

Tugas Desain Database dengan Teknik ORM

Diajukan Untuk Memenuhi Tugas Mata Kuliah RPL: Desain dan Implementasi CII3J4

Dosen Pengampu: Ati Suci Dian Martha, S.Kom., M.T.



Disusun Oleh:

Ni Made Dwipadini Puspitarini	(1301194141)
Ahmad Julius Tarigan	(1301190345)
Afrizal Syahruluddin Yusuf	(1301194288)
Andi Muh. Syahrul Ulum	(1301190459)
Rizki Tri Setiawan	(1301194012)

PROGRAM STUDI INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY

2021/2022

Tugas Desain Database

1. Pengertian ORM

ORM (*Object Relation Mapping*) merupakan teknik yang merubah suatu *table* menjadi sebuah *object* yang nantinya mudah untuk digunakan. Pada laravel dikenal istilah Eloquent ORM dimana pembuatan ORM dilakukan dengan membuat model dengan flag migration

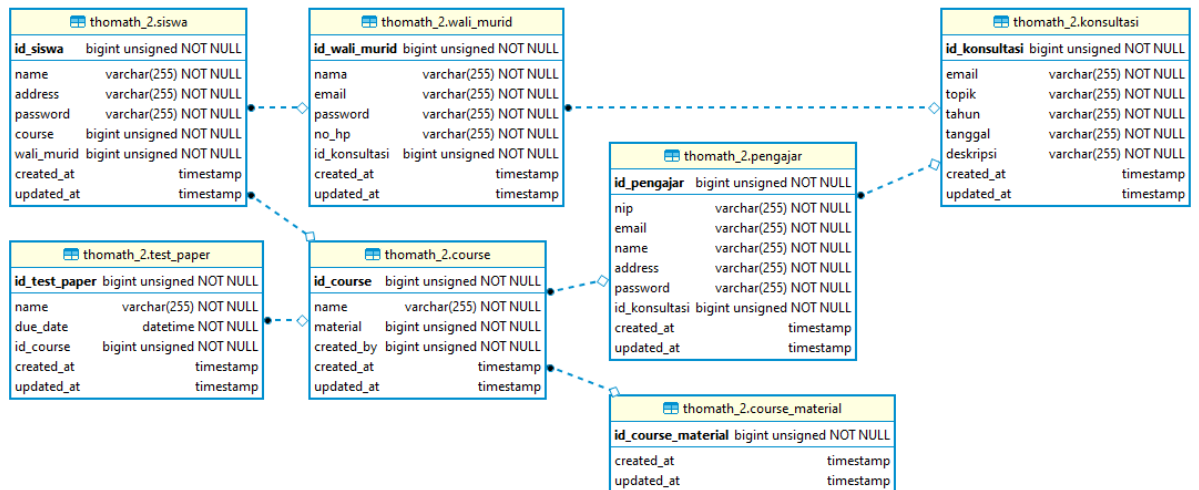
```
php artisan make:model Pengajar --migration
```

Dan hasil dari perintah tersebut adalah sebuah Model dan Table yang sudah terhubung

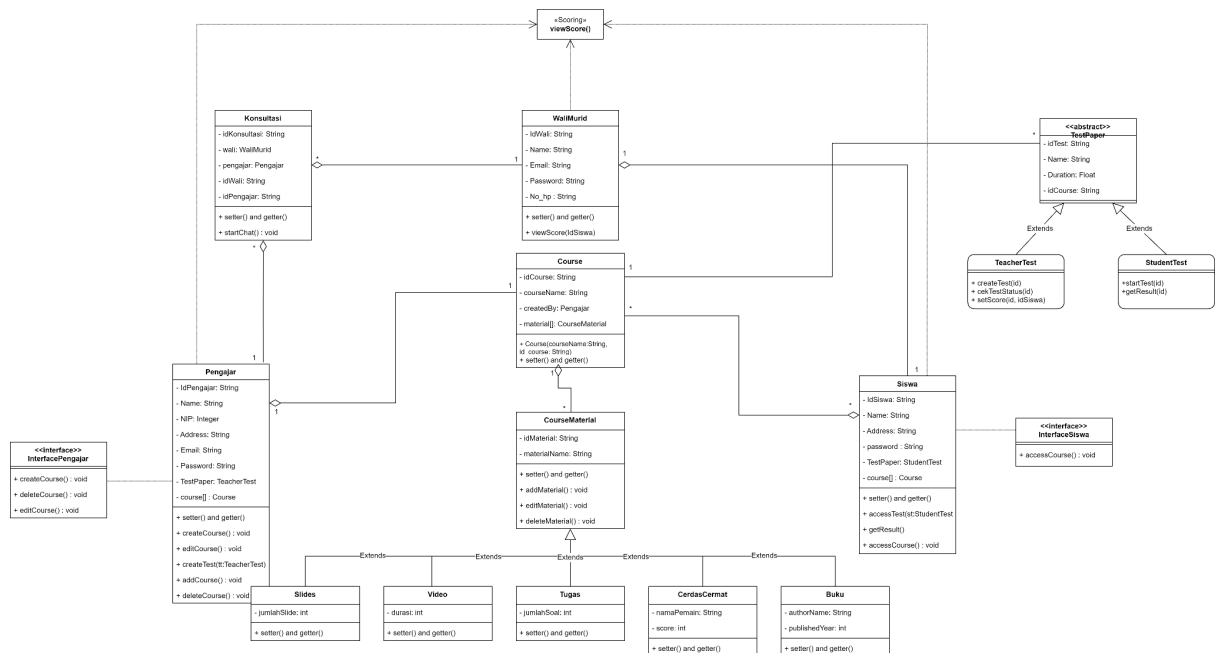
```
Pengajar.php X
app > Models > Pengajar.php > ...
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7
8  class Pengajar extends Model
9  {
10     use HasFactory;
11 }
12
```

```
2022_04_03_151458_create_pengajar_table.php ●
database > migrations > 2022_04_03_151458_create_pengajar_table.php
6
7  return new class extends Migration
8  {
9      /**
10       * Run the migrations.
11       *
12       * @return void
13       */
14     public function up()
15     {
16         Schema::create('pengajar', function (Blueprint $table) {
17             $table->id();
18             $table->timestamps();
19         });
20     }
21
22     /**
23      * Reverse the migrations.
24      *
25      * @return void
26      */
27     public function down()
28     {
29         Schema::dropIfExists('pengajar');
30     }
31 };
```

2. Relasi antar tabel



Revisi Class Diagram



a. Siswa - Wali Murid (One to One)

Kami memetakan relasi one-to-one untuk tabel siswa dan wali murid dengan meng-assign id wali_murid menjadi unique. Artinya 1 siswa hanya dapat memiliki 1 wali murid saja.

```

public function up()
{
    Schema::create('siswa', function (Blueprint $table) {
        $table->id('id_siswa');
        $table->string('name');
        $table->string('address');
        $table->string('password');
        $table->unsignedBigInteger('course');
        $table->unsignedBigInteger('wali_murid')->unique();
        $table->foreign('course')->references('id_course')->on('course');
        $table->foreign('wali_murid')->references('id_wali_murid')->on('wali_murid');
        $table->timestamps();
    });
}

```

b. Siswa - Course (Many to Many)

Kami memetakan relasi many-to-many untuk tabel siswa dan course. Artinya banyak siswa bisa mengakses banyak course.

```
public function up()
{
    Schema::create('siswa', function (Blueprint $table) {
        $table->id('id_siswa');
        $table->string('name');
        $table->string('address');
        $table->string('password');
        $table->unsignedBigInteger('course');
        $table->unsignedBigInteger('wali_murid')->unique();
        $table->foreign('course')->references('id_course')->on('course');
        $table->foreign('wali_murid')->references('id_wali_murid')->on('wali_murid');
        $table->timestamps();
    });
}
```

c. Wali Murid - Konsultasi (One to Many)

Kami memetakan relasi one-to-many untuk tabel wali murid dan konsultasi. Artinya 1 wali murid dapat melakukan banyak konsultasi dengan id_konsultasi yang berbeda.

```
public function up()
{
    Schema::create('wali_murid', function (Blueprint $table) {
        $table->id('id_wali_murid');
        $table->string('nama');
        $table->string('email')->unique();
        $table->string('password');
        $table->string('no_hp');
        $table->unsignedBigInteger('id_konsultasi')->unique();
        $table->foreign('id_konsultasi')->references('id_konsultasi')->on('konsultasi');
        $table->timestamps();
    });
}
```

d. Pengajar - Konsultasi (One to Many)

Kami memetakan relasi one-to-many untuk tabel pengajar dan konsultasi. Artinya 1 pengajar dapat melakukan banyak konsultasi dengan id_konsultasi yang berbeda.

```
public function up()
{
    Schema::create('pengajar', function (Blueprint $table) {
        $table->id('id_pengajar');
        $table->string('nip');
        $table->string('email');
        $table->string('name');
        $table->string('address');
        $table->string('password');
        $table->unsignedBigInteger('id_konsultasi')->unique();
        $table->foreign('id_konsultasi')->references('id_konsultasi')->on('konsultasi');
        $table->timestamps();
    });
}
```

e. Course - Pengajar (One to One)

Kami memetakan relasi one-to-one untuk tabel course dan pengajar. Artinya 1 course beserta materinya hanya bisa dibuat oleh 1 pengajar.

```

public function up()
{
    Schema::create('course', function (Blueprint $table) {
        $table->id('id_course');
        $table->string('name');
        $table->unsignedBigInteger('material');
        $table->unsignedBigInteger('created_by');
        $table->foreign('material')->references('id_course_material')->on('course_material');
        $table->foreign('created_by')->references('id_pengajar')->on('pengajar')->unique();
        $table->timestamps();
    });
}

```

f. Course - Test Paper (One to Many)

Kami memetakan relasi one-to-many untuk tabel course dan test paper. Artinya 1 course dapat memiliki banyak test.

```

public function up()
{
    Schema::create('test_paper', function (Blueprint $table) {
        $table->id('id_test_paper');
        $table->string('name');
        $table->dateTime('due_date');
        $table->unsignedBigInteger('id_course')->unique();
        $table->foreign('id_course')->references('id_course')->on('course');
        $table->timestamps();
    });
}

```

g. Course - Course Material (One to Many)

Kami memetakan relasi one-to-many untuk tabel course dan course material. Artinya 1 course dapat memiliki banyak materi.

```

public function up()
{
    Schema::create('course', function (Blueprint $table) {
        $table->id('id_course');
        $table->string('name');
        $table->unsignedBigInteger('material');
        $table->unsignedBigInteger('created_by');
        $table->foreign('material')->references('id_course_material')->on('course_material')->unique();
        $table->foreign('created_by')->references('id_pengajar')->on('pengajar')->unique();
        $table->timestamps();
    });
}

```

3. Tabel di database

Berikut ini merupakan keseluruhan tabel yang dibangun pada database.

