

MODUL 7

ORM ELOQUENT DAN CRUD

Topik

Membuat CRUD – Create Read Update Delete Data dengan Framework Laravel

Tujuan

1. Mahasiswa mampu memahami konsep ORM
2. Mahasiswa mampu melakukan operasi CRUD dengan ORM

A. Pengenalan ORM

Eloquent ORM (Object Relation Mapping) adalah sebuah fitur dari Lumen / Laravel yang di dalamnya terdapat fungsi-fungsi active record (query SQL) untuk mengelola data di database. Dengan menggunakan Eloquent ORM, database bisa kita bungkus ke dalam objek, sehingga operasi CRUD (Create, Read, Update, Delete) pada tabel database dapat dilakukan tanpa melibatkan perintah / query SQL sama sekali, bahkan sampai relasi antar tabelnya juga.

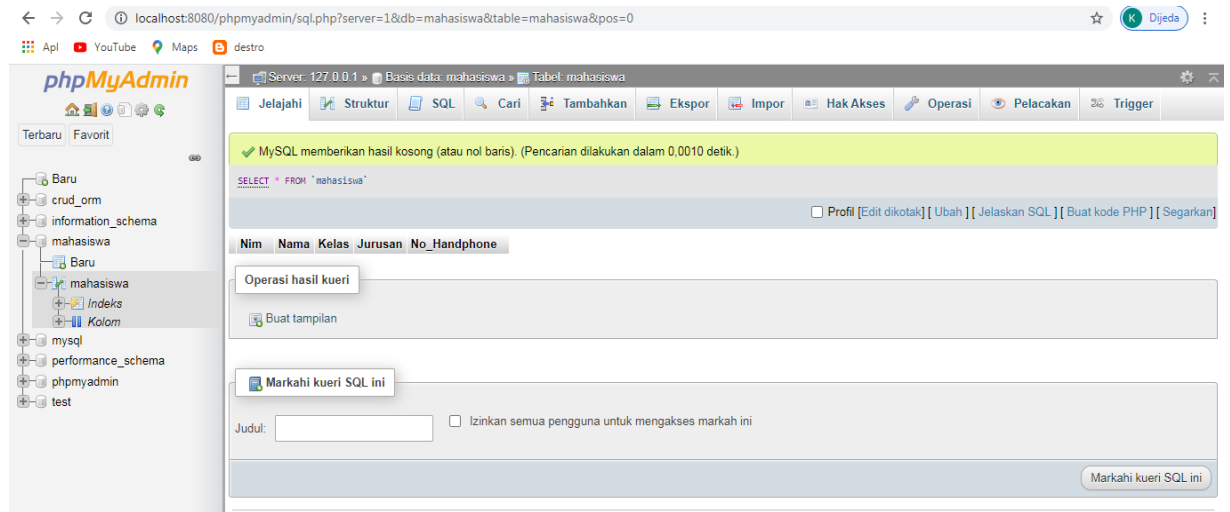
Agar Eloquent ORM dapat bekerja, kita harus terlebih dahulu membuat Model yang mewakili tabel database terkait. Sebagai contoh, tabel mahasiswa diwakili oleh Model mahasiswa (penamaan tabel dan Model disarankan sama). Pada sistem MVC (Model View Controller), Model ditandai sebagai M, merupakan sebuah class yang terdiri dari beberapa variabel dan method. Melalui class ini lah jalan pintas ke query SQL nantinya dilakukan.

B. Praktikum CRUD Dengan ORM

1. ORM yang disertakan dengan Laravel menyediakan implementasi ActiveRecord yang baik dan sederhana untuk bekerja dengan database. Setiap tabel database memiliki "Model" yang sesuai yang digunakan untuk berinteraksi dengan tabel itu. Sebelum memulai, pastikan untuk mengonfigurasi koneksi di database.

config/database.php

2. Buatlah database di dalam phpMyadmin dengan nama database **mahasiswa** dengan nama tabel mahasiswa yang isinya adalah: **Nim, Nama, Kelas, Jurusan, No_Handphone**.



3. Buatlah project laravel baru dengan nama **mahasiswa**. Buka Terminal, Tulislah perintah berikut:

laravel new mahasiswa



4. Selanjutnya lakukanlah konfigurasi database di Laravel. Untuk dapat melakukannya maka bukalah database di file **.env** pada project rubahlah **DB_DATABASE=mahasiswa** seperti dibawah ini:

```

1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:6B6ASgn18vOjj3pzHp4sB2S42FIX8r+YRZHbwRTnISQ=
4 APP_DEBUG=true
5 APP_URL=http://mahasiswa.test
6
7 LOG_CHANNEL=stack
8 LOG_LEVEL=debug
9
10 DB_CONNECTION=mysql
11 DB_HOST=127.0.0.1
12 DB_PORT=3306
13 DB_DATABASE=mahasiswa
14 DB_USERNAME=root
15 DB_PASSWORD=

```

5. Install Composer Create-Project seperti berikut. Ketikkan pada Command Line Kalian:

composer create-project --prefer-dist laravel/laravel mahasiswa

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL

Windows PowerShell
Copyright (C) 2009 Microsoft Corporation. All rights reserved.

PS C:\xampp\htdocs\mahasiswa> composer create-project --prefer-dist laravel/laravel mahasiswa
Creating a "laravel/laravel" project at "./mahasiswa"
Installing laravel/laravel (v8.5.10)
- Installing laravel/laravel (v8.5.10): Extracting archive
Created project in C:\xampp\htdocs\mahasiswa\mahasiswa
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information

```

6. Langkah selanjutnya adalah migrasi, maka langsung jalankan perintah migrasi karena laravel sudah menyediakan file migrasi untuk tabel mahasiswa. Sekarang ketikkan perintah berikut:

php artisan migrate

```

PS C:\xampp\htdocs\mahasiswa> php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated: 2014_10_12_000000_create_users_table (1,099.06ms)
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated: 2014_10_12_100000_create_password_resets_table (412.02ms)
Migrating: 2019_08_19_000000_create_failed_jobs_table
Migrated: 2019_08_19_000000_create_failed_jobs_table (612.04ms)
PS C:\xampp\htdocs\mahasiswa>

```

7. Membuat Resource Controller dan Route yang berfungsi untuk route CRUD sehingga tidak perlu repot-repot membuat masing-masing route seperti post, get, delete dan update. Berikut perintahnya: sekarang tambahkan kode pada route/web.php

php artisan make:controller MahasiswaController --resource

```
PS C:\xampp\htdocs\mahasiswa> php artisan make:controller MahasiswaController --resource
Controller created successfully.
PS C:\xampp\htdocs\mahasiswa> █
```

sekarang tambahkan kode pada **route/web.php**

```
use Illuminate\Support\Facades\Route;
use App\Http\Controllers\MahasiswaController;
use Illuminate\Http\Request;

Route::resource('mahasiswas', MahasiswaController::class);
```

8. Pada file Controller yang kita dapat secara otomatis akan terdapat 7 fungsi berikut yang dapat kita gunakan untuk membuat operasi CRUD.
- index()
 - create()
 - store()
 - show()
 - edit()
 - update()
 - destroy()
9. Selanjutnya bukalah **app/Http/Controllers/MahasiswaController.php** kemudian ketikkan kodenya seperti berikut ini:

```
<?php

namespace App\Http\Controllers;

use App\Models\Mahasiswa;
use Illuminate\Http\Request;

class MahasiswaController extends Controller
{
    /**
     * Display a listing of the resource.
     */
}
```

```

* @return \Illuminate\Http\Response
*/
public function index()
{
    //fungsi eloquent menampilkan data menggunakan pagination
    $mahasiswas = Mahasiswa::all(); // Mengambil semua isi tabel
    $posts = Mahasiswa::orderBy('Nim', 'desc')->paginate(6);
    return view('mahasiswas.index', compact('mahasiswas'));
    with('i', (request()->input('page', 1) - 1) * 5);
}
public function create()
{
    return view('mahasiswas.create');
}
public function store(Request $request)
{
    //melakukan validasi data
    $request->validate([
        'Nim' => 'required',
        'Nama' => 'required',
        'Kelas' => 'required',
        'Jurusan' => 'required',
        'No_Handphone' => 'required',
    ]);

    //fungsi eloquent untuk menambah data
    Mahasiswa::create($request->all());

    //jika data berhasil ditambahkan, akan kembali ke halaman utama
    return redirect()->route('mahasiswas.index')
        ->with('success', 'Mahasiswa Berhasil Ditambahkan');
}

public function show($Nim)
{
    //menampilkan detail data dengan menemukan/berdasarkan Nim Mahasiswa
    $Mahasiswa = Mahasiswa::find($Nim);
    return view('mahasiswas.detail', compact('Mahasiswa'));
}

public function edit($Nim)
{
    //menampilkan detail data dengan menemukan berdasarkan Nim Mahasiswa untuk
    diedit
    $Mahasiswa = Mahasiswa::find($Nim);
    return view('mahasiswas.edit', compact('Mahasiswa'));
}

public function update(Request $request, $Nim)
{

```

```

//melakukan validasi data
$request->validate([
    'Nim' => 'required',
    'Nama' => 'required',
    'Kelas' => 'required',
    'Jurusan' => 'required',
    'No_Handphone' => 'required',
]);

//fungsi eloquent untuk mengupdate data inputan kita
Mahasiswa::find($Nim)->update($request->all());

//jika data berhasil diupdate, akan kembali ke halaman utama
return redirect()->route('mahasiswas.index')
->with('success', 'Mahasiswa Berhasil Diupdate');
}
public function destroy( $Nim)
{
//fungsi eloquent untuk menghapus data
Mahasiswa::find($Nim)->delete();
return redirect()->route('mahasiswas.index')
-> with('success', 'Mahasiswa Berhasil Dihapus');
}
};

```

Untuk penjelasan tiap kode silahkan baca di komen pada kode di atas. Jadi intinya setiap aksi yang berhasil kita lakukan, akan memberikan pesan success yang nantinya akan ditangkap oleh fungsi **session** pada view.

- Setting nilai **\$fillable** di **Model User** berdasarkan data yang ingin dapat kita tambah atau edit, pada kasus ini **Nim, Nama, Kelas, Jurusan dan No_Handphone**.

```

<?php

namespace App\Models;

use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Foundation\Auth\Mahasiswa as Authenticatable;
use Illuminate\Notifications\Notifiable;
use Illuminate\Database\Eloquent\Model; //Model Eloquent

class Mahasiswa extends Model //Definisi Model
{
    protected $table="mahasiswa"; // Eloquent akan membuat model mahasiswa
    menyimpan record di tabel mahasiswas
    public $timestamps= false;
    protected $primaryKey = 'Nim'; // Memanggil isi DB Dengan primaryKey
    /**
     * The attributes that are mass assignable.

```

```

*
* @var array
*/
protected $fillable = [
    'Nim',
    'Nama',
    'Kelas',
    'Jurusan',
    'No_Handphone',
];
};

```

11. Selanjutnya buatlah folder **users** pada **resources/views**. Terakhir buat file-file berikut ini didalam **users**.

- layout.blade.php
- index.blade.php
- create.blade.php
- edit.blade.php
- detail.blade.php

Kemudian ketikkan kode berikut pada masing-masing file.

a. Layout.blade.php

```

<html>
  <head>
    <title>Sistem Informasi Mahasiswa</title>
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-Vkoo8x4CGs03+Hhvxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymous">
    <script src="https://code.jquery.com/jquery-3.4.1.slim.min.js" integrity="sha384-J6qa4849blE2+poT4WnyKhv5vZF5SrPo0iEjwBvKU7imGFAV0wwj1yYfoRSJoZ+n" crossorigin="anonymous"></script>
    <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js" integrity="sha384-Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvoxMfooAo" crossorigin="anonymous"></script>
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/js/bootstrap.min.js" integrity="sha384-wfSDF2E50Y2D1uUdj03uMBJnjuUD4Ih7YwaYd1iqfktj0Uod8GCExl30g8ifwB6" crossorigin="anonymous"></script>
  </head>
  <body>
    <div class="container">
      @yield('content')
    </div>
  </body>
</html>

```

b. Index.blade.php

```
@extends('mahasiswa.layout')

@section('content')
    <div class="row">
        <div class="col-lg-12 margin-tb">
            <div class="pull-left mt-2">
                <h2>JURUSAN TEKNOLOGI INFORMASI-POLITEKNIK NEGERI MALANG</h2>
            </div>
            <div class="float-right my-2">
                <a class="btn btn-
success" href="{{ route('mahasiswa.create') }}"> Input Mahasiswa</a>
            </div>
        </div>
    </div>

    @if ($message = Session::get('success'))
        <div class="alert alert-success">
            <p>{{ $message }}</p>
        </div>
    @endif

    <table class="table table-bordered">
        <tr>
            <th>Nim</th>
            <th>Nama</th>
            <th>Kelas</th>
            <th>Jurusan</th>
            <th>No_Handphone</th>
            <th width="280px">Action</th>
        </tr>
        @foreach ($mahasiswas as $Mahasiswa)
            <tr>
                <td>{{ $Mahasiswa->Nim }}</td>
                <td>{{ $Mahasiswa->Nama }}</td>
                <td>{{ $Mahasiswa->Kelas }}</td>
                <td>{{ $Mahasiswa->Jurusan }}</td>
                <td>{{ $Mahasiswa->No_Handphone }}</td>
                <td>
                    <form action="{{ route('mahasiswa.destroy',$Mahasiswa-
>Nim) }}" method="POST">

                        <a class="btn btn-
info" href="{{ route('mahasiswa.show',$Mahasiswa->Nim) }}">Show</a>

                        <a class="btn btn-
primary" href="{{ route('mahasiswa.edit',$Mahasiswa->Nim) }}">Edit</a>

                    </form>
                </td>
            </tr>
        @endforeach
    </table>

    @csrf
</pre>
```



```

                @method('DELETE')

                <button type="submit" class="btn btn-danger">Delete</button>
            </form>
        </td>
    </tr>
@endforeach
</table>
@endsection

```

c. Create.blade.php

```

@extends('mahasiswas.layout')

@section('content')

<div class="container mt-5">

    <div class="row justify-content-center align-items-center">
        <div class="card" style="width: 24rem;">
            <div class="card-header">
                Tambah Mahasiswa
            </div>
            <div class="card-body">
                @if ($errors->any())
                    <div class="alert alert-danger">
                        <strong>Whoops!</strong> There were some problems with your i
nput.<br><br>
                        <ul>
                            @foreach ($errors->all() as $error)
                                <li>{{ $error }}</li>
                            @endforeach
                        </ul>
                    </div>
                @endif
                <form method="post" action="{{ route('mahasiswas.store') }}" id="myFo
rm">
                    @csrf
                    <div class="form-group">
                        <label for="Nim">Nim</label>
                        <input type="text" name="Nim" class="form-
control" id="Nim" aria-describedby="Nim" >
                    </div>
                    <div class="form-group">
                        <label for="Nama">Nama</label>
                        <input type="text" name="Nama" class="form-
control" id="Nama" aria-describedby="Nama" >
                    </div>
                    <div class="form-group">
                        <label for="Kelas">Kelas</label>

```

```

        <input type="Kelas" name="Kelas" class="form-
control" id="Kelas" aria-describedby="password" >
    </div>
    <div class="form-group">
        <label for="Jurusan">Jurusan</label>
        <input type="Jurusan" name="Jurusan" class="form-
control" id="Jurusan" aria-describedby="Jurusan" >
    </div>
    <div class="form-group">
        <label for="No_Handphone">No_Handphone</label>

        <input type="No_Handphone" name="No_Handphone" class="form-
control" id="No_Handphone" aria-describedby="No_Handphone" >
    </div>
    <button type="submit" class="btn btn-primary">Submit</button>
</form>
</div>
</div>
</div>
@endsection

```

d. Edit.blade.php

```

@extends('mahasiswas.layout')

@section('content')

<div class="container mt-5">

    <div class="row justify-content-center align-items-center">
        <div class="card" style="width: 24rem;">
            <div class="card-header">
                Edit Mahasiswa
            </div>
            <div class="card-body">
                @if ($errors->any())
                    <div class="alert alert-danger">
                        <strong>Whoops!</strong> There were some problems with your i
nput.<br><br>
                        <ul>
                            @foreach ($errors->all() as $error)
                                <li>{{ $error }}</li>
                            @endforeach
                        </ul>
                    </div>
                @endif
                <form method="post" action="{{ route('mahasiswas.update', $Mahasiswa-
>Nim) }}" id="myForm">
                    @csrf
                    @method('PUT')

```

```

        <div class="form-group">
            <label for="Nim">Nim</label>
            <input type="text" name="Nim" class="form-
control" id="Nim" value="{{ $Mahasiswa->Nim }}" aria-
describedby="Nim" >
        </div>
        <div class="form-group">
            <label for="Nama">Nama</label>
            <input type="text" name="Nama" class="form-
control" id="Nama" value="{{ $Mahasiswa->Nama }}" aria-
describedby="Nama" >
        </div>
        <div class="form-group">
            <label for="Kelas">Kelas</label>
            <input type="Kelas" name="Kelas" class="form-
control" id="Kelas" value="{{ $Mahasiswa->Kelas }}" aria-
describedby="Kelas" >
        </div>
        <div class="form-group">
            <label for="Jurusan">Jurusan</label>
            <input type="Jurusan" name="Jurusan" class="form-
control" id="Jurusan" value="{{ $Mahasiswa->Jurusan }}" aria-
describedby="Jurusan" >
        </div>
        <div class="form-group">
            <label for="No_Handphone">No_Handphone</label>

            <input type="No_Handphone" name="No_Handphone" class="form-
control" id="No_Handphone" value="{{ $Mahasiswa->No_Handphone }}" aria-
describedby="No_Handphone" >
        </div>
        <button type="submit" class="btn btn-primary">Submit</button>
    </form>
</div>
</div>
@endsection

```

e. Detail.blade.php

```

@extends('mahasiswas.layout')

@section('content')

<div class="container mt-5">
    <div class="row justify-content-center align-items-center">
        <div class="card" style="width: 24rem;">
            <div class="card-header">
                Detail Mahasiswa
            </div>

```

```

        <div class="card-body">
            <ul class="list-group list-group-flush">
                <li class="list-group-item"><b>Nim: </b>{{$Mahasiswa-
>Nim}}</li>
                <li class="list-group-item"><b>Nama: </b>{{$Mahasiswa-
>Nama}}</li>
                <li class="list-group-item"><b>Kelas: </b>{{$Mahasiswa-
>Kelas}}</li>
                <li class="list-group-item"><b>Jurusan: </b>{{$Mahasiswa-
>Jurusan}}</li>
                <li class="list-group-
item"><b>No_Handphone: </b>{{$Mahasiswa->No_Handphone}}</li>
            </ul>
        </div>
        <a class="btn btn-success mt-
3" href="{{ route('mahasiswas.index') }}">Kembali</a>

    </div>
</div>
</div>
@endsection

```

12. Terakhir untuk dapat di jalankan projek diatas maka ketikkan command-line dibawah ini:

php artisan serve

```

PS C:\xampp\htdocs\mahasiswa> php artisan serve
Starting Laravel development server: http://127.0.0.1:8000

```

13. Hasilnya dapat dilihat dibawah ini:

a. Tampilan awal

Nim	Nama	Kelas	Jurusan	No_Handphone	Action
-----	------	-------	---------	--------------	--------

b. Tampilan Input Mahasiswa

Tambah Mahasiswa

Nim

Nama

Kelas

Jurusan

No_Handphone

Submit

c. Tampilan Edit

Edit Mahasiswa

Nim

12345

Nama

Ane Fany

Kelas

TI2H

Jurusan

Teknologi Informasi

No_Handphone

085469199998

Submit

d. Tampilan Detail

Detail Mahasiswa

Nim: 12345

Nama: Ane Fany

Kelas: TI2H

Jurusan: Teknologi Informasi

No_Handphone: 085469199998

Kembali

C. Tugas Praktikum

1. Tambahkan 2 kolom lagi pada tabel dengan isi tabel yaitu E-mail dan Tanggal Lahir.
2. Buatlah view yang ditampilkan hanya 5 mahasiswa dan dapat link ke mahasiswa berikutnya.
3. Tambahkan menu search untuk mencari nama mahasiswa yang diinginkan.