

CRUD

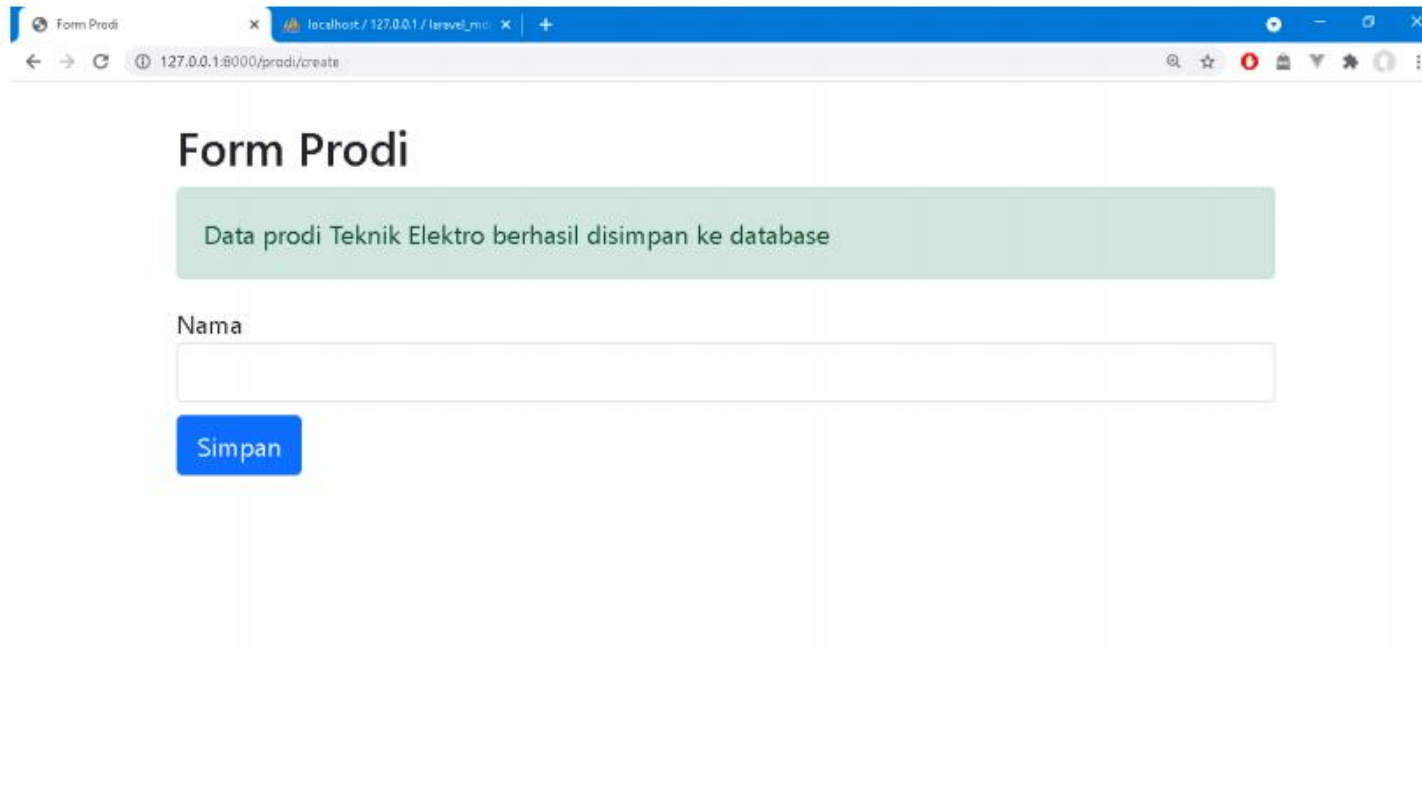
Universitas Multi Data Palembang

Pengertian

- CRUD singkatan dari Create, Read, Update dan Delete
- Melalui CRUD kita dapat mengelola data yang ada di database

Create

- Proses memasukkan data ke database
- Pada pertemuan sebelumnya, kita sudah membuat bagian Create menggunakan Form Prodi



The screenshot shows a web browser window with the address bar displaying 'localhost/127.0.0.1/1ersvel_mdi'. The page title is 'Form Prodi'. The URL in the address bar is '127.0.0.1:8000/prodi/create'. The page content includes a green message box stating 'Data prodi Teknik Elektro berhasil disimpan ke database'. Below this message is a text input field labeled 'Nama' and a blue button labeled 'Simpan'.

Read

- Proses membaca data yang ada di database
- Terdapat 2 route untuk proses read:

Route menampilkan semua data program studi:

```
Route::get('/prodi', [Prodi Controller::class,  
'index'])->name('prodi.index');
```

Route menampilkan satu data (detail) program studi:

```
Route::get('/prodi/{id}', [Prodi Controller::class,  
'show'])->name('prodi.show');
```

Read

(Menampilkan semua data program studi)

- Proses menampilkan semua data program studi yang ada di tabel prodi dilakukan oleh route get /prodi
- Tambahkan route di bawah ini pada file routes/web.php
`Route::get('/prodi', [ProdiController::class, 'index']);`
- Tambahkan atau ubah (jika sudah ada) method index() pada ProdiController.php

```
public function index()
{
    $prodis = Prodi::all();
    return view('prodi.index')->with('prodis', $prodis);
}
```

Read

(Menampilkan semua data program studi)

- `$prodis = Prodi::all();` merupakan perintah eloquent untuk mengambil semua data yang ada di tabel prodi.
- Variabel `$prodis` akan berisi collection dari semua data yang ada di tabel prodi.
- Selanjutnya variabel `$prodis` dikirim ke view `prodi.index`

Read

(Menampilkan semua data program studi)

- Berikut isi dari file resources/views/prodi/index.blade.php

```
web.php X ProdiController.php index.blade.php X create.blade.php master.blade.php
lara_mi4a > resources > views > prodi > index.blade.php > div.row.pt-4 > div.col > table.table.table-striped.table-hover > tbody
1  @extends('layout.master')
2  @section('title', 'Halaman List Prodi')
3
4  @section('content')
5      <div class="row pt-4">
6          <div class="col">
7              <h2>Prodi</h2>
8              <div class="d-md-flex justify-content-md-end">
9                  <a href="{{ route('prodi.create') }}" class="btn btn-primary">Tambah</a>
10             </div>
11             <table class="table table-striped table-hover">
12                 <thead>
13                     <tr>
14                         <th>Nama</th><th>Aksi</th>
15                     </tr>
16                 </thead>
```

Read

(Menampilkan semua data program studi)

- Berikut isi dari file resources/views/prodi/index.blade.php (lanjutan)

```
web.php × ProdiController.php index.blade.php × create.blade.php master.blade.php
lara_mi4a > resources > views > prodi > index.blade.php > ...
17         <tbody>
18             @foreach ($prodis as $item)
19                 <tr>
20                     <td> {{ $item->nama }} </td>
21                     <td><a href="{{ url('/prodi/'.$item->id) }}" class="btn btn-warning">Detail</a></td>
22                 </tr>
23             @endforeach
24         </tbody>
25     </table>
26 </div>
27 </div>
28 @endsection
29
```


Read

(Menampilkan semua data program studi)

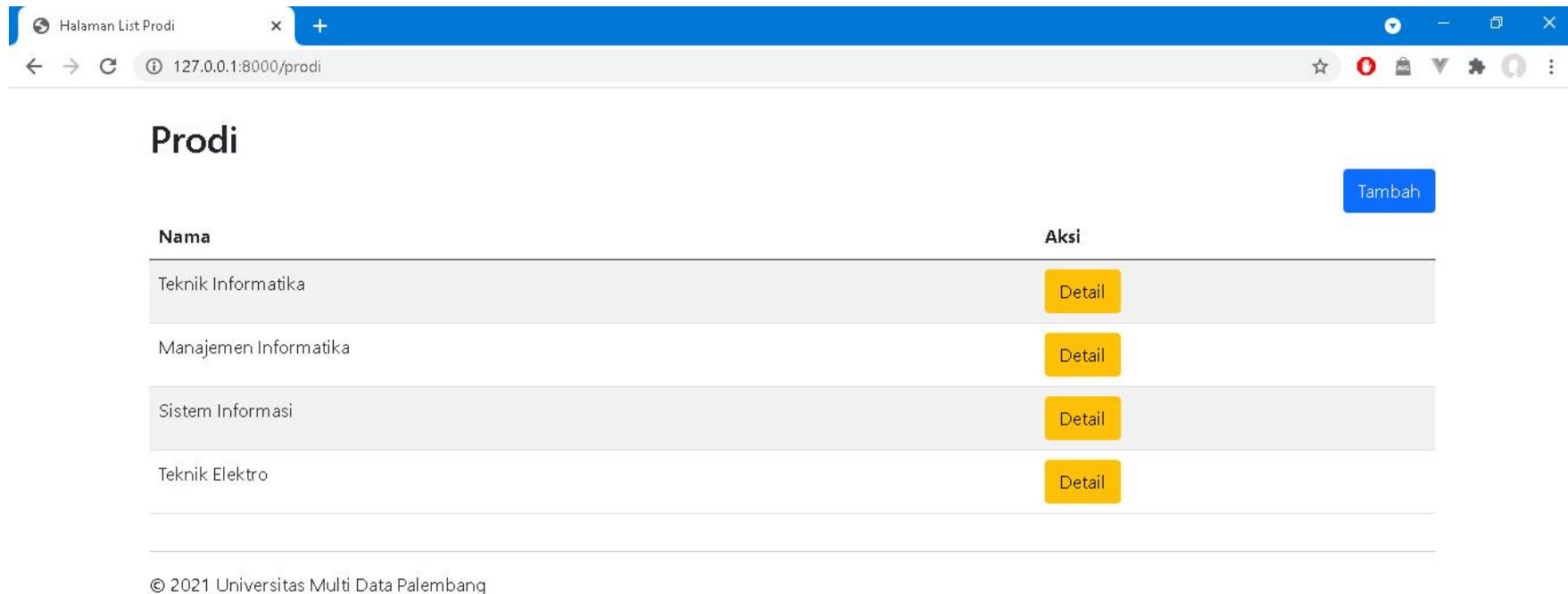
- Berikut isi dari file resources/views/layout/master.blade.php

```
web.php x ProdiController.php index.blade.php create.blade.php master.blade.php x
lara_mi4a > resources > views > layout > master.blade.php > ...
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <meta http-equiv="X-UA-Compatible" content="ie=edge">
7     <link rel="stylesheet" href="{{ asset('css/bootstrap.min.css') }}">
8     <title>@yield('title')</title>
9 </head>
10 <body>
11     <div class="container">
12         @yield('content')
13
14         <hr>
15         &copy; {{ date('Y') }} Universitas Multi Data Palembang
16     </div>
17 </body>
18 </html>
```

Read

(Menampilkan semua data program studi)

- Silakan buka 127.0.0.1/prodi, hasilnya seperti gambar di bawah ini



Read

(Menampilkan satu data program studi)

- Proses menampilkan satu data program studi yang ada di tabel prodi dilakukan oleh route get /prodi/{prodi}
- Tambahkan route di bawah ini pada file routes/web.php
`Route::get('/prodi/{prodi}', [ProdiController::class, 'show'])->name('prodi.show');`
- Tambahkan method show(Prodi \$prodi) pada ProdiController.php menggunakan route model binding, hasilnya parameter \$prodi langsung berisi object prodi sesuai id yang dikirim route

```
public function show(Prodi $prodi)
{
    return view('prodi.show', ['prodi' => $prodi]);
}
```

Read

(Menampilkan satu data program studi)

- Karena sudah berisi 1 object Prodi, maka dapat langsung dikirim ke view `prodi.show`

Read

(Menampilkan satu data program studi)

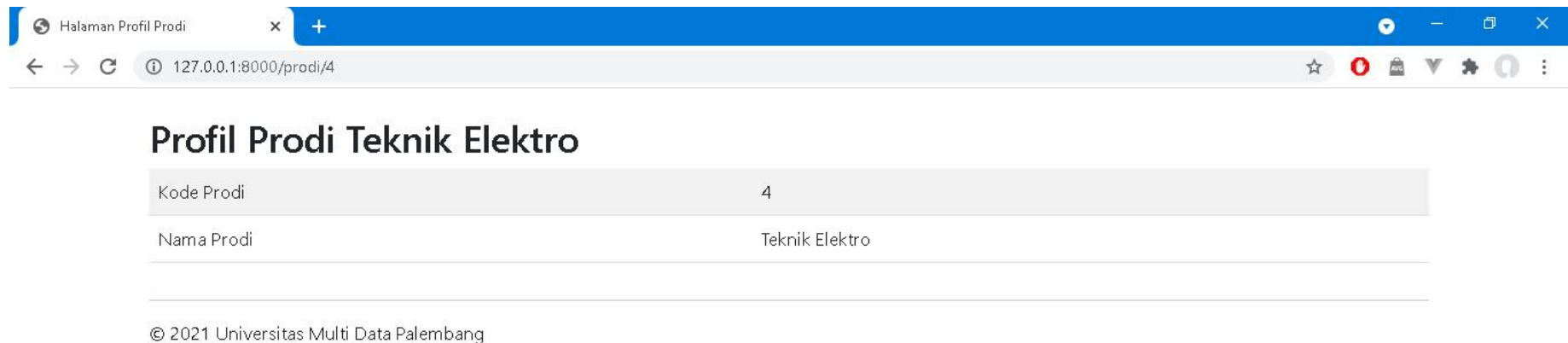
- Berikut isi dari file resources/views/prodi/show.blade.php

```
ProdiController.php Controller.php index.blade.php show.blade.php X
lara_mi4a > resources > views > prodi > show.blade.php > ...
4 @section('content')
5 <div class="row pt-4">
6   <div class="col">
7     <h2>Profil Prodi {{$prodi->nama}}</h2>
8     <table class="table table-striped">
9       <tr>
10        <td>Kode Prodi</td>
11        <td>{{$prodi->id}}</td>
12      </tr>
13      <tr>
14        <td>Nama Prodi</td>
15        <td>{{$prodi->nama}}</td>
16      </tr>
17    </table>
18  </div>
19 </div>
20 @endsection
```

Read

(Menampilkan satu data program studi)

- Silakan buka 127.0.0.1/prodi dan tekan tombol detail di salah satu baris prodi



Halaman Profil Prodi

127.0.0.1:8000/prodi/4

Profil Prodi Teknik Elektro

Kode Prodi	4
Nama Prodi	Teknik Elektro

© 2021 Universitas Multi Data Palembang

Update

- Proses mengubah data yang ada di database
- Proses update terdiri dari 2 bagian, yaitu menampilkan form yang berisi data dari database, serta menjalankan perintah eloquent untuk mengupdate data

Route menampilkan form edit program studi:

```
Route::get('/prodi/{prodi}/edit', [ProdiController::class, 'edit'])->name('prodi.edit');
```

Route untuk proses update data program studi:

```
Route::patch('/prodi/{prodi}', [ProdiController::class, 'update'])->name('prodi.update');
```

Update (Form Edit)

- Tambahkan route di bawah ini pada file routes/web.php

```
Route::get('/prodi/{prodi}/edit', [ProdiController::class, 'edit'])->name('prodi.edit');
```

- Tambahkan method edit() pada ProdiController.php

```
public function edit(Prodi $prodi)
{
    return view('prodi.edit', ['prodi' => $prodi]);
}
```


Update (Form Edit)

- Berikut isi dari file resources/views/prodi/edit.blade.php

```
web.php  ProdiController.php  index.blade.php  show.blade.php  edit.blade.php x  mast  🔍  📄

lara_mi4a > resources > views > prodi > edit.blade.php > div.row.pt-4 > div.col > div.alert.alert-success
1  @extends('layout.master')
2  @section('title', 'Halaman Edit Prodi')
3
4  @section('content')
5      <div class="row pt-4">
6          <div class="col">
7              <h2>Form Edit Prodi</h2>
8              @if (session()->has('info'))
9                  <div class="alert alert-success">
10                     {{ session()->get('info') }}
11                 </div>
12             @endif
13             <form action="{{ route('prodi.update',['prodi' => $prodi->id]) }}" method="post">
14                 @method('PATCH')
15                 @csrf
```

Update (Form Edit)

- Berikut isi dari file resources/views/prodi/edit.blade.php

```
web.php  ProdiController.php  index.blade.php  show.blade.php  edit.blade.php  mast  ↺  □

lara_mi4a > resources > views > prodi > edit.blade.php > ...
16      <div class="form-group">
17          <label for="nama">Nama</label>
18          <input type="text" name="nama" id="nama" class="form-control"
19              value="{{ old('nama') ?? $prodi->nama }}"
20              {{-- null coalescing operator --}}
21              @error('nama')
22                  <div class="text-danger"> {{ $message }} </div>
23              @enderror
24          </div>
25          <button type="submit" class="btn btn-primary mt-2">Ubah</button>
26      </form>
27  </div>
28 </div>
29 @endsection
30
```

Update (Proses Update)

- Tambahkan route di bawah ini pada file routes/web.php

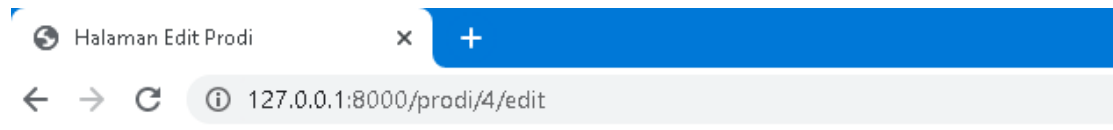
```
Route::patch('/prodi/{prodi}', [ProdiController::class, 'update'])
->name('prodi.update');
```

- Tambahkan method update() pada ProdiController.php

```
public function update(Request $request, Prodi $prodi)
{
    // dump($request->all());
    // dump($prodi);
    $validateData = $request->validate([
        'nama' => 'required|min:5|max:20',
    ]);

    Prodi::where('id', $prodi->id)->update($validateData);
    $request->session()->flash('info', "Data prodi $prodi->nama berhasil diubah");
    return redirect()->route('prodi.index');
}
```

Update



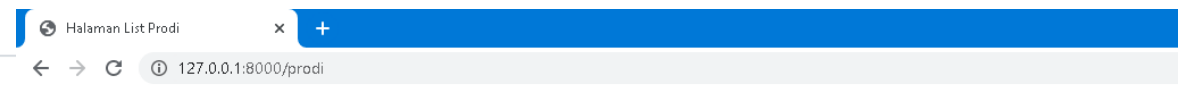
Form Edit Prodi

Nama

Teknik Elektro (S1)

Ubah

© 2021 Universitas Multi Data Palembang



Prodi

Nama

Aksi

Teknik Informatika

Detail

Manajemen Informatika

Detail

Sistem Informasi

Detail

Teknik Elektro (S1)

Detail

© 2021 Universitas Multi Data Palembang

Perubahan file

resources/views/prodi/index.blade.php

```
index.blade.php - Untitled (Workspace) - Visual Studio Code  
ProdiController.php show.blade.php index.blade.php x master.blade.php create.blade.php  
ara_mi4a2 > resources > views > prodi > index.blade.php > div.row.pt-4 > div.col > table.table-striped.table-hover > tbody >  
11 @if (session()->has('info'))  
12     <div class="alert alert-success">  
13         {{ session()->get('info')}}  
14     </div>  
15 @endif  
16 <table class="table table-striped table-hover">  
17     <thead>  
18         <tr>  
19             <th>Nama Prodi</th>  
20             <th>Aksi</th>  
21         </tr>  
22     </thead>  
23     <tbody>  
24         @foreach ($prodi as $item)  
25             <tr>  
26                 <td>{{ $item->nama }}</td>  
27                 <td>  
28                     <a href="{{ url('prodi/'.$item->id) }}" class="btn btn-warning">Detail</a>  
29                     <a href="{{ url('prodi/'.$item->id.'/edit') }}" class="btn btn-info">Ubah</a>  
30                 </td>  
31             </tr>
```

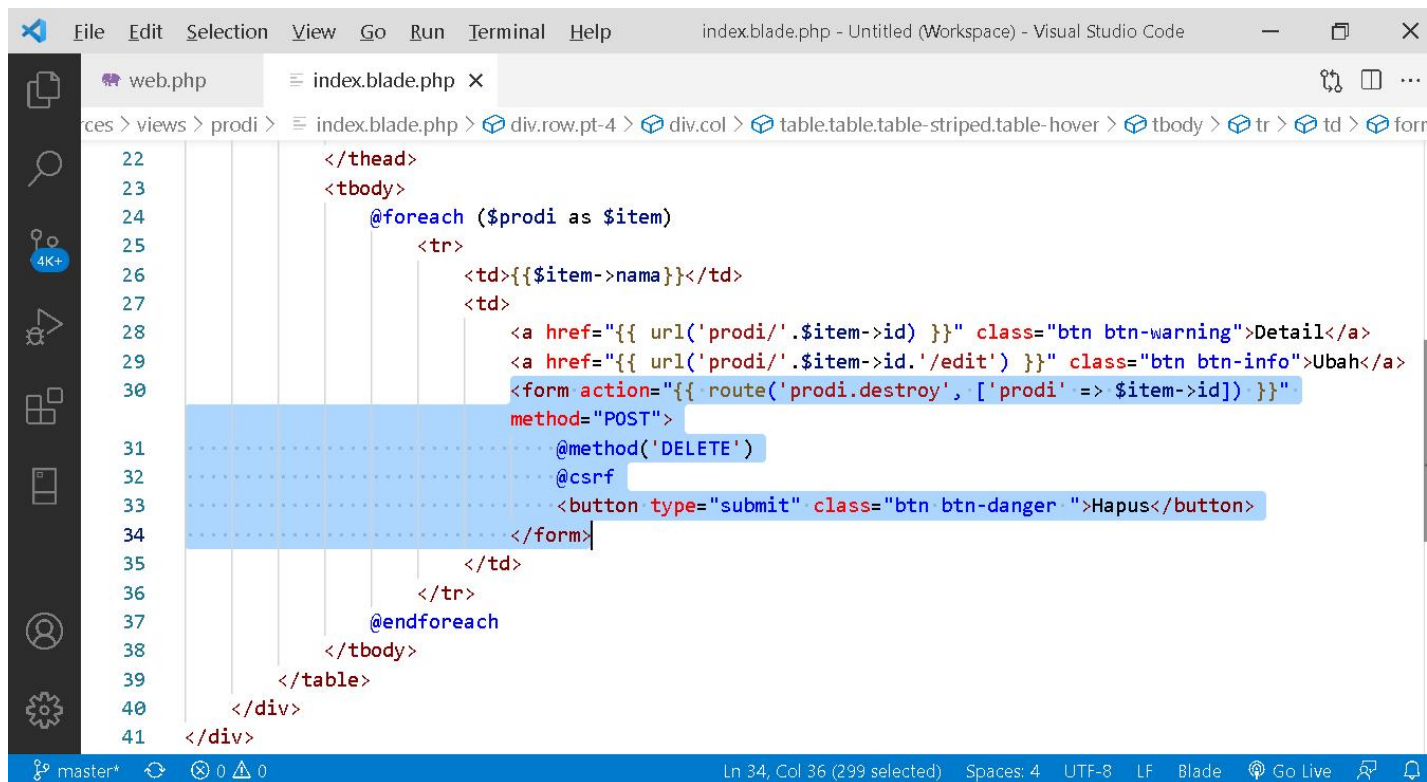
Delete

- Proses menghapus data dari database
- Menggunakan method delete
- Tambahkan route delete

```
Route::delete('/prodi/{prodi}', [ProdiController::class, 'destroy'])->name('prodi.destroy');
```

Delete

- Modifikasi file resources/views/prodi/index.blade.php



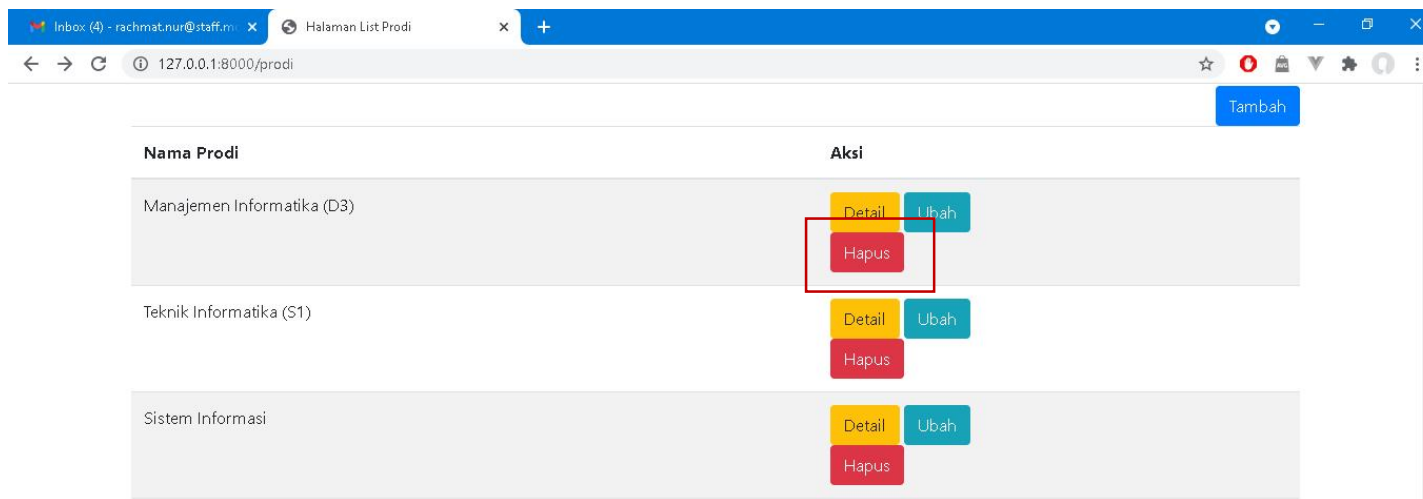
The screenshot shows the Visual Studio Code editor with the file `index.blade.php` open. The editor is displaying a Blade template for a table of products. The table has columns for name, detail, edit, and delete. The delete column is being modified to include a delete button. The code is as follows:

```
22 </thead>
23 <tbody>
24     @foreach ($prodi as $item)
25         <tr>
26             <td>{{ $item->nama }}</td>
27             <td>
28                 <a href="{{ url('prodi/'.$item->id) }}" class="btn btn-warning">Detail</a>
29                 <a href="{{ url('prodi/'.$item->id.'/edit') }}" class="btn btn-info">Ubah</a>
30                 <form action="{{ route('prodi.destroy', ['prodi' => $item->id]) }}"
31                     method="POST">
32                     @method('DELETE')
33                     @csrf
34                     <button type="submit" class="btn btn-danger">Hapus</button>
35                 </form>
36             </td>
37         </tr>
38     @endforeach
39 </tbody>
40 </table>
41 </div>
```

The status bar at the bottom indicates the current position is Line 34, Column 36 (299 selected), with 4 spaces, UTF-8 encoding, LF line endings, and the Blade templating engine.

Delete

- Hasil dari modifikasi file resources/views/prodi/index.blade.php
- Tombol Hapus berada di bawah tombol Detail



- Supaya tombol Hapus berada di samping tombol Ubah, perlu dilakukan perubahan pada file index.blade.php

Delete

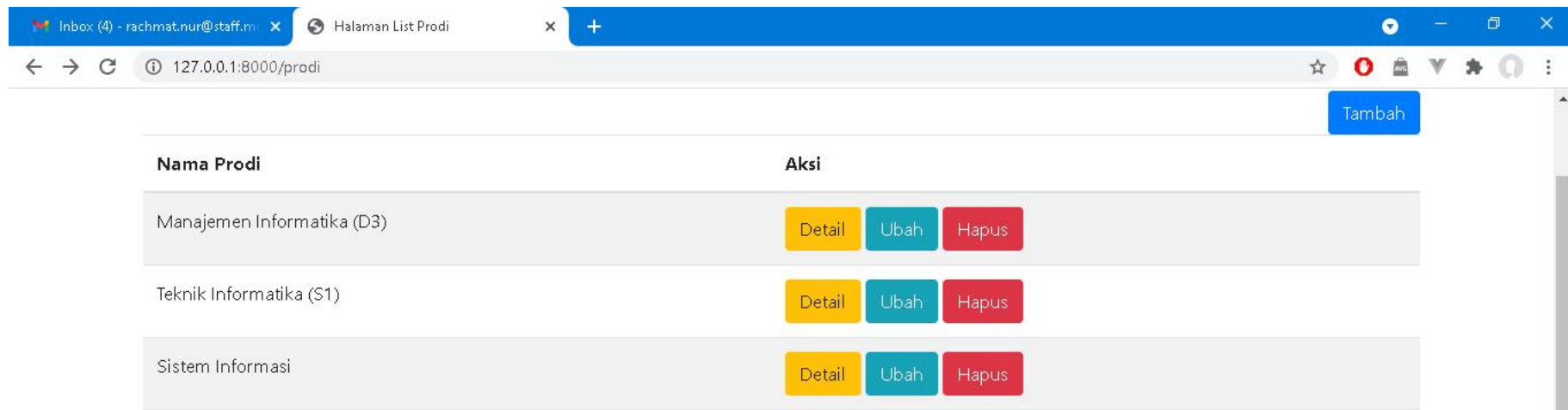
- Tombol Detail dan Ubah pindahkan ke dalam form



```
22     </thead>
23     <tbody>
24         @foreach ($prodi as $item)
25             <tr>
26                 <td>{{ $item->nama }}</td>
27                 <td>
28                     <form action="{{ route('prodi.destroy', ['prodi' => $item->id]) }}"
29                         method="POST">
30                         <a href="{{ url('prodi/' . $item->id) }}" class="btn btn-warning">Detail</a>
31                         <a href="{{ url('prodi/' . $item->id . '/edit') }}" class="btn
32                         btn-info">Ubah</a>
33                     @method('DELETE')
34                     @csrf
35                     <button type="submit" class="btn btn-danger ">Hapus</button>
36                 </form>
37             </td>
38         </tr>
39     @endforeach
40 </tbody>
</table>
</div>
```

Delete

- Hasil dari modifikasi file resources/views/prodi/index.blade.php



Delete

- Tambahkan method `destroy(Prodi $prodi)` pada `ProdiController`

```
public function destroy(Prodi $prodi)
{
    $prodi->delete();
    return redirect()->route('prodi.index')
        ->with("info", "Prodi $prodi->nama berhasil dihapus.");
}
```

- Method `destroy` menerima argumen `$prodi` yang berisi object 1 `prodi`
- `$prodi->delete()` proses menghapus data `prodi`
- Dilanjutkan `redirect` ke route `prodi.index` dengan mengirimkan pesan `Prodi $prodi->nama berhasil dihapus.`

Delete

- Ketika tombol Hapus ditekan, akan diarahkan ke route `prodi.index` dengan menampilkan pesan Prodi ... berhasil dihapus

