

# **LAPORAN TUGAS PEMROGRAMAN WEB LANJUT**

**JOBSHEET 10**



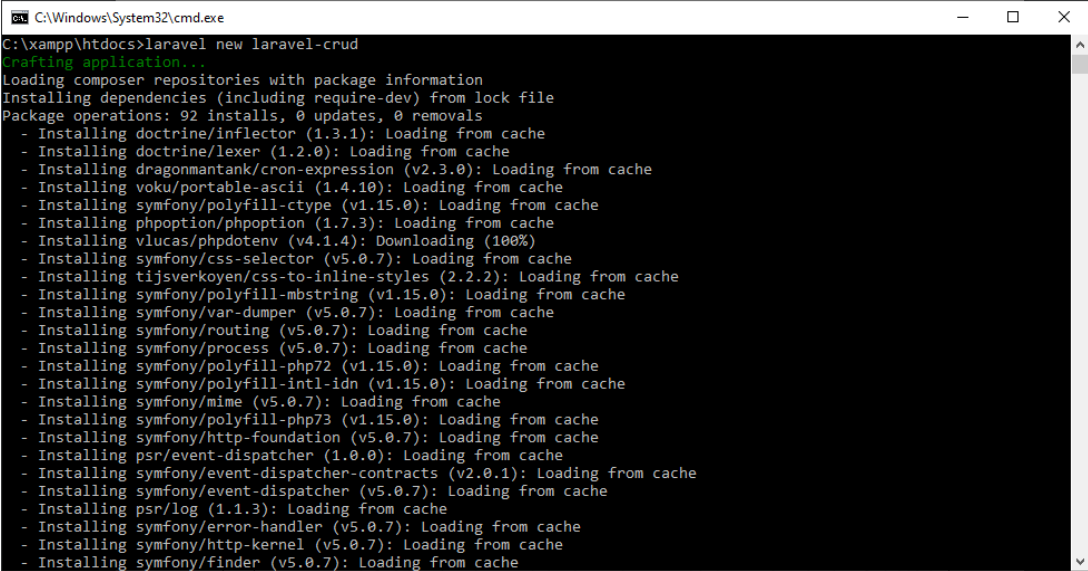
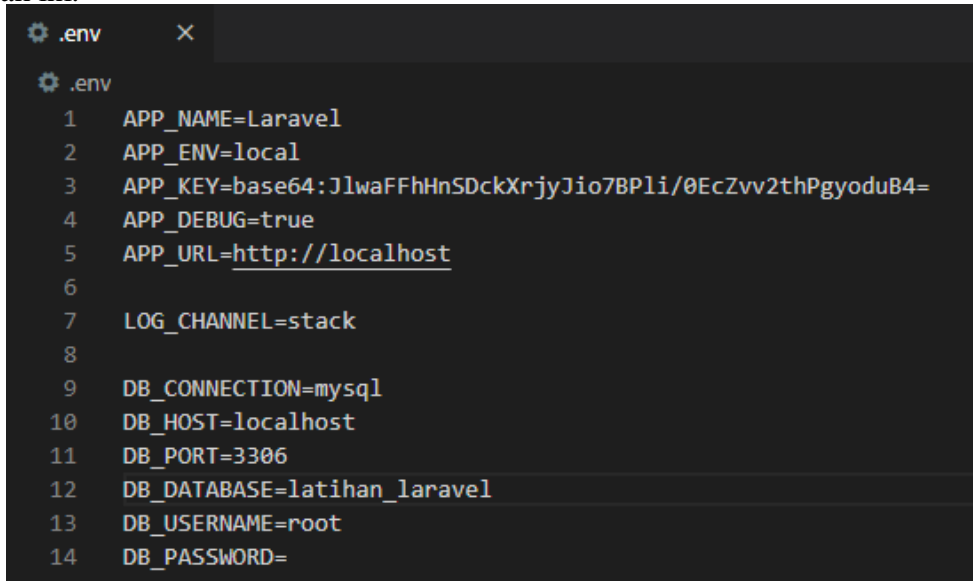
**Oleh :**

**Rahmad Alfian Maskuri / 1841720048**

**PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNOLOGI INFORMASI  
POLITEKNIK NEGERI MALANG  
2020**

Praktikum – Bagian 1: Membuat CRUD di Laravel menggunakan Query Builder Query Builder adalah salah satu fitur untuk menjalankan query database. Laravel menyediakan fungsifungsi yang dapat digunakan untuk menjalankan query sehingga penulisan query menjadi lebih singkat dan efisien.

**a. Konfigurasi Database dan Pembuatan Tabel di MySQL**

No	Langkah Percobaan
1	<p>Buatlah project Laravel baru dengan nama <b>laravel-crud</b>. Buka command prompt, tuliskan perintah berikut.</p>  <pre> C:\xampp\htdocs&gt;laravel new laravel-crud Crafting application... Loading composer repositories with package information Installing dependencies (including require-dev) from lock file Package operations: 92 installs, 0 updates, 0 removals - Installing doctrine/inflector (1.3.1): Loading from cache - Installing doctrine/lexer (1.2.0): Loading from cache - Installing dragonmantank/cron-expression (v2.3.0): Loading from cache - Installing voku/portable-ascii (1.4.10): Loading from cache - Installing symfony/polyfill-ctype (v1.15.0): Loading from cache - Installing phpoption/phpoption (1.7.3): Loading from cache - Installing vlucas/phpdotenv (v4.1.4): Downloading (100%) - Installing symfony/css-selector (v5.0.7): Loading from cache - Installing tijsverkoyen/css-to-inline-styles (2.2.2): Loading from cache - Installing symfony/polyfill-mbstring (v1.15.0): Loading from cache - Installing symfony/var-dumper (v5.0.7): Loading from cache - Installing symfony/routing (v5.0.7): Loading from cache - Installing symfony/process (v5.0.7): Loading from cache - Installing symfony/polyfill-php72 (v1.15.0): Loading from cache - Installing symfony/polyfill-intl-idn (v1.15.0): Loading from cache - Installing symfony/mime (v5.0.7): Loading from cache - Installing symfony/polyfill-php73 (v1.15.0): Loading from cache - Installing symfony/http-foundation (v5.0.7): Loading from cache - Installing psr/event-dispatcher (1.0.0): Loading from cache - Installing symfony/event-dispatcher-contracts (v2.0.1): Loading from cache - Installing symfony/event-dispatcher (v5.0.7): Loading from cache - Installing psr/log (1.1.3): Loading from cache - Installing symfony/error-handler (v5.0.7): Loading from cache - Installing symfony/http-kernel (v5.0.7): Loading from cache - Installing symfony/finder (v5.0.7): Loading from cache </pre>
2	<p>Selanjutnya kita lakukan konfigurasi database di Laravel. Untuk melakukan konfigurasi database, bukalah file <b>.env</b> pada project laravel-crud. Ubah seperti di bawah ini.</p>  <pre> APP_NAME=Laravel APP_ENV=local APP_KEY=base64:JlwaFFhHnSDckXrjyJio7BPli/0EcZvv2thPgyoduB4= APP_DEBUG=true APP_URL=http://localhost  LOG_CHANNEL=stack  DB_CONNECTION=mysql DB_HOST=localhost DB_PORT=3306 DB_DATABASE=latihan_laravel DB_USERNAME=root DB_PASSWORD= </pre>
3	<p>Jalankan xampp, selanjutnya buat tabel dengan nama <b>mahasiswa</b> di mysql pada database latihan_laravel.</p>

Server: 127.0.0.1 » Database: latihan\_laravel » Table: mahasiswa

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
2	nama	varchar(100)	latin1_swedish_ci		No	None			Change Drop More
3	nim	varchar(10)	latin1_swedish_ci		No	None			Change Drop More
4	email	varchar(50)	latin1_swedish_ci		No	None			Change Drop More
5	jurusan	varchar(20)	latin1_swedish_ci		No	None			Change Drop More

4 Isilah beberapa data pada tabel **mahasiswa** tersebut.

Server: 127.0.0.1 » Database: latihan\_laravel » Table: mahasiswa

Showing rows 0 - 1 (2 total, Query took 0.0015 seconds.)

```
SELECT * FROM `mahasiswa`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Re

Show all | Number of rows: 25

+ Options

	id	nama	nim	email	jurusan
<input type="checkbox"/> Edit Copy Delete	1	nanda	1001001001	nandan@gmail.com	teknik informatika
<input type="checkbox"/> Edit Copy Delete	2	wahyu	1001001002	wahyu@gmail.com	teknik elektro

## b. Menampilkan Data dari Database

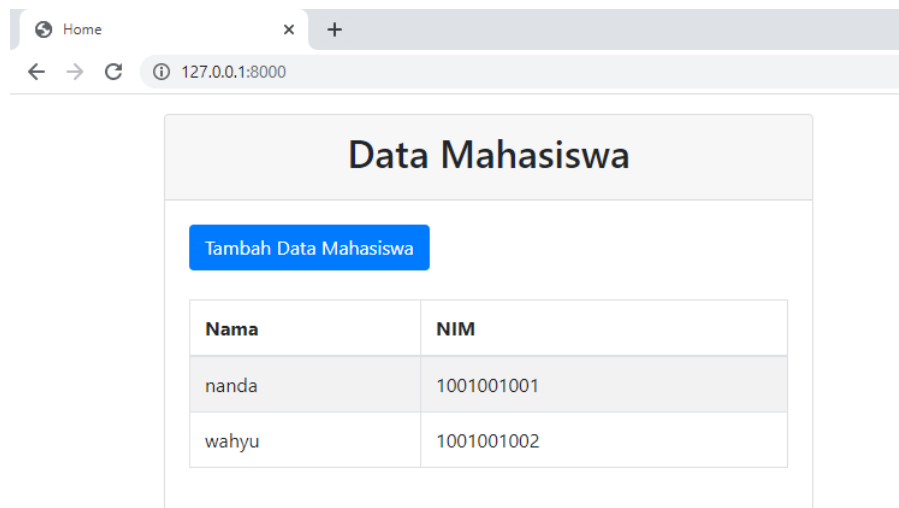
No	Langkah Percobaan
1	<p>Setelah kita memiliki beberapa data pada tabel mahasiswa, kita akan mencoba untuk menampilkan data tersebut ketika project dijalankan.</p> <p>Pertama, buatlah <i>route</i> pada <b>routes/web.php</b> sehingga ketika pertama kali project dijalankan akan terbuka halaman yang menampilkan data.</p>  <pre> 1 &lt;?php 2 3 use Illuminate\Support\Facades\Route; 4 5 /* 6  ----- 7   Web Routes 8  ----- 9   10   Here is where you can register web routes for your appl 11   routes are loaded by the RouteServiceProvider within a 12   contains the "web" middleware group. Now create somethi 13   14 */ 15 16 //Route::get('/', function () { 17 //    return view('welcome'); 18 // }); 19 20 Route::get('/', 'MahasiswaController@index'); 21 </pre>

2	<p>Buat controller baru menggunakan command prompt yaitu MahasiswaController menggunakan php artisan</p> <p><b>cd laravel-crud</b></p> <p><b>php artisan make:controller MahasiswaController</b></p> <pre>C:\xampp\htdocs\laravel-crud&gt;php artisan make:controller MahasiswaController Controller created successfully.</pre>
3	<p>Buat method <b>index</b> pada MahasiswaController.php pada folder <b>app/Http/Controllers</b></p> <pre>.env web.php MahasiswaController.php X app &gt; Http &gt; Controllers &gt; MahasiswaController.php 1  &lt;?php 2 3  namespace App\Http\Controllers; 4 5  use Illuminate\Http\Request; 6  use Illuminate\Support\Facades\DB; 7 8  class MahasiswaController extends Controller 9  { 10     public function index() 11     { 12         //Ambil data tabel Mhs 13         \$mahasiswa = DB::table('mahasiswa')-&gt;get(); 14 15         //Kirim data Mhs ke view index 16         return view('index',['mahasiswa' =&gt; \$mahasiswa]); 17     } 18 19 } 20</pre>
4	<p>Selanjutnya kita akan membuat view untuk menampilkan data mahasiswa dengan nama index.blade.php. Tetapi agar pembuatan view selanjutnya menjadi lebih mudah, terlebih dahulu kita akan membuat template blade (seperti file header-footer pada pembahasan CI) dengan nama master.blade.php pada folder resources/views.</p> <pre>.env web.php MahasiswaController.php master.blade.php X resources &gt; views &gt; master.blade.php 1  &lt;html&gt; 2 3  &lt;head&gt; 4      &lt;meta charset="utf-8"&gt; 5      &lt;meta name="viewport" content="width=device-width, initial-scale=1"&gt; 6      &lt;link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css" integrity="sha384-ggOyR01XCbMQv3XIpma3 7  &lt;title&gt; @yield('title') &lt;/title&gt; 8  &lt;/head&gt; 9 10 &lt;body&gt; 11     &lt;div class="container"&gt; 12         &lt;div class="row mt-3"&gt; 13             &lt;div class="col-md-6"&gt; 14                 &lt;div class="card-header text-center"&gt; 15                     &lt;!-- Judul Halaman --&gt; 16                     &lt;h2&gt; @yield('judul_halaman') &lt;/h2&gt; 17                 &lt;/div&gt; 18                 &lt;div class="card-body"&gt; 19                     &lt;!-- Konten Halaman --&gt; 20                     @yield('konten') 21                 &lt;/div&gt; 22             &lt;/div&gt; 23         &lt;/div&gt; 24     &lt;/div&gt; 25 &lt;/body&gt; 26 27 &lt;/html&gt;</pre>

- 5 Sekarang buat file index.blade.php yang menerapkan template master.blade.php dan akan digunakan untuk menampilkan data.

```
resources > views > index.blade.php
1  @extends('master')
2
3  @section('title', 'Home')
4
5  @section('judul_halaman', 'Data Mahasiswa')
6  @section('konten')
7      <a href="/mahasiswa/tambah" class="btn btn-primary">Tambah Data Mahasiswa</a>
8      <br/>
9      <br/>
10     <table class="table table-bordered table-hover table-striped">
11         <thead>
12             <th>Nama</th>
13             <th>NIM</th>
14         </thead>
15         <tbody>
16             @foreach($mahasiswa as $mhs)
17                 <tr>
18                     <td>{{ $mhs->nama }}</td>
19                     <td>{{ $mhs->nim }}</td>
20                 </tr>
21             @endforeach
22         </tbody>
23     </table>
24 @endsection
```

- 6 Jalankan command prompt, tuliskan perintah untuk menjalankan project laravel-crud php artisan serve Buka browser dan ketikkan localhost:8000, maka akan tampil sebagai berikut



c. Memasukkan Data (Create) ke Database

No	Langkah Percobaan
1	<p>Buatlah route baru pada routes/web.php dengan nama /mahasiswa/tambah yang akan menjalankan fungsi tambah pada MahasiswaController</p> <pre>Route::get('/mahasiswa/tambah', 'MahasiswaController@tambah');</pre>
2	<p>Buat method tambah pada MahasiswaController.php di folder app/Http/Controllers yang akan menampilkan view tambah.</p> <pre>MahasiswaController.php X master.blade.php index.blade.php t op &gt; Http &gt; Controllers &gt; MahasiswaController.php 1  &lt;?php 2 3  namespace App\Http\Controllers; 4 5  use Illuminate\Http\Request; 6  use Illuminate\Support\Facades\DB; 7 8  class MahasiswaController extends Controller 9  { 10     public function index() 11     { 12         //Ambil data tabel Mhs 13         \$mahasiswa = DB::table('mahasiswa')-&gt;get(); 14 15         //Kirim data Mhs ke view index 16         return view('index', ['mahasiswa' =&gt; \$mahasiswa]); 17     } 18 19     public function tambah() 20     { 21         //panggil view tambah 22         return view('tambah'); 23     } 24 25 } 26</pre>
3	<p>Kemudian buatlah view tambah.blade.php yang berisi form untuk memasukkan data baru pada folder resources/views. View tambah juga mengaplikasikan master.blade.php.</p>

	<div> <div> .envweb.phpMahasiswaController.phpmaster.blade.phpindex.blade.phptambah.blade.php X </div> <div> <pre>resources &gt; views &gt; tambah.blade.php 1  @extends('master') 2 3  @section('title','Tambah Data') 4 5  @section('judul_halaman','Tambah Data Mahasiswa') 6 7  @section('konten') 8  &lt;a href="/" class="btn btn-danger"&gt;Kembali&lt;/a&gt; 9  &lt;br&gt; 10 &lt;br&gt; 11 &lt;form action="/mahasiswa/simpan" method="POST"&gt; 12     {{ csrf_field() }} 13     &lt;div class="form-group"&gt; 14         &lt;label&gt;Nama&lt;/label&gt; 15         &lt;input type="text" class="form-control" required="required" name="namamhs"&gt;&lt;br&gt; 16     &lt;/div&gt; 17     &lt;div class="form-group"&gt; 18         &lt;label&gt;NIM&lt;/label&gt; 19         &lt;input type="number" class="form-control" required="required" name="nimhs"&gt;&lt;br&gt; 20     &lt;/div&gt; 21     &lt;div class="form-group"&gt; 22         &lt;label&gt;Email&lt;/label&gt; 23         &lt;input type="email" class="form-control" required="required" name="emailmhs"&gt;&lt;br&gt; 24     &lt;/div&gt; 25     &lt;div class="form-group"&gt; 26         &lt;label&gt;Jurusan&lt;/label&gt; 27         &lt;input type="text" class="form-control" required="required" name="jurusanmhs"&gt;&lt;br&gt; 28     &lt;/div&gt; 29     &lt;input type="submit" class="btn btn-primary float-right" value="Simpan Data"&gt; 30 &lt;/form&gt; 31 @endsection</pre> </div> </div>
4	<p>Ketika tombol simpan ditekan, akan dipanggil routes /mahasiswa/simpan. Oleh karena itu, kita buat terlebih dahulu route tersebut.</p> <pre>Route::post('/mahasiswa/simpan','MahasiswaController@simpan');</pre>
5	<p>Buat method simpan pada MahasiswaController untuk menyimpan data ke database.</p> <div> <div> MahasiswaController.php Xmaster.blade.phpindex.blade.phptambah.blade.php X </div> <div> <pre>app &gt; Http &gt; Controllers &gt; MahasiswaController.php 1  use Illuminate\Http\Request; 2  use Illuminate\Support\Facades\DB; 3 4  class MahasiswaController extends Controller 5  { 6 7      public function index() 8      { 9          //Ambil data tabel Mhs 10         \$mahasiswa = DB::table('mahasiswa')-&gt;get(); 11 12         //Kirim data Mhs ke view index 13         return view('index',['mahasiswa' =&gt; \$mahasiswa]); 14     } 15 16     public function tambah() 17     { 18         //panggil view tambah 19         return view('tambah'); 20     } 21 22     public function simpan(Request \$request) 23     { 24         //tambah data ke table mhs 25         DB::table('mahasiswa')-&gt;insert([ 26             'nama' =&gt; \$request-&gt;namamhs, 27             'nim' =&gt; \$request-&gt;nimmhs, 28             'email' =&gt; \$request-&gt;emailmhs, 29             'jurusan' =&gt; \$request-&gt;jurusanmhs 30         ]); 31         return redirect('/mahasiswa'); 32     } 33 }</pre> </div> </div>

6

Kembali coba jalankan localhost:8000 dan pilih tombol ‘Tambah Data Mahasiswa’, maka akan ditampilkan halaman tambah yang berisi form untuk memasukkan data baru. Kita coba isikan data pada form tersebut.

The screenshot shows a web browser window with a single tab titled 'Tambah Data'. The address bar displays '127.0.0.1:8000/mahasiswa/tambah'. The main content area features a form with a light gray header containing the title 'Tambah Data Mahasiswa'. Below the header, there is a red button labeled 'Kembali'. The form contains four input fields: 'Nama', 'NIM', 'Email', and 'Jurusan'. At the bottom right of the form is a blue button labeled 'Simpan Data'.



Tambah Data

127.0.0.1:8000/mahasiswa/tambah

### Tambah Data Mahasiswa

Kembali

Nama

Rahmad

NIM

1001001003

Email

Rahmad@gmail.com

Jurusan

Teknik Informatika

Simpan Data

Home

New Tab

127.0.0.1:8000/mahasiswa

### Data Mahasiswa

Tambah Data Mahasiswa

Nama	NIM
nanda	1001001001
wahyu	1001001002
Rahmad	1001001003
Alfian	1001001004
Maskuri	1001001005

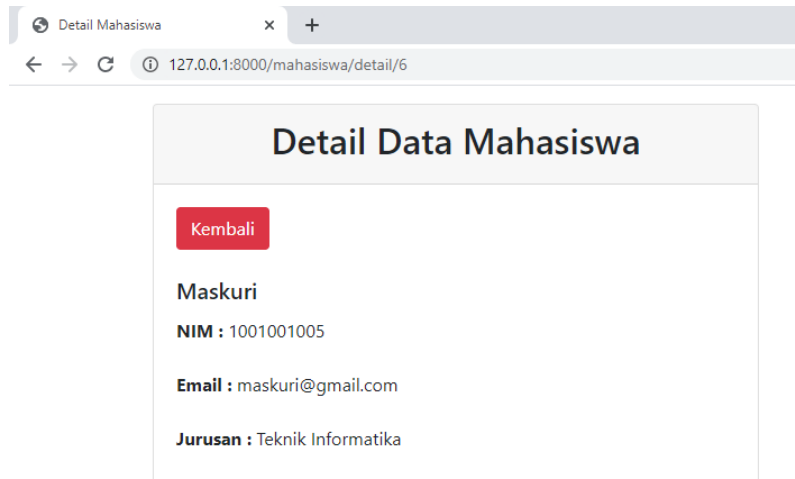
#### d. Melihat Detail Data dari Database

No	Langkah Percobaan
1	<p>Buatlah tombol untuk melihat detail data (Line 28) pada file index.blade.php di folder resources/views</p> <pre>&lt;tbody&gt;   @foreach(\$mahasiswa as \$mhs)     &lt;tr&gt;       &lt;td&gt;{{ \$mhs-&gt;nama }}&lt;/td&gt;       &lt;td&gt;{{ \$mhs-&gt;nim }}&lt;/td&gt;       &lt;td&gt;         &lt;a href="/mahasiswa/detail/{{ \$mhs-&gt;id }}" class="badge badge-info"&gt;Detail&lt;/a&gt;       &lt;/td&gt;     &lt;/tr&gt;   @endforeach &lt;/tbody&gt;</pre>
2	<p>Buatlah route baru pada routes/web.php dengan nama /mahasiswa/detail yang akan menjalankan fungsi detail pada MahasiswaController</p> <pre>Route::get('/mahasiswa/detail/{id}', 'MahasiswaController@detail');</pre>
3	<p>Buat method detail pada MahasiswaController.php di folder app/Http/Controllers yang akan menampilkan data mahasiswa pada view detail.blade.php.</p> <pre>14 15 //Kirim data Mhs ke view index 16 return view('index', ['mahasiswa' =&gt; \$mahasiswa]); 17 18 19 public function tambah() 20 21 //panggil view tambah 22 return view('tambah'); 23 24 25 public function simpan(Request \$request) 26 27 //tambah data ke table mhs 28 DB::table('mahasiswa')-&gt;insert([ 29     'nama' =&gt; \$request-&gt;namamhs, 30     'nim' =&gt; \$request-&gt;nimmhs, 31     'email' =&gt; \$request-&gt;emailmhs, 32     'jurusan' =&gt; \$request-&gt;jurusanmhs 33 ]); 34 return redirect('/mahasiswa'); 35 36 37 38 public function detail(\$id) 39 40 //mengambil data mhs brdsr id 41 \$mahasiswa = DB::table('mahasiswa')-&gt;where('id', \$id)-&gt;get(); 42 //kirim data mhs yg diambil ke edit.blade.php 43 return view('detail', ['mahasiswa' =&gt; \$mahasiswa]); 44 45 46</pre>

- 4 Kemudian buatlah view detail.blade.php yang menampilkan detail data mahasiswa pada folder resources/views. View detail juga mengaplikasikan master.blade.php.

```
resources > views > detail.blade.php
1
2 @extends('master')
3
4 @section('title','Detail Mahasiswa')
5
6 @section('judul_halaman','Detail Data Mahasiswa')
7
8 @section('konten')
9     <a href="/" class="btn btn-danger">Kembali</a>
10     <br/>
11     <br/>
12
13     @foreach($mahasiswa as $mhs)
14         <h5 class="card-title">{{ $mhs->nama }}</h5>
15         <p class="card-text">
16             <label for=""><b>NIM : </b></label>
17             {{ $mhs->nim }}
18         </p>
19         <p class="card-text">
20             <label for=""><b>Email : </b></label>
21             {{ $mhs->email }}
22         </p>
23         <p class="card-text">
24             <label for=""><b>Jurusan : </b></label>
25             {{ $mhs->jurusan }}
26         </p>
27     @endforeach
28 @endsection
```

- 5 Jalankan localhost:8000 dan pilih tombol 'Detail' di suatu data yang ingin kita lihat detailnya.



### e. Mengubah Data (Update) dari Database

No	Langkah Percobaan
1	<p>Buatlah tombol untuk mengubah data (Line 29) pada file index.blade.php di folder resources/views</p> <pre data-bbox="289 346 1409 672"> @foreach(\$mahasiswa as \$mhs) &lt;tr&gt; &lt;td&gt;{{ \$mhs-&gt;nama }}&lt;/td&gt; &lt;td&gt;{{ \$mhs-&gt;nim }}&lt;/td&gt; &lt;td&gt; &lt;a href="/mahasiswa/detail/{{ \$mhs-&gt;id }}" class="badge badge-info"&gt;Detail&lt;/a&gt; &lt;a href="/mahasiswa/edit/{{ \$mhs-&gt;id }}" class="badge badge-warning"&gt;Edit&lt;/a&gt; &lt;/td&gt; &lt;/tr&gt; &lt;/foreach&gt; </pre>
2	<p>Buatlah route baru pada routes/web.php dengan nama /mahasiswa/edit yang akan menjalankan fungsi edit pada MahasiswaController</p> <pre data-bbox="337 745 1312 804"> Route::get('/mahasiswa/edit/{id}', 'MahasiswaController@edit'); </pre>
3	<p>Buat method edit pada MahasiswaController.php di folder app/Http/Controllers yang akan menampilkan view edit.</p> <pre data-bbox="397 882 1214 1843"> app &gt; Http &gt; Controllers &gt; MahasiswaController.php 22     return view('tambah'); 23 24 25     public function simpan(Request \$request) 26     { 27         //tambah data ke table mhs 28         DB::table('mahasiswa')-&gt;insert([ 29             'nama' =&gt; \$request-&gt;namamhs, 30             'nim' =&gt; \$request-&gt;nimmhs, 31             'email' =&gt; \$request-&gt;emailmhs, 32             'jurusan' =&gt; \$request-&gt;jurusanmhs 33         ]); 34         return redirect('/mahasiswa'); 35     } 36 37 38     public function detail(\$id) 39     { 40         //mengambil data mhs brdsr id 41         \$mahasiswa = DB::table('mahasiswa')-&gt;where('id',\$id)-&gt;get() 42         //kirim data mhs yg diambil ke edit.blade.php 43         return view('detail',['mahasiswa' =&gt; \$mahasiswa]); 44     } 45 46     public function edit(\$id) 47     { 48         //ambil data mhs brdsr id 49         \$mahasiswa = DB::table('mahasiswa')-&gt;where('id',\$id)-&gt;get() 50         //kirim data mhs ke view edit 51         return view('edit',['mahasiswa' =&gt; \$mahasiswa]); 52     } 53 54 </pre>

4	<p>Kemudian buatlah view edit.blade.php yang berisi form untuk mengubah data pada folder resources/views. View edit juga mengaplikasikan master.blade.php.</p>  <pre> 1  @extends('master') 2 3  @section('title','Edit Data') 4 5  @section('judul_halaman','Edit Data Mahasiswa') 6 7  @section('konten') 8      &lt;a href="/" class="btn btn-danger"&gt;Kembali&lt;/a&gt; 9      &lt;br&gt; 10     &lt;br&gt; 11     @foreach(\$mahasiswa as \$mhs) 12         &lt;form action="/mahasiswa/update" method="POST"&gt; 13             {{ csrf_field() }} 14             &lt;input type="hidden" name="id" value="{{ \$mhs-&gt;id }}"&gt; 15             &lt;div class="form-group"&gt; 16                 &lt;label&gt;Nama&lt;/label&gt; 17                 &lt;input type="text" class="form-control" value="{{ \$mhs-&gt;nama }}" required="required" name="namamhs"&gt;&lt;br&gt; 18             &lt;/div&gt; 19             &lt;div class="form-group"&gt; 20                 &lt;label&gt;NIM&lt;/label&gt; 21                 &lt;input type="number" class="form-control" value="{{ \$mhs-&gt;nim }}" required="required" name="nimmhs"&gt;&lt;br&gt; 22             &lt;/div&gt; 23             &lt;div class="form-group"&gt; 24                 &lt;label&gt;Email&lt;/label&gt; 25                 &lt;input type="email" class="form-control" value="{{ \$mhs-&gt;email }}" required="required" name="emailmhs"&gt;&lt;br&gt; 26             &lt;/div&gt; 27             &lt;div class="form-group"&gt; 28                 &lt;label&gt;Jurusan&lt;/label&gt; 29                 &lt;input type="text" class="form-control" value="{{ \$mhs-&gt;jurusan }}" required="required" name="jurusanmhs"&gt;&lt;br&gt; 30             &lt;/div&gt; 31             &lt;input type="submit" class="btn btn-primary float-right" value="Update Data"&gt; 32         &lt;/form&gt; 33     @endforeach 34 @endsection </pre>
5	<p>Ketika tombol simpan ditekan, akan dipanggil routes /mahasiswa/update. Oleh karena itu, kita buat terlebih dahulu route tersebut.</p> <pre>Route::post('/mahasiswa/update','MahasiswaController@update');</pre>
6	<p>Buat method update pada MahasiswaController untuk menyimpan data yang diubah ke database.</p>  <pre> 45 46 public function edit(\$id) 47 48     //ambil data mhs brdsr id 49     \$mahasiswa = DB::table('mahasiswa')-&gt;where('id',\$id)-&gt;get(); 50     //kirim data mhs ke view edit 51     return view('edit',['mahasiswa' =&gt; \$mahasiswa]); 52 53 54 public function update(Request \$request) 55 56     //tambah data ke table mhs 57     DB::table('mahasiswa')-&gt;where('id',\$request-&gt;id)-&gt;update([ 58         'nama' =&gt; \$request-&gt;namamhs, 59         'nim' =&gt; \$request-&gt;nimmhs, 60         'email' =&gt; \$request-&gt;emailmhs, 61         'jurusan' =&gt; \$request-&gt;jurusanmhs 62     ]); 63     return redirect('/mahasiswa'); 64 65 66 </pre>
7	<p>Jalankan localhost:8000 dan pilih tombol 'Edit', maka akan ditampilkan halaman edit yang berisi form untuk mengubah data baru.</p>

Home x +

← → ↻ ⓘ 127.0.0.1:8000

## Data Mahasiswa

Tambah Data Mahasiswa

Nama	NIM	Aksi
nanda	1001001001	<a href="#">Detail</a> <a href="#">Edit</a>
wahyu	1001001002	<a href="#">Detail</a> <a href="#">Edit</a>
Rahmad	1001001003	<a href="#">Detail</a> <a href="#">Edit</a>
Alfian	1001001004	<a href="#">Detail</a> <a href="#">Edit</a>
Maskuri	1001001005	<a href="#">Detail</a> <a href="#">Edit</a>

Klik edit di data pertama, kita akan mengubah namanya.

Edit Data x +

← → ↻ ⓘ 127.0.0.1:8000/mahasiswa/edit/1

## Edit Data Mahasiswa

[Kembali](#)

Nama

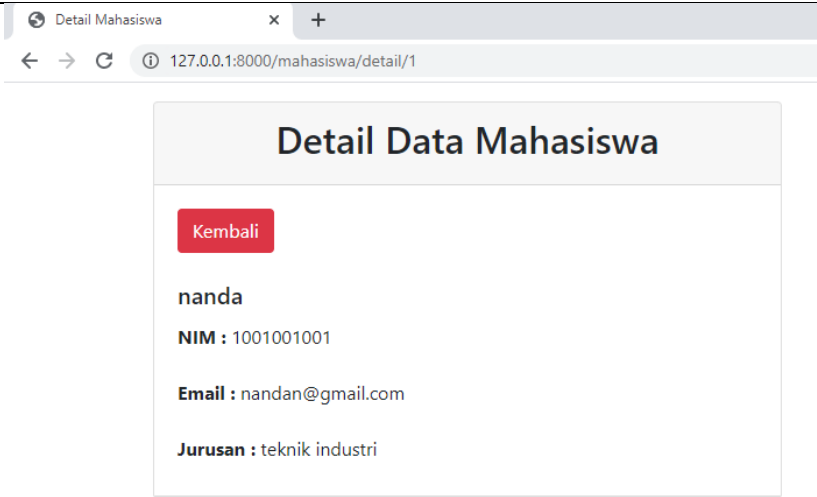
NIM

Email

Jurusan

[Update Data](#)

Setelah kita klik simpan Data, maka data pertama akan berubah.

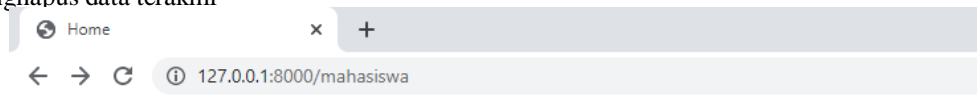
	 <p>Keterangan : Jurusan berubah menjadi teknik industri</p>
--	--

#### f. Menghapus Data (Delete) dari Database

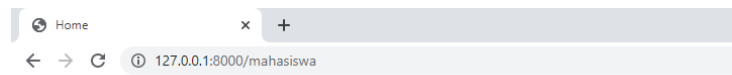
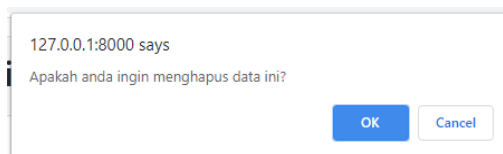
No	Langkah Percobaan			
1	<p>Buatlah tombol untuk menghapus data (Line 30) pada file index.blade.php di folder resources/views</p> <pre>foreach(\$mahasiswa as \$mhs) <tr> <td>&lt;td&gt;{{ \$mhs-&gt;nama }}&lt;/td&gt; <td>&lt;td&gt;{{ \$mhs-&gt;nim }}&lt;/td&gt; <td> &lt;a href="/mahasiswa/detail/{{ \$mhs-&gt;id }}" class="badge badge-info"&gt;Detail&lt;/a&gt; &lt;a href="/mahasiswa/edit/{{ \$mhs-&gt;id }}" class="badge badge-warning"&gt;Edit&lt;/a&gt; &lt;a href="/mahasiswa/hapus/{{ \$mhs-&gt;id }}" class="badge badge-danger" onclick="return confirm('Apakah anda ingin menghapus data ini?')"&gt;Hapus&lt;/a&gt; </td> </td></td></tr> </pre>	<td>{{ \$mhs->nama }}</td> <td>&lt;td&gt;{{ \$mhs-&gt;nim }}&lt;/td&gt; <td> &lt;a href="/mahasiswa/detail/{{ \$mhs-&gt;id }}" class="badge badge-info"&gt;Detail&lt;/a&gt; &lt;a href="/mahasiswa/edit/{{ \$mhs-&gt;id }}" class="badge badge-warning"&gt;Edit&lt;/a&gt; &lt;a href="/mahasiswa/hapus/{{ \$mhs-&gt;id }}" class="badge badge-danger" onclick="return confirm('Apakah anda ingin menghapus data ini?')"&gt;Hapus&lt;/a&gt; </td> </td>	<td>{{ \$mhs->nim }}</td> <td> &lt;a href="/mahasiswa/detail/{{ \$mhs-&gt;id }}" class="badge badge-info"&gt;Detail&lt;/a&gt; &lt;a href="/mahasiswa/edit/{{ \$mhs-&gt;id }}" class="badge badge-warning"&gt;Edit&lt;/a&gt; &lt;a href="/mahasiswa/hapus/{{ \$mhs-&gt;id }}" class="badge badge-danger" onclick="return confirm('Apakah anda ingin menghapus data ini?')"&gt;Hapus&lt;/a&gt; </td>	<a href="/mahasiswa/detail/{{ \$mhs->id }}" class="badge badge-info">Detail</a> <a href="/mahasiswa/edit/{{ \$mhs->id }}" class="badge badge-warning">Edit</a> <a href="/mahasiswa/hapus/{{ \$mhs->id }}" class="badge badge-danger" onclick="return confirm('Apakah anda ingin menghapus data ini?')">Hapus</a>
<td>{{ \$mhs->nama }}</td> <td>&lt;td&gt;{{ \$mhs-&gt;nim }}&lt;/td&gt; <td> &lt;a href="/mahasiswa/detail/{{ \$mhs-&gt;id }}" class="badge badge-info"&gt;Detail&lt;/a&gt; &lt;a href="/mahasiswa/edit/{{ \$mhs-&gt;id }}" class="badge badge-warning"&gt;Edit&lt;/a&gt; &lt;a href="/mahasiswa/hapus/{{ \$mhs-&gt;id }}" class="badge badge-danger" onclick="return confirm('Apakah anda ingin menghapus data ini?')"&gt;Hapus&lt;/a&gt; </td> </td>	<td>{{ \$mhs->nim }}</td> <td> &lt;a href="/mahasiswa/detail/{{ \$mhs-&gt;id }}" class="badge badge-info"&gt;Detail&lt;/a&gt; &lt;a href="/mahasiswa/edit/{{ \$mhs-&gt;id }}" class="badge badge-warning"&gt;Edit&lt;/a&gt; &lt;a href="/mahasiswa/hapus/{{ \$mhs-&gt;id }}" class="badge badge-danger" onclick="return confirm('Apakah anda ingin menghapus data ini?')"&gt;Hapus&lt;/a&gt; </td>	<a href="/mahasiswa/detail/{{ \$mhs->id }}" class="badge badge-info">Detail</a> <a href="/mahasiswa/edit/{{ \$mhs->id }}" class="badge badge-warning">Edit</a> <a href="/mahasiswa/hapus/{{ \$mhs->id }}" class="badge badge-danger" onclick="return confirm('Apakah anda ingin menghapus data ini?')">Hapus</a>		
2	<p>Buatlah route baru pada routes/web.php dengan nama /mahasiswa/hapus yang akan menjalankan fungsi hapus pada MahasiswaController</p> <pre>Route::get('/mahasiswa/hapus/{id}', 'MahasiswaController@hapus');</pre>			
3	<p>Buat method hapus pada MahasiswaController.php di folder app/Http/Controllers yang akan menjalankan fungsi hapus.</p> <pre>54 public function update(Request \$request) 55 { 56     //tambah data ke table mhs 57     DB::table('mahasiswa')-&gt;where('id',\$request-&gt;id)-&gt;update([ 58         'nama' =&gt; \$request-&gt;namamhs, 59         'nim' =&gt; \$request-&gt;nimmhs, 60         'email' =&gt; \$request-&gt;emailmhs, 61         'jurusan' =&gt; \$request-&gt;jurusanmhs 62     ]); 63     return redirect('/mahasiswa'); 64 } 65 66 public function hapus(\$id) 67 { 68     //ambil data mhs brdsr id 69     \$mahasiswa = DB::table('mahasiswa')-&gt;where('id',\$id)-&gt;delete; 70     //kirim data mhs ke view edit 71     return redirect('/mahasiswa'); 72 } 73 74</pre>			

4

Jalankan localhost:8000 dan pilih tombol 'Hapus', maka data yang terpilih akan dihapus. Contoh: menghapus data terakhir



Data Mahasiswa		
Tambah Data Mahasiswa		
Nama	NIM	Aksi
nanda	1001001001	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
wahyu	1001001002	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
Rahmad	1001001003	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
Alfian	1001001004	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
Maskuri	1001001005	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>

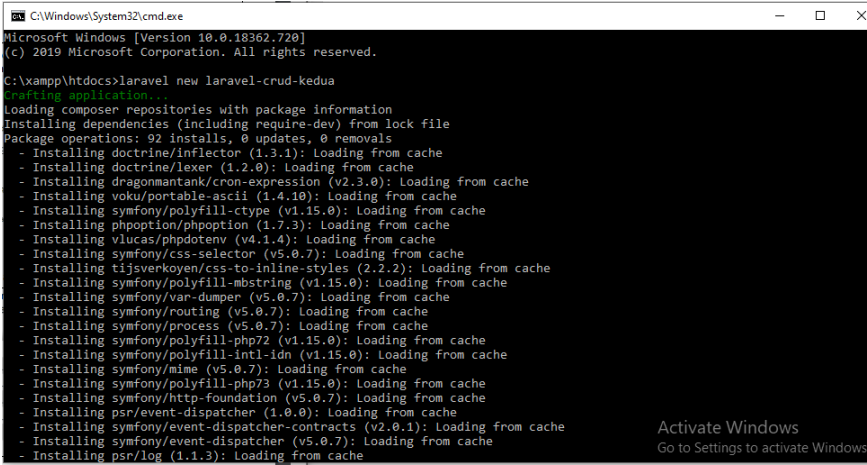
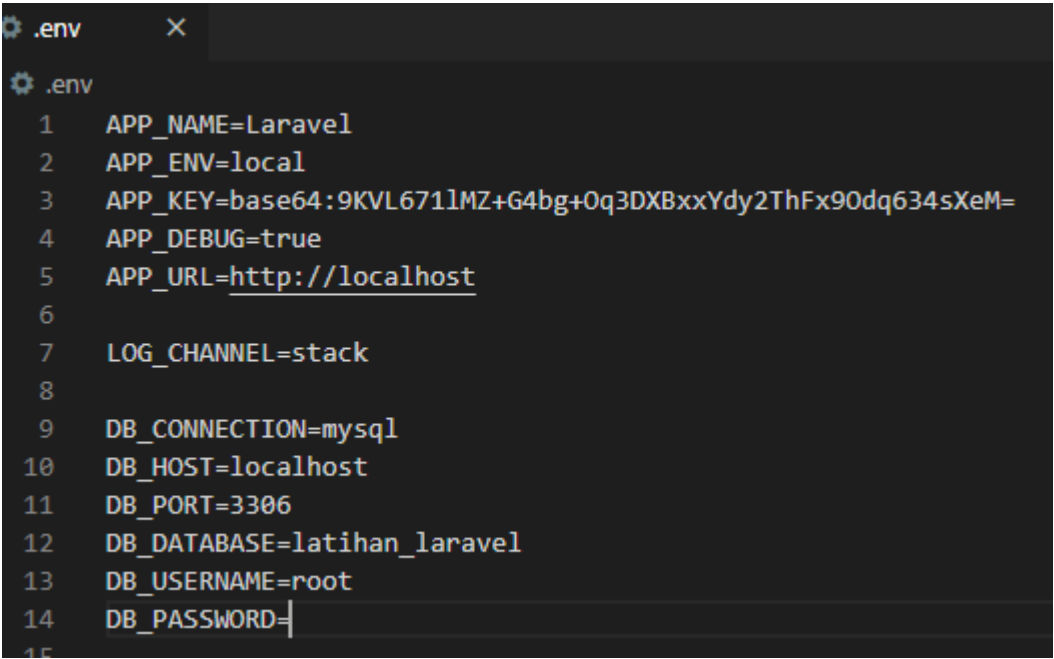


Data Mahasiswa		
Tambah Data Mahasiswa		
Nama	NIM	Aksi
nanda	1001001001	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
wahyu	1001001002	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
Rahmad	1001001003	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
Alfian	1001001004	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>



Praktikum – Bagian 2: Membuat CRUD di Laravel menggunakan Eloquent Eloquent adalah sebuah fitur untuk mengelola data yang ada pada database. Eloquent ORM menyediakan fungsi-fungsi active record, atau fungsi-fungsi query sql untuk mengelola data pada database. Dengan menggunakan Eloquent pada Laravel, kita bisa mengelola data pada database dari hanya satu buah model.

#### a. Konfigurasi Database

No	Langkah Percobaan
1	<p>Buatlah project Laravel baru dengan nama laravel-crud-kedua. Buka command prompt, tuliskan perintah berikut.</p> 
2	<p>Selanjutnya kita lakukan konfigurasi database di Laravel. Untuk melakukan konfigurasi database, bukalah file .env pada project laravel-crud. Ubah seperti di bawah ini.</p> 
3	<p>Pada project ini kita gunakan database dan tabel yang sebelumnya digunakan pada Praktikum Bagian 1. Tambahkan kolom created_at dan updated_at yang bertipe TIMESTAMP dan default nilainya NULL</p>

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	id	int(11)			No	None		AUTO_INCREMENT	Change  Drop  More
<input type="checkbox"/> 2	nama	varchar(100)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/> 3	nim	varchar(10)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/> 4	email	varchar(50)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/> 5	jurusan	varchar(20)	latin1_swedish_ci		No	None			Change  Drop  More
<input type="checkbox"/> 6	created_at	timestamp			Yes	NULL			Change  Drop  More
<input type="checkbox"/> 7	updated_at	timestamp			Yes	NULL			Change  Drop  More

## b. Menampilkan Data dari Database

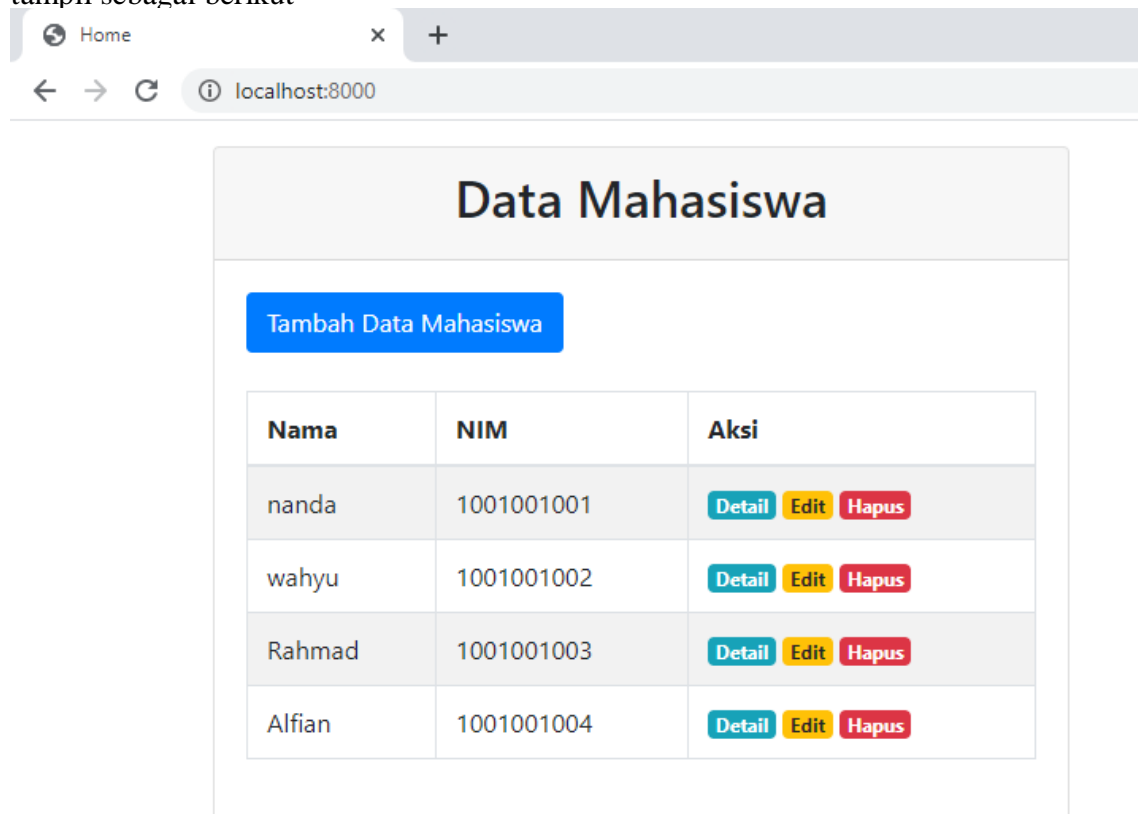
No	Langkah Percobaan
1	<p>Setelah kita memiliki beberapa data pada tabel mahasiswa, kita akan mencoba untuk menampilkan data tersebut ketika project dijalankan. Pertama, buatlah route pada routes/web.php sehingga ketika pertama kali project dijalankan akan terbuka halaman yang menampilkan data.</p> <pre> 19 20 Route::get('/', 'MahasiswaController@index'); 21 Route::get('/mahasiswa', 'MahasiswaController@index'); 22 </pre>
2	<p>Buat model menggunakan command prompt dengan nama Mahasiswa menggunakan php artisan</p> <pre> C:\xampp\htdocs&gt;cd laravel-crud-kedua C:\xampp\htdocs\laravel-crud-kedua&gt;php artisan make:model Mahasiswa Model created successfully. </pre>
3	<p>Ubah model Mahasiswa.php pada folder App menjadi seperti berikut.</p> <pre> app &gt; Mahasiswa.php 1  &lt;?php 2 3  namespace App; 4 5  use Illuminate\Database\Eloquent\Model; 6 7  class Mahasiswa extends Model 8  { 9      protected \$table = "mahasiswa"; 10 } 11 </pre>
4	<p>Buat controller baru yaitu MahasiswaController menggunakan php artisan php artisan make:controller MahasiswaController</p> <pre> C:\xampp\htdocs\laravel-crud-kedua&gt;php artisan make:controller MahasiswaController Controller created successfully. </pre>
5	<p>Buat method index pada MahasiswaController.php pada folder app/Http/Controllers</p>

	<pre> app &gt; Http &gt; Controllers &gt; 🐞 MahasiswaController.php 1  &lt;?php 2 3  namespace App\Http\Controllers; 4 5  use Illuminate\Http\Request; 6  use App\Mahasiswa; 7  class MahasiswaController extends Controller 8  { 9      public function index() 10     { 11         \$mahasiswa = Mahasiswa::all(); 12         return view('index',['mahasiswa' =&gt; \$mahasiswa]); 13     } 14 } 15 </pre>
6	<p>Selanjutnya kita akan membuat view untuk menampilkan data mahasiswa dengan nama index.blade.php. Tetapi agar pembuatan view selanjutnya menjadi lebih mudah, terlebih dahulu kita akan membuat template blade (seperti pada Bagian 1). Pada bagian view kita buat seperti bagian 1 (copy-paste dari bagian 1)</p> <pre> resources &gt; views &gt; 🐞 master.blade.php 1  &lt;html&gt; 2 3  &lt;head&gt; 4      &lt;meta charset="utf-8"&gt; 5      &lt;meta name="viewport" content="width=device-width, initial 6      &lt;link rel="stylesheet" href="https://stackpath.bootstrapcdn.c 7      &lt;title&gt; @yield('title') &lt;/title&gt; 8  &lt;/head&gt; 9 10 &lt;body&gt; 11     &lt;div class="container"&gt; 12         &lt;div class="row mt-3"&gt; 13             &lt;div class="col-md-6"&gt; 14                 &lt;div class="card"&gt; 15                     &lt;div class="card-header text-center"&gt; 16                         &lt;!-- Judul Halaman --&gt; 17                         &lt;h2&gt; @yield('judul_halaman') &lt;/h2&gt; 18                     &lt;/div&gt; 19                     &lt;div class="card-body"&gt; 20                         &lt;!-- Konten Halaman --&gt; 21                         @yield('konten') 22                     &lt;/div&gt; 23                 &lt;/div&gt; 24             &lt;/div&gt; 25         &lt;/div&gt; 26     &lt;/div&gt; 27 &lt;/body&gt; 28 29 &lt;/html&gt; </pre>

```
resources > views > index.blade.php
1 @extends('master')
2
3 @section('title','Home')
4
5 @section('judul_halaman','Data Mahasiswa')
6 @section('konten')
7     <a href="/mahasiswa/tambah" class="btn btn-primary">Tambah Data Mahasiswa</a>
8     <br/>
9     <br/>
10    <table class="table table-bordered table-hover table-striped">
11        <thead>
12            <th>Nama</th>
13            <th>NIM</th>
14            <th>Aksi</th>
15        </thead>
16        <tbody>
17            @foreach($mahasiswa as $mhs)
18                <tr>
19                    <td>{{ $mhs->nama }}</td>
20                    <td>{{ $mhs->nim }}</td>
21                    <td>
22                        <a href="/mahasiswa/detail/{{ $mhs->id }}" class="badge badge-info">Detail</a>
23                        <a href="/mahasiswa/edit/{{ $mhs->id }}" class="badge badge-warning">Edit</a>
24                        <a href="/mahasiswa/hapus/{{ $mhs->id }}" class="badge badge-danger" onclick="return confirm('Apakah anda ingin menghapus data i
25                    </td>
26                </tr>
27            @endforeach
28        </tbody>
29    </table>
30 @endsection
```

7

Jalankan command prompt, tuliskan perintah untuk menjalankan project laravel-  
crud php artisan serve Buka browser dan ketikkan localhost:8000, maka akan  
tampil sebagai berikut



Home x +

localhost:8000

## Data Mahasiswa

Tambah Data Mahasiswa

Nama	NIM	Aksi
nanda	1001001001	<a href="#" style="background-color: #17a2b8; color: white; padding: 2px 5px;">Detail</a> <a href="#" style="background-color: #ffc107; color: black; padding: 2px 5px;">Edit</a> <a href="#" style="background-color: #dc3545; color: white; padding: 2px 5px;">Hapus</a>
wahyu	1001001002	<a href="#" style="background-color: #17a2b8; color: white; padding: 2px 5px;">Detail</a> <a href="#" style="background-color: #ffc107; color: black; padding: 2px 5px;">Edit</a> <a href="#" style="background-color: #dc3545; color: white; padding: 2px 5px;">Hapus</a>
Rahmad	1001001003	<a href="#" style="background-color: #17a2b8; color: white; padding: 2px 5px;">Detail</a> <a href="#" style="background-color: #ffc107; color: black; padding: 2px 5px;">Edit</a> <a href="#" style="background-color: #dc3545; color: white; padding: 2px 5px;">Hapus</a>
Alfian	1001001004	<a href="#" style="background-color: #17a2b8; color: white; padding: 2px 5px;">Detail</a> <a href="#" style="background-color: #ffc107; color: black; padding: 2px 5px;">Edit</a> <a href="#" style="background-color: #dc3545; color: white; padding: 2px 5px;">Hapus</a>

**c. Memasukkan Data (Create) ke Database**

No	Langkah Percobaan
1	<p>Buatlah route baru pada routes/web.php dengan nama /mahasiswa/tambah yang akan menjalankan fungsi tambah pada MahasiswaController ketika tombol Tambah Data Mahasiswa ditekan.</p> <pre>Route::get('/mahasiswa/tambah','MahasiswaController@tambah');</pre>
2	<p>Buat method tambah pada MahasiswaController.php di folder app/Http/Controllers yang akan menampilkan view tambah.</p> <pre>app &gt; Http &gt; Controllers &gt; MahasiswaController.php 1  &lt;?php 2 3  namespace App\Http\Controllers; 4 5  use Illuminate\Http\Request; 6  use App\Mahasiswa; 7  class MahasiswaController extends Controller 8  { 9      public function index() 10     { 11         \$mahasiswa = Mahasiswa::all(); 12         return view('index',['mahasiswa' =&gt; \$mahasiswa]); 13     } 14     public function tambah() 15     { 16         return view('tambah'); 17     } 18 } 19</pre>
3	<p>Kemudian buatlah view tambah.blade.php yang berisi form untuk memasukkan data baru pada folder resources/views. Kita lakukan copy paste dari tambah.blade.php di Bagian 1</p>



	<pre> 13     } 14     public function tambah() 15     { 16         return view('tambah'); 17     } 18     public function simpan(Request \$request) 19     { 20         //tambah data ke table mhs 21         Mahasiswa::create([ 22             'nama' =&gt; \$request-&gt;namamhs, 23             'nim' =&gt; \$request-&gt;nimmhs, 24             'email' =&gt; \$request-&gt;emailmhs, 25             'jurusan' =&gt; \$request-&gt;jurusanmhs 26         ]); 27         return redirect('/mahasiswa'); 28     } 29 } 30 } 31 </pre>
6	<p>Karena kita menggunakan fungsi create pada Controller, maka butuh ditambahkan code pada Line 10 yang disebut Mass Assignment pada model Mahasiswa.php. Mass Assignment digunakan untuk memfilter kolom mana yang boleh dan tidak boleh diinput.</p> <pre> app &gt; Mahasiswa.php 1  &lt;?php 2 3  namespace App; 4 5  use Illuminate\Database\Eloquent\Model; 6 7  class Mahasiswa extends Model 8  { 9      protected \$table = "mahasiswa"; 10     protected \$fillable = ['nama', 'nim', 'email', 'jurusan']; 11 } 12 </pre>
7	<p>Kembali coba jalankan localhost:8000 dan pilih tombol 'Tambah Data Mahasiswa', maka akan ditampilkan halaman tambah yang berisi form untuk memasukkan data baru. Kita coba isikan data pada form tersebut.</p>

## Tambah Data Mahasiswa

Kembali

Nama

Aditya

NIM

1001001006

Email

adit@gmail.com

Jurusan

Teknik Mesin

Simpan Data

## Data Mahasiswa

Tambah Data Mahasiswa

Nama	NIM	Aksi
nanda	1001001001	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
wahyu	1001001002	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
Rahmad	1001001003	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
Alfian	1001001004	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>
Aditya	1001001006	<a href="#">Detail</a> <a href="#">Edit</a> <a href="#">Hapus</a>

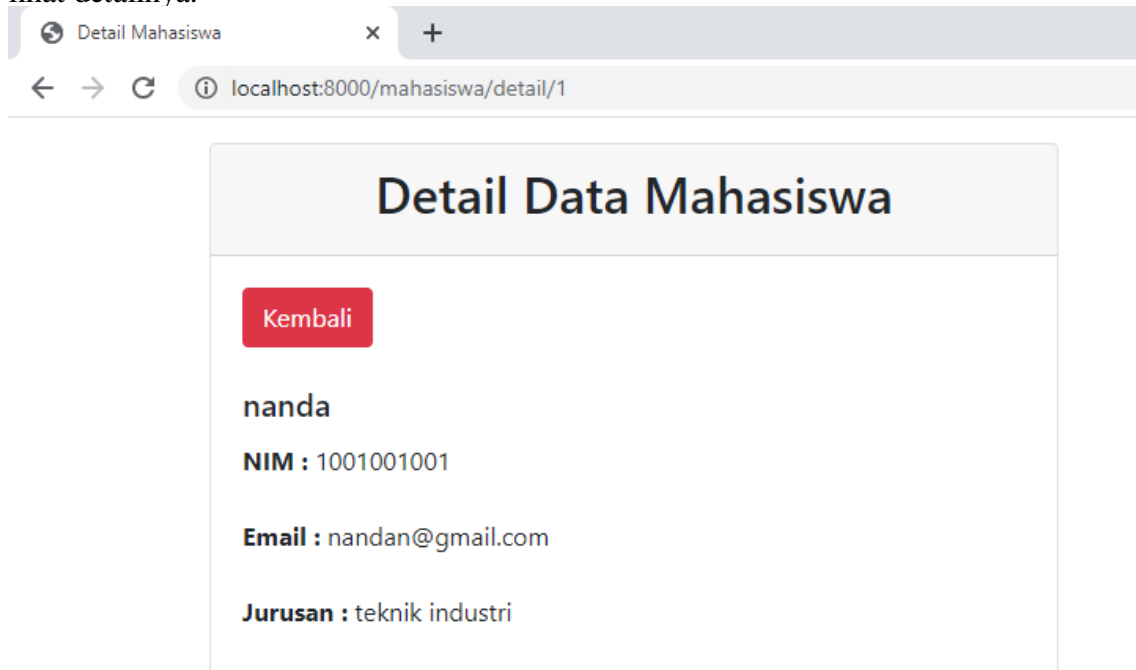


**d. Melihat Detail Data dari Database**

No	Langkah Percobaan
1	<p>Buatlah route baru pada routes/web.php dengan nama /mahasiswa/detail yang akan menjalankan fungsi detail pada MahasiswaController</p> <pre>Route::post('/mahasiswa/simpan', 'MahasiswaController@simpan'); Route::get('/mahasiswa/detail/{id}', 'MahasiswaController@detail');</pre>
2	<p>Buat method detail pada MahasiswaController.php di folder app/Http/Controllers yang akan menampilkan data mahasiswa pada view detail.blade.php.</p> <pre>3 public function simpan(Request \$request) 4 { 5     //tambah data ke table mhs 6     Mahasiswa::create([ 7         'nama' =&gt; \$request-&gt;namamhs, 8         'nim' =&gt; \$request-&gt;nimmhs, 9         'email' =&gt; \$request-&gt;emailmhs, 10        'jurusan' =&gt; \$request-&gt;jurusanmhs 11    ]); 12    return redirect('/mahasiswa'); 13 } 14 15 public function detail(\$id) 16 { 17     //mengambil data mhs brdsr id 18     \$mahasiswa = Mahasiswa::find(\$id); 19     //kirim data mhs yg diambil ke edit.blade.php 20     return view('detail', ['mahasiswa' =&gt; \$mahasiswa]); 21 }</pre>
3	<p>Kemudian buatlah view detail.blade.php yang menampilkan detail data mahasiswa pada folder resources/views. View detail juga mengaplikasikan master.blade.php.</p>

```
resources > views > detail.blade.php
1
2 @extends('master')
3
4 @section('title','Detail Mahasiswa')
5
6 @section('judul_halaman','Detail Data Mahasiswa')
7
8 @section('konten')
9     <a href="/" class="btn btn-danger">Kembali</a>
10     <br/>
11     <br/>
12
13     <h5 class="card-title">{{ $mahasiswa->nama }}</h5>
14     <p class="card-text">
15         <label for=""><b>NIM : </b></label>
16         {{ $mahasiswa->nim }}
17     </p>
18     <p class="card-text">
19         <label for=""><b>Email : </b></label>
20         {{ $mahasiswa->email }}
21     </p>
22     <p class="card-text">
23         <label for=""><b>Jurusan : </b></label>
24         {{ $mahasiswa->jurusan }}
25     </p>
26 @endsection
```

- 4 Jalankan localhost:8000 dan pilih tombol 'Detail' di suatu data yang ingin kita lihat detailnya.



**e. Mengubah Data (Update) dari Database**

No	Langkah Percobaan
1	Buatlah route baru pada routes/web.php dengan nama /mahasiswa/edit yang akan menjalankan fungsi edit pada MahasiswaController <pre>Route::get('/mahasiswa/edit/{id}', 'MahasiswaController@edit');</pre>
2	Buat method edit pada MahasiswaController.php di folder app/Http/Controllers yang akan menampilkan view edit. <pre>public function edit(\$id) {     //ambil data mhs brdsr id     \$mahasiswa = Mahasiswa::find(\$id);     //kirim data mhs ke view edit     return view('edit', ['mahasiswa' =&gt; \$mahasiswa]); }</pre>
3	Kemudian buatlah view edit.blade.php yang berisi form untuk mengubah data pada folder resources/views. View edit juga mengaplikasikan master.blade.php. <pre>resources &gt; views &gt; edit.blade.php 1 @extends('master') 2 3 @section('title', 'Edit Data') 4 5 @section('judul_halaman', 'Edit Data Mahasiswa') 6 7 @section('konten') 8 &lt;a href="/" class="btn btn-danger"&gt;Kembali&lt;/a&gt; 9 &lt;br&gt; 10 &lt;br&gt; 11 &lt;form action="/mahasiswa/update" method="POST"&gt; 12     {{ csrf_field() }} 13     &lt;input type="hidden" name="id" value="{{ \$mahasiswa-&gt;id }}"&gt; 14     &lt;div class="form-group"&gt; 15         &lt;label&gt;Nama&lt;/label&gt; 16         &lt;input type="text" class="form-control" value="{{ \$mahasiswa-&gt;nama }}" required="required" name="namamhs"&gt;&lt;br&gt; 17     &lt;/div&gt; 18     &lt;div class="form-group"&gt; 19         &lt;label&gt;NIM&lt;/label&gt; 20         &lt;input type="number" class="form-control" value="{{ \$mahasiswa-&gt;nim }}" required="required" name="nimhs"&gt;&lt;br&gt; 21     &lt;/div&gt; 22     &lt;div class="form-group"&gt; 23         &lt;label&gt;Email&lt;/label&gt; 24         &lt;input type="email" class="form-control" value="{{ \$mahasiswa-&gt;email }}" required="required" name="emailmhs"&gt;&lt;br&gt; 25     &lt;/div&gt; 26     &lt;div class="form-group"&gt; 27         &lt;label&gt;Jurusan&lt;/label&gt; 28         &lt;input type="text" class="form-control" value="{{ \$mahasiswa-&gt;jurusan }}" required="required" name="jurusanmhs"&gt;&lt;br&gt; 29     &lt;/div&gt; 30     &lt;input type="submit" class="btn btn-primary float-right" value="Update Data"&gt; 31 &lt;/form&gt; 32 @endsection</pre>
4	Ketika tombol simpan ditekan, akan dipanggil routes /mahasiswa/update/{id}. Oleh karena itu, kita buat terlebih dahulu route tersebut. <pre>Route::post('/mahasiswa/update', 'MahasiswaController@update');</pre>
5	Buat method update pada MahasiswaController untuk menyimpan data yang diubah ke database.

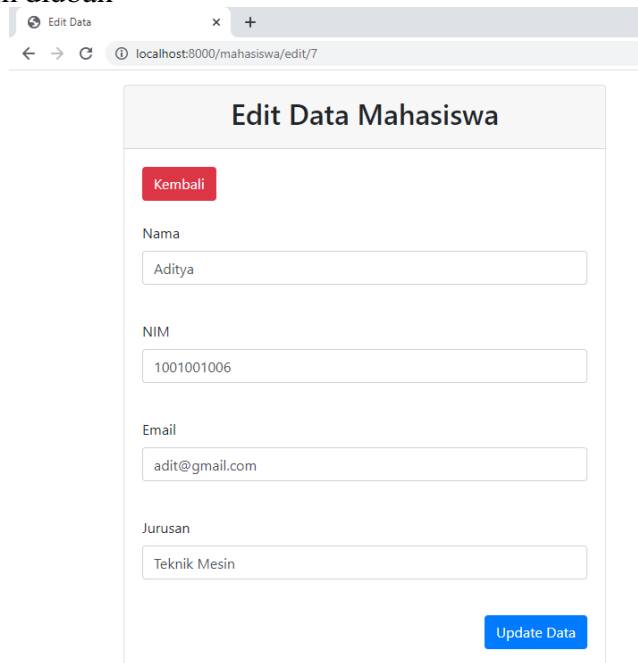
```

public function update(Request $request)
{
    //tambah data ke table mhs
    $mahasiswa = Mahasiswa::find($request->id);
    $mahasiswa->nama = $request->namamhs;
    $mahasiswa->nim = $request->nimmhs;
    $mahasiswa->email = $request->emailmhs;
    $mahasiswa->jurusan = $request->jurusanmhs;
    $mahasiswa->save();
    return redirect('/');
}

```

- 6 Jalankan localhost:8000 dan pilih tombol 'Edit', maka akan ditampilkan halaman edit yang berisi form untuk mengubah data baru. Cobalah untuk mengedit data.

- Sebelum diubah



The screenshot shows a web browser window with the address bar displaying 'localhost:8000/mahasiswa/edit/7'. The page title is 'Edit Data'. The main content area is titled 'Edit Data Mahasiswa' and contains a form with the following fields and values:

- Kembali** (button)
- Nama**: Aditya
- NIM**: 1001001006
- Email**: adit@gmail.com
- Jurusan**: Teknik Mesin
- Update Data** (button)

Edit Data

localhost:8000/mahasiswa/edit/7

### Edit Data Mahasiswa

Kembali

Nama

Maskuri

NIM

1001001005

Email

maskuri@gmail.com

Jurusan

Teknik Informatika

Update Data

- Setelah diubah

Home

localhost:8000

### Data Mahasiswa

Tambah Data Mahasiswa

Nama	NIM	Aksi
nanda	1001001001	<div>DetailEditHapus</div>
wahyu	1001001002	<div>DetailEditHapus</div>
Rahmad	1001001003	<div>DetailEditHapus</div>
Alfian	1001001004	<div>DetailEditHapus</div>
Maskuri	1001001005	<div>DetailEditHapus</div>

**f. Menghapus Data (Delete) dari Database**

No	Langkah Percobaan
1	<p>Buatlah route baru pada routes/web.php dengan nama /mahasiswa/hapus yang akan menjalankan fungsi hapus pada MahasiswaController</p> <pre>Route::get('/mahasiswa/hapus/{id}', 'MahasiswaController@hapus');</pre>
2	<p>Buat method hapus pada MahasiswaController.php di folder app/Http/Controllers yang akan menjalankan fungsi hapus.</p> <pre>public function hapus(\$id) {     //ambil data mhs brdsr id     \$mahasiswa = Mahasiswa::find(\$id);     \$mahasiswa-&gt;delete();     //kirim data mhs ke view edit     return redirect('/mahasiswa'); }</pre>
3	<p>Jalankan localhost:8000 dan pilih tombol 'Hapus', maka data yang terpilih akan dihapus.</p> 