpan data pengguna
  - password\_reset\_tokens: untuk reset password
  - failed\_jobs: untuk menangani kesalahan ketika ada yang gagal dieksekusi
  - personal\_access\_tokens: digunakan untuk menyimpan token pengguna yang ingin mengakses API aplikasi
4. Nantinya file tersebut akan menjadi 2 kolom `created_at` dan `updated_at` di database, sehingga digunakan untuk mengetahui waktu atau tanggal insert/update data
5. Menghasilkan tipe data `bigint`
6. Penambahan string didalam kurung untuk menamai kolom tersebut, jika dibiarkan kosong maka nama kolom akan default

7. Unique digunakan agar tidak ada data sama yang masuk lagi
8. Pada tabel m\_user kolom level\_id merupakan Foreign Key, sedangkan di tabel m\_level merupakan Primary Key
9. Hash digunakan untuk mengamankan password dimana password 12345 akan menjadi potongan data string dan integer dimana adalah kode alfanumerik untuk mewakili data
10. Merupakan posisi variabel pada query SQL dimana data yang akan dimasukkan dituliskan setelahnya
11. Digunakan untuk nama tabel dan primary key yang digunakan oleh model UserModel
12. Lebih mudah menggunakan Eloquent ORM karena menggunakan teknik konversi data ke dalam bentuk object dimana dalam satu model akan merepresentasikan satu tabel database.