

**KONFIGURASI BASIS DATA  
(LARAGON)**



**Disusun Oleh  
ABDULLAH AZZAM RABBANI  
10240038**

**PROGRAM STUDI REKAYASA PERANGKAT LUNAK  
UNIVERSITAS BINA SARANA INFORMATIKA  
MARGONDA  
2025/2026**



## 1. PENDAHULUAN

Pada dokumen ini menjelaskan langkah-langkah implementasi akses basis data menggunakan **Laravel 10** dan **MySQL**. Implementasi dilakukan dengan framework Laravel yang menyediakan sistem ORM (Eloquent) untuk akses database yang aman dan efisien.

## 2. LINGKUNGAN PENGEMBANGAN

### 2.1. Server yang digunakan

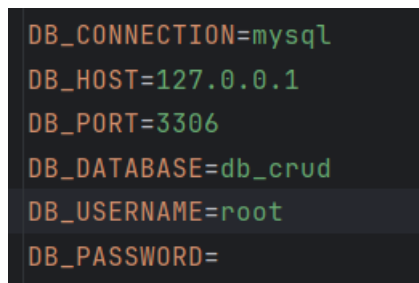
- Local Development Environment: Laragon
- Versi Laragon: 6.0.0
- Sistem Operasi : windows 11
- Web Server: Apache
- PHP Version : 8.1+
- Framework : Laravel 10
- Database Management System : MySQL 8.0

Langkah ini membuktikan bahwa proyek fungsional dan siap diakses, yang merupakan bagian dari Kemudahan interaksi

## 3. KONFIGURASI DATABASE

### 3.1. Persiapan Enviroment

Langkah pertama cari file yang bernama **.env**



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

Gambar 1: tampilah file .env

Dan sesuaikan dengan database kalian yang sudah kalian buat, pastikan username dan password juga sesuai.

### 3.2. Migrasi Database

Untuk membuat migration tabel anggota, jalankan perintah berikut di terminal Laragon:

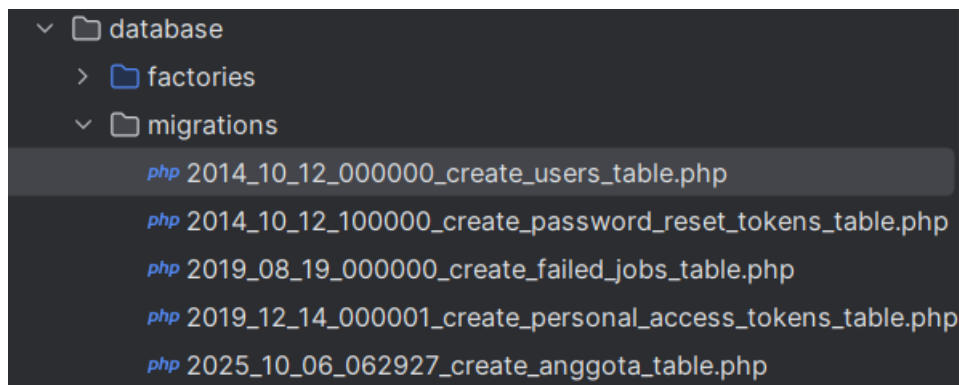
**php artisan make:migration create\_anggota\_table**

Atau dengan opsi yang lebih spesifik:

**php artisan make:migration create\_anggota\_table --create=anggota**

### 3.3. Membuat Field Pada Table Yang Sudah Di Buat

File pathnya ada di: database/migrations/[timestamp]\_create\_anggota\_table.php



Gambar 1.1: Tangkapam layer dari file migrasi database

kemudian untuk membuat field baru pada table anggota table yaitu seperti berikut

```
return new class extends Migration
{
    /**
     * Run the migrations.
     */
    no usages
    public function up(): void
    {
        Schema::create( table: 'users', function (Blueprint $table) {
            $table->id();
            $table->string( column: 'name');
            $table->string( column: 'email')->unique();
            $table->timestamp( column: 'email_verified_at')->nullable();
            $table->string( column: 'password');
            $table->rememberToken();
            $table->timestamps();
        });
    }
}
```

Gambar 1.2: Tangkapam layer dari file migrasi database

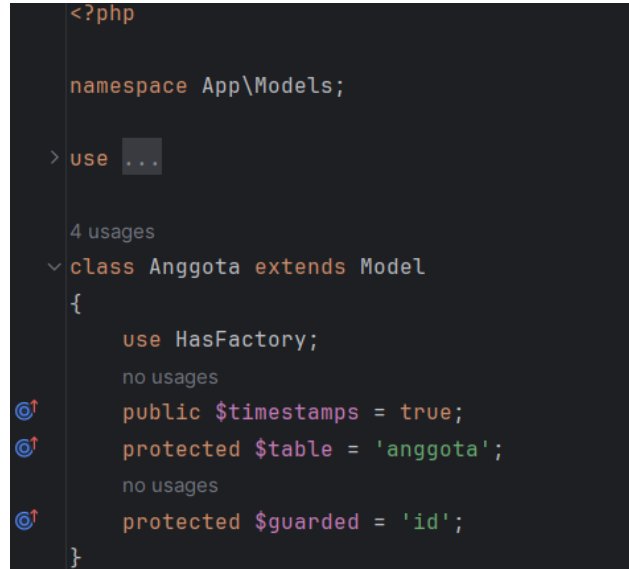
Perhatikan juga jenis tipe datanya pada masing – masing column

### 3.3. Membuat Model

Setelah membuat migration, buat juga modelnya dengan perintah berikut:

**php artisan make:model Anggota**

dan akses file nya pada folder: app/Models/Anggota.php



```
<?php

namespace App\Models;

> use ...

4 usages
class Anggota extends Model
{
    use HasFactory;
    no usages
    public $timestamps = true;
    protected $table = 'anggota';
    no usages
    protected $guarded = 'id';
}
```

Gambar 1.3: Tangkapan layar dari file model anggota

### 3.4. Menjalankan Migration

Setelah membuat file migration, jalankan perintah:

**php artisan migrate**

## 4. Verifikasi Database

Dan yang terakhir untuk memastikan table yang Anda sudah dibuat, Anda bisa check status migration dengan perintah : **php artisan migrate:status**



```
migrations:status
2014_10_12_000000_create_users_table ..... [1] Ran
2014_10_12_100000_create_password_reset_tokens_table ..... [1] Ran
2019_12_14_000001_create_failed_jobs_table ..... [1] Ran
2019_12_14_000001_create_personal_access_tokens_table ..... [1] Ran
2025_10_06_062927_create_anggota_table ..... [1] Ran
```

Gambar 1.4: Tangkapan layar dari status migrasi

Jika berhasil maka akan muncul tulisan **ran**.

Atau anda bisa check langsung di database server langsung dengan perintah:

USE db\_crud;

SHOW TABLES;

DESCRIBE anggota;

```
mysql> use db_crud;
Database changed
mysql> show tables;
+-----+
| Tables_in_db_crud |
+-----+
| anggota            |
| failed_jobs        |
| migrations          |
| password_reset_tokens |
| personal_access_tokens |
| users              |
+-----+
6 rows in set (0.01 sec)

mysql> describe anggota;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id         | bigint unsigned | NO   | PRI | NULL    | auto_increment |
| nama       | varchar(255)   | NO   |     | NULL    |                |
| hp         | varchar(13)    | NO   |     | NULL    |                |
| created_at | timestamp      | YES  |     | NULL    |                |
| updated_at | timestamp      | YES  |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.02 sec)
```

Jika berhasil  
Atau anda bisa  
USE db\_crud  
SHOW TABLES  
DESCRIBE

Gambar 1.4: Tangkapam layer darii mysql server