

**PEMROGRAMAN WEB LANJUT
JOBSHEET 5 - BLADE VIEW,
WEB TEMPLATING (ADMINLTE), DATATABLES**



Disusun oleh:

Stefanus Ageng Budi Utomo

(2241720126)

**JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
TAHUN AJARAN 2023/2024**

Praktikum 1 - Integrasi Laravel dengan AdminLTE3

1. Dalam root folder project lakukan command berikut, untuk mendefinisikan requirement project.

`composer require jeroennoten/laravel-adminlte`

```
PS D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS> composer require jeroennoten/laravel-adminlte

laravel/sanctum ..... DONE
laravel/tinker ..... DONE
nesbot/carbon ..... DONE
nunomaduro/collision ..... DONE
nunomaduro/termwind ..... DONE
spatie/laravel-ignition ..... DONE

84 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
> @php artisan vendor:publish --tag=laravel-assets --ansi --force

[INFO] No publishable resources for tag [laravel-assets].

A connection timeout was encountered. If you intend to run Composer without connecting to the internet, run the command again prefixed with COMPOSER_DISABLE_NETWORK=1 to make Com
poser run in offline mode.
https://repo.packagist.org could not be fully loaded (curl error 28 while downloading https://repo.packagist.org/p2/brick/math.json: Send failure: Connection was aborted), packag
e information was loaded from the local cache and may be out of date
No security vulnerability advisories found.
Using version ^3.11 for jeroennoten/laravel-adminlte
PS D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS>
```

2. Melakukan instalasi requirement project di atas dengan command berikut:

`php artisan adminlte:install`

```
PS D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS> php artisan adminlte:install
Basic assets installed successfully.
Configuration file installed successfully.
Translation files installed successfully.
The installation is complete.
```

Perintah di atas akan meng-install:

- AdminLTE distribution files dan dependensi nya (Bootstrap, jQuery, etc.) dalam folder public/vendor .
 - Konfigurasi package di file config/adminlte.php
 - Paket translasi di folder lang/vendor/adminlte/
 - Dalam composer.json akan otomatis ditambahkan require untuk laravel-adminlte
3. Buat file resources/views/layout/app.blade.php. Isi dengan kode berikut

```

@extends('adminlte::page')
{{-- Extend and customize the browser title --}}
@section('title')
    {{ config('adminlte.title') }}
    @hasSection('subtitle')
        | @yield('subtitle')
    @endif
@stop

{{-- Extend and customize the page content header --}}

@section('content_header')
    @hasSection('content_header_title')
        <h1 class="text-muted">
            @yield('content_header_title')
            @hasSection('content_header_subtitle')
                <small class="text-dark">
                    <i class="fas fa-xs fa-angle-right text-muted"></i>
                    @yield('content_header_subtitle')
                </small>
            @endif
        </h1>
    @endif
@stop

{{-- Rename section content to content_body --}}

@section('content')
    @yield('content_body')
@stop

@section('footer')
    <div class="float-right">
        Version: {{ config('app.version', '1.0.0') }}
    </div>
    <strong>
        <a href="{{ config('app.company_url', '#') }}">
            {{ config('app.company_name', 'My company') }}
        </a>
    </strong>
@stop

{{-- Add common Javascript/Jquery code --}}

@push('js')
    <script>
        $(document).ready(function() {
            // Add your common script logic here...
        });
    </script>
@endpush

```

```

{{-- Add common CSS customizations --}}

@push('css')
    <style type="text/css">
        {{-- You can add AdminLTE customizations here --}}
        /*
         * .card-header {
         *   border-bottom: none;
         * }
         * .card-title {
         *   font-weight: 600;
         * }
         */
    </style>
@endpush

```

4. Edit resources/views/welcome.blade.php, kemudian replace seluruh kodenya dengan kode berikut

```

@extends('layout.app')

{{-- Customize layout sections --}}
@section('subtitle', 'Welcome')
@section('content_header_title', 'Home')
@section('content_header_subtitle', 'Welcome')

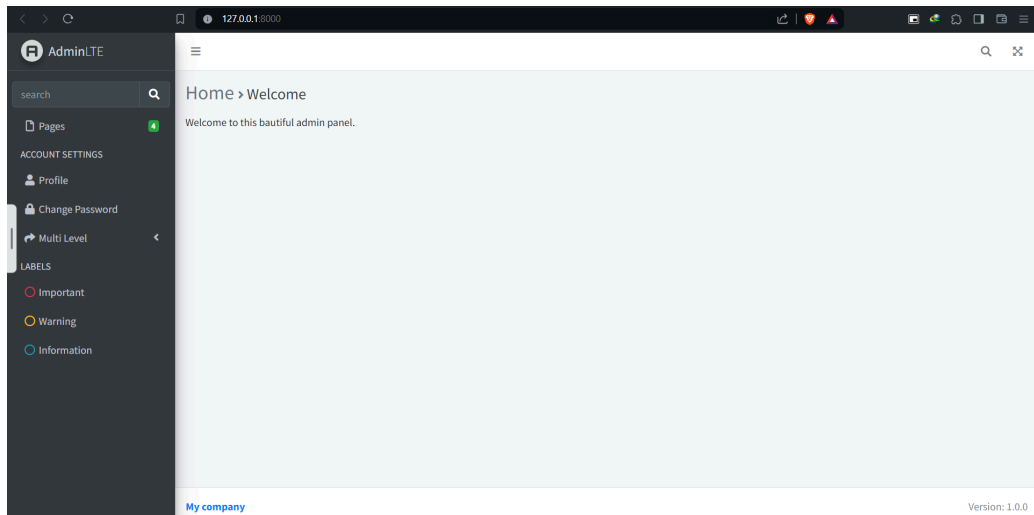
{{-- Content body: main --}}
@section('content_body')
    <p>Welcome to this beautiful admin panel.</p>
@stop

{{-- Push extra CSS --}}
@push('css')
    {{-- Add here extra stylesheets --}}
    {{-- <link rel="stylesheet" href="/css/admin_custom.css"> --}}
@endpush

{{-- Push extra scripts --}}
@push('js')
    <script>
        console.log("Hi, I'm using the Laravel-AdminLTE package!");
    </script>
@endpush

```

5. Kembali ke browser, menuju ke halaman awal.



Praktikum 2 - Integrasi dengan DataTables

1. Install Laravel DataTables

```
composer require laravel/ui --dev
```

```
composer require yajra/laravel-datatables:^10.0
```

```
PS D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS> composer require laravel/ui --dev
./composer.json has been updated
Running composer update laravel/ui
Loading composer repositories with package information
Updating dependencies
Lock file operations: 1 install, 0 updates, 0 removals
- Locking laravel/ui (v4.5.0)
Writing lock file
Installing dependencies from lock file (including require-dev)
Package operations: 1 install, 0 updates, 0 removals
- Downloading laravel/ui (v4.5.0)
```

```
PS D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS> composer require yajra/laravel-datatables:^10.0
The "10.0" constraint for "yajra/laravel-datatables" appears too strict and will likely not match what you want.
./composer.json has been updated
Running composer update yajra/laravel-datatables
Loading composer repositories with package information
Updating dependencies
Lock file operations: 7 installs, 0 updates, 0 removals
- Locking league/fractal (0.20.1)
- Locking yajra/laravel-datatables (v10.0.0)
- Locking yajra/laravel-datatables-buttons (v10.0.9)
- Locking yajra/laravel-datatables-editor (v1.25.4)
- Locking yajra/laravel-datatables-fractal (v10.0.0)
- Locking yajra/laravel-datatables-html (v10.12.0)
- Locking yajra/laravel-datatables-oracle (v10.11.4)
Writing lock file
Installing dependencies from lock file (including require-dev)
```

2. Pastikan nodejs sudah terinstall, dengan perintah npm -v. Jika belum install nodejs package manager dari <https://nodejs.org/dist/v20.11.1/nodev20.11.1-x64.msi> Npm digunakan untuk manajemen package javascript.

3. Install Laravel DataTables Vite dan sass

```
npm i laravel-datatables-vite --save-dev
```

```
npm install -D sass
```

```
PS D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS> npm i laravel-datatables-vite --save-dev
added 33 packages, and audited 34 packages in 20s

8 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
npm notice
npm notice New major version of npm available! 8.19.2 -> 10.5.0
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.5.0
npm notice Run `npm install -g npm@10.5.0` to update!
npm notice
PS D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS>
```

```
PS D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS> npm install -D sass
added 15 packages, and audited 49 packages in 5s

10 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS>
```

4. Edit file resources/js/app.js

```
resources > js > JS app.js
1 import './bootstrap';
2 import '../sass/app.scss';
3 import 'laravel-datatables-vite';
4
```

5. Buatlah file resources/sass/app.scss

```
resources > sass > app.scss
1 // Fonts
2 @import url('https://fonts.bunny.net/css?family=Nunito');
3
4 // Bootstrap
5 @import 'bootstrap/scss/bootstrap';
6
7 // DataTables
8 @import 'bootstrap-icons/font/bootstrap-icons.css';
9 @import 'datatables.net-bs5/css/dataTables.bootstrap5.min.css';
10 @import 'datatables.net-buttons-bs5/css/buttons.bootstrap5.min.css';
11 @import 'datatables.net-select-bs5/css/select.bootstrap5.min.css';
12
```

6. Jalankan dengan npm run dev

```
PS D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS> npm run dev
> dev
> vite

VITE v5.1.6 ready in 420 ms

→ Local:   http://localhost:5173/
→ Network: use --host to expose
→ press h + enter to show help

LARAVEL v10.47.0 plugin v1.0.2

→ APP_URL: http://localhost
```

7. Selanjutnya kita akan buat datatables untuk kategori

php artisan datatables:make Kategori

```
PS D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS> php artisan datatables:make
Kategori

INFO DataTable [D:\Abi\kuliah\semester 4\Pemrograman Web Lanjut\code\PWL_POS\app\DataTables\KategoriDataTable.php] created successfully.
```

8. Kita edit KategoriDataTable untuk mengatur kolom apa saja yang ingin ditampilkan
Sesuaikan dengan kodeprogram berikut

```

app > DataTables > KategoriDataTable.php > ...
1  <?php
2
3  namespace App\DataTables;
4
5  use App\Models\Kategori;
6  use App\Models\KategoriModel;
7  use Illuminate\Database\Eloquent\Builder as QueryBuilder;
8  use Yajra\DataTables\EloquentDataTable;
9  use Yajra\DataTables\Html\Builder as HtmlBuilder;
10 use Yajra\DataTables\Html\Button;
11 use Yajra\DataTables\Html\Column;
12 use Yajra\DataTables\Html\Editor\Editor;
13 use Yajra\DataTables\Html\Editor\Fields;
14 use Yajra\DataTables\Services\DataTable;
15
16 class KategoriDataTable extends DataTable
17 {
18     /**
19      * Build the DataTable class.
20      *
21      * @param QueryBuilder $query Results from query() method.
22      */
23     public function dataTable(QueryBuilder $query): EloquentDataTable
24     {
25         return (new EloquentDataTable($query))
26             ->addColumn('action', 'kategori.action')
27             ->setRowId('id');
28     }
29
30     /**
31      * Get the query source of dataTable.
32      */
33     public function query(KategoriModel $model): QueryBuilder
34     {
35         return $model->newQuery();
36     }
37

```

```

38     /**
39      * Optional method if you want to use the html builder.
40      */
41     public function html(): HtmlBuilder
42     {
43         return $this->builder()
44             ->setTableId('kategori-table')
45             ->columns($this->getColumns())
46             ->minifiedAjax()
47             //->dom('Bfrtip')
48             ->orderBy(1)
49             ->selectStyleSingle()
50             ->buttons([
51                 Button::make('excel'),
52                 Button::make('csv'),
53                 Button::make('pdf'),
54                 Button::make('print'),
55                 Button::make('reset'),
56                 Button::make('reload')
57             ]);
58     }
59
60     /**
61      * Get the dataTable columns definition.
62      */
63     public function getColumns(): array
64     {
65         return [
66             Column::computed('action')
67                 ->exportable(false)
68                 ->printable(false)
69                 ->width(60)
70                 ->addClass('text-center'),
71             Column::make('id'),
72             Column::make('add your columns'),
73             Column::make('created_at'),
74             Column::make('updated_at'),
75         ];
76     }
77

```

9. Ubah kategori model, sesuaikan seperti berikut

```

app > Models > KategoriModel.php > ...
1  <?php
2
3  namespace App\Models;
4
5  use Illuminate\Database\Eloquent\Factories\HasFactory;
6  use Illuminate\Database\Eloquent\Model;
7  use Illuminate\Database\Eloquent\Relations\HasMany;
8
9  class KategoriModel extends Model
10 {
11     // use HasFactory;
12     protected $model = 'm_kategori';
13     protected $primaryKey = 'kategori_id';
14
15     protected $fillable = ['kategori_kode', 'kategori_nama'];
16
17     public function barang(): HasMany
18     {
19         return $this->hasMany(BarangModel::class, 'barang_id', 'barang_id');
20     }
21 }

```

10. Ubah Kategori Controller sesuaikan dengan skrip berikut:

```

app > Http > Controllers > KategoriController.php > KategoriController >
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\DataTables\KategoriDataTable;
6  use Illuminate\Http\Request;
7  use Illuminate\Support\Facades\DB;
8
9  class KategoriController extends Controller
10 {
11     public function index(KategoriDataTable $dataTable)
12     {
13         return $dataTable->render('kategori.index');
14     }
15 }

```

11. Buat folder kategori di dalam resources/view, kemudian buat view blade index untuk kategori resources/views/kategori/index.blade.php


```
resources > views > kategori > index.blade.php > ...
1  @extends('layouts.app')
2
3  {{-- Customize layout sections --}}
4
5  @section('subtitle', 'Kategori')
6  @section('content_header_title', 'Home')
7  @section('content_header_subtitle', 'Kategori')
8
9  @section('content')
10     <div class="container">
11         <div class="card">
12             <div class="card-header">Manager Kategori</div>
13             <div class="card-body">
14                 {{ $dataTable->table() }}
15             </div>
16         </div>
17     </div>
18 @endsection
19
20 @push('scripts')
21     {{ $dataTable->scripts() }}
22 @endpush
```

12. Pastikan route untuk kategori sudah tersedia

```
Route::get('/kategori', [KategoriController::class, 'index']);
```

13. Sesuaikan app layout seperti kode berikut

```

resources > views > layout > app.blade.php > link
1  @extends('adminlte::page')
2  {{-- Extend and customize the browser title --}}
3  @section('title')
4      {{ config('adminlte.title') }}
5      @hasSection('subtitle')
6          | @yield('subtitle')
7      @endif
8  @stop
9
10 @vite('resource/js/app.js')
11
12 {{-- Extend and customize the page content header --}}
13
14 @section('content_header')
15     @hasSection('content_header_title')
16         <h1 class="text-muted">
17             @yield('content_header_title')
18             @hasSection('content_header_subtitle')
19                 <small class="text-dark">
20                     <i class="fas fa-xs fa-angle-right text-muted"></i>
21                     @yield('content_header_subtitle')
22                 </small>
23             @endif
24         </h1>
25     @endif
26 @stop
27
28 {{-- Rename section content to content_body --}}
29
30 @section('content')
31     @yield('content_body')
32 @stop
33
34 {{-- Create a common footer --}}
35
36 @section('footer')
37     <div class="float-right">
38         Version: {{ config('app.version', '1.0.0') }}
39     </div>
40     <strong>
41         <a href="{{ config('app.company_url', '#') }}">
42             {{ config('app.company_name', 'My company') }}
43         </a>
44     </strong>
45 @stop
46
47 {{-- Add common Javascript/Query code --}}
48
49 @push('js')
50     <script>
51         $(document).ready(function() {
52             // Add your common script logic here...
53         });
54     </script>
55     <script src="https://cdn.datatables.net/2.0.2/js/dataTables.js"></script>
56 @endpush
57
58 @stack('scripts')
59
60 {{-- Add common CSS customizations --}}
61
62 @push('css')
63     <link rel="stylesheet" href="https://cdn.datatables.net/2.0.2/css/dataTables.dataTables.css" />
64     <style type="text/css">
65         {{-- You can add AdminLTE customizations here --}}
66         /*
67         .card-header {
68             border-bottom: none;
69         }
70         .card-title {
71             font-weight: 600;
72         }
73         */
74     </style>
75 @endpush
76

```

14. Menset ViteJs / script type defaults

```

app > Providers > AppServiceProvider.php > AppServiceProvider > boot
1  <?php
2
3  namespace App\Providers;
4
5  use Illuminate\Support\ServiceProvider;
6  use Yajra\DataTables\Html\Builder;
7
8  class AppServiceProvider extends ServiceProvider
9  {
10     /**
11      * Register any application services.
12      */
13     public function register(): void
14     {
15         //
16     }
17
18     /**
19      * Bootstrap any application services.
20      */
21     public function boot(): void
22     {
23         //
24         Builder::useVite();
25     }
26 }

```

15. Isikan beberapa data ke table kategori
16. Datatables sudah dapat di load di /kategori

127.0.0.1:8000/kategori

Home > Kategori

Manager Kategori

10 entries per page Search:

Kategori Id	Kategori Kode	Kategori Nama	Created At	Updated At
5	KBE5	keyboard		
4	KBE4	headphones		
3	KBE3	camera		
2	KBE2	smartphone		
1	KBE1	laptop		

Showing 1 to 5 of 5 entries

« < 1 > »

Praktikum 3 - Membuat form lalu menyimpan data dalam database

1. Menyesuaikan routing, tambahkan dua routing berikut

```
Route::get('/kategori/create', [KategoriController::class, 'create']);
Route::get('/kategori', [KategoriController::class, 'store']);
```

2. Tambahkan dua function berikut dalam KategoriController

```
app > Http > Controllers > KategoriController.php > KategoriController > store
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use App\DataTables\KategoriDataTable;
6  use App\Models\KategoriModel;
7  use Illuminate\Http\Request;
8  use Illuminate\Support\Facades\DB;
9
10 class KategoriController extends Controller
11 {
12     public function index(KategoriDataTable $dataTable)
13     {
14         return $dataTable->render('kategori.index');
15     }
16
17     public function create()
18     {
19         return view('kategori.create');
20     }
21
22     public function store(Request $request)
23     {
24         KategoriModel::create([
25             'kategori_kode' => $request->kodeKategori,
26             'kategori_nama' => $request->namaKategori,
27         ]);
28         return redirect('/kategori');
29     }
30 }
```

3. Dalam folder views/kategori, buatlah file dengan nama create.blade.php

```

@extends('layout.app')
{{-- Customize layout sections --}}
@section('subtitle', 'Kategori')
@section('content_header_title', 'Kategori')
@section('content_header_subtitle', 'Create')
{{-- Content body: main page content --}}
@section('content')
    <div class="container">
        <div class="card card-primary">
            <div class="card-header">
                <h3 class="card-title">Buat kategori baru</h3>
            </div>

            <form action=".."../kategori" method="post">
                <div class="card-body">
                    <div class="form-group">
                        <label for="kodeKategori">Kode Kategori</label>
                        <input type="text" class="form-control" id="kodeKategori" name="kodeKategori"
                        placeholder="">
                    </div>
                    <div class="form-group">
                        <label for="kodeKategori">Kode Kategori</label>
                        <input type="text" class="form-control" id="kodeKategori" name="kodeKategori"
                        placeholder="">
                    </div>
                </div>

                <div class="card-footer">
                    <button type="submit" class="btn btn-primary">Submit</button>
                </div>
            </form>
        </div>
    </div>
@endsection

```

4. Kita lakukan pengecualian proteksi CsrfToken. Karena kita belum melakukan otentikasi . Kita edit dalam file berikut:

```

<?php

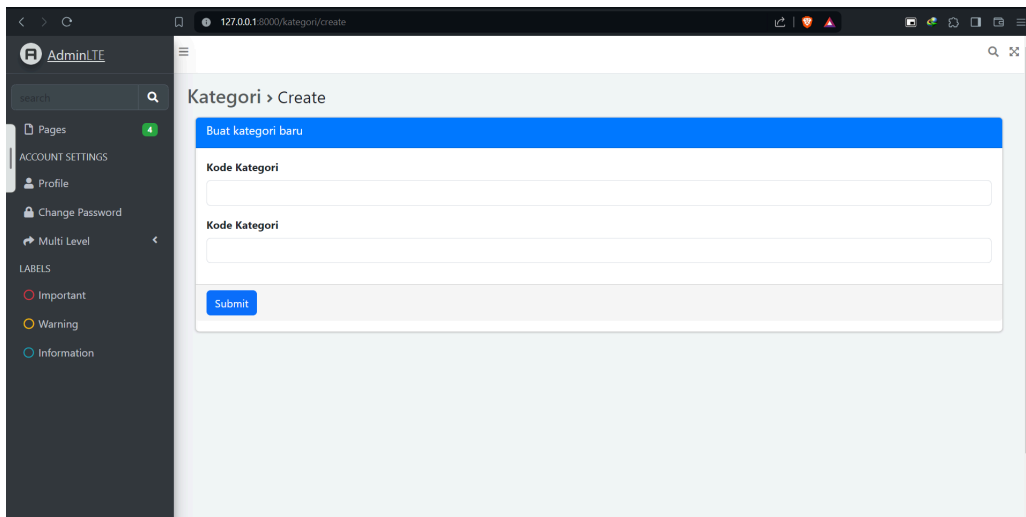
namespace App\Http\Middleware;

use Illuminate\Foundation\Http\Middleware\VerifyCsrfToken as Middleware;

class VerifyCsrfToken extends Middleware
{
    /**
     * The URIs that should be excluded from CSRF verification.
     *
     * @var array<int, string>
     */
    protected $except = [
        //
        '/kategori'
    ];
}

```

5. Akses kategori/create



6. Halaman kategori

Home > Kategori

Manager Kategori

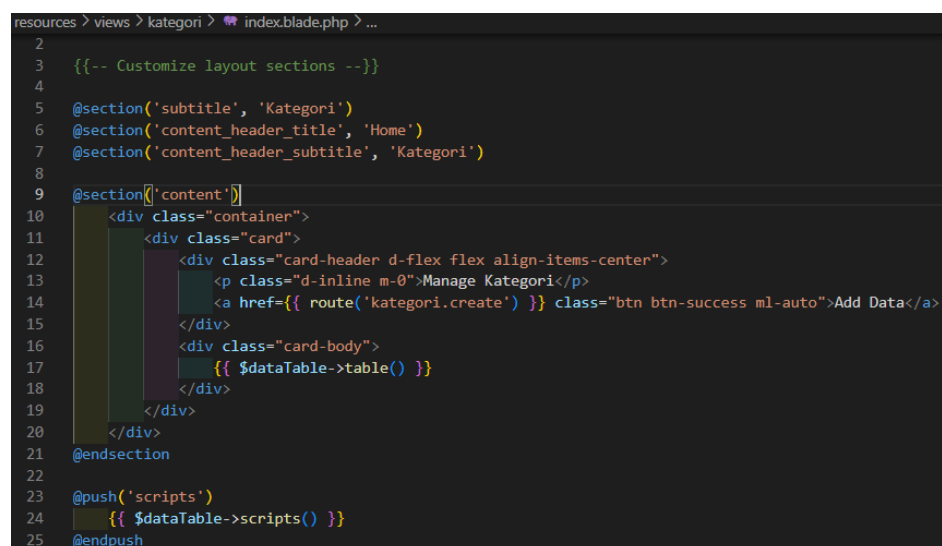
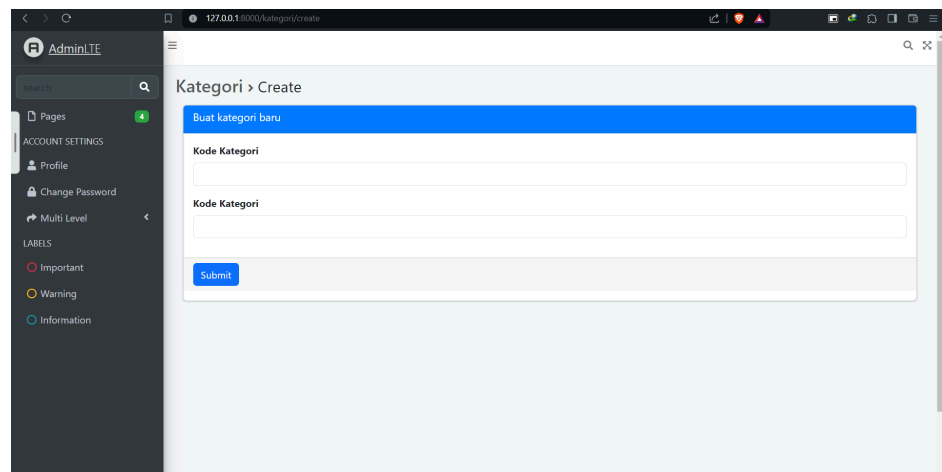
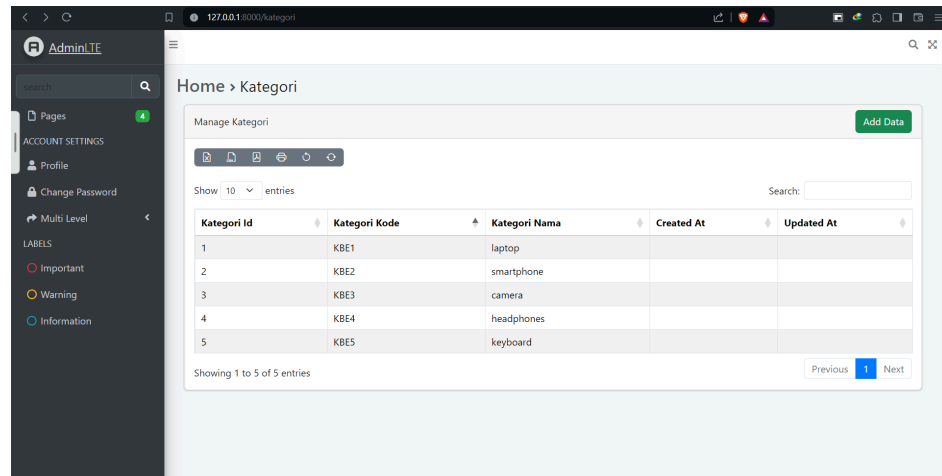
Show 10 entries Search:

Kategori Id	Kategori Kode	Kategori Nama	Created At	Updated At
5	KBES	keyboard		
4	KBE4	headphones		
3	KBE3	camera		
2	KBE2	smartphone		
1	KBE1	laptop		

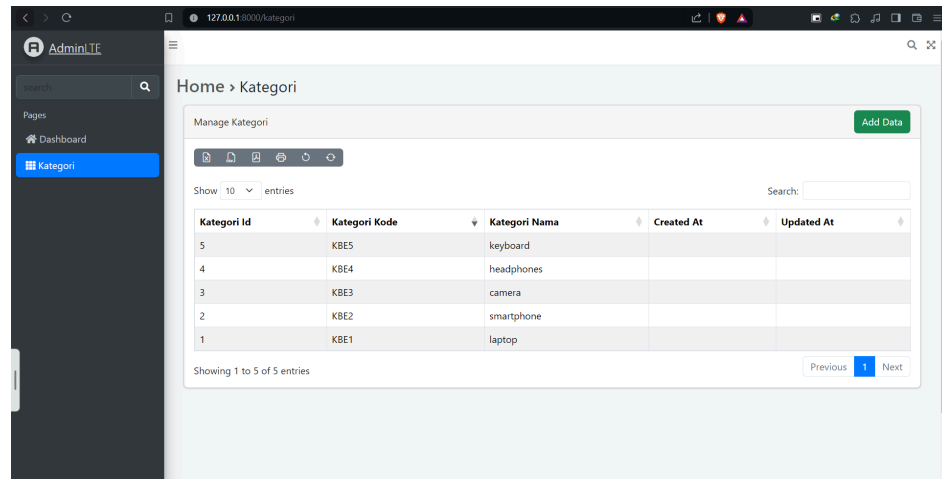
Showing 1 to 5 of 5 entries Previous 1 Next

Tugas Praktikum

1. Tambahkan button Add di halam manage kategori, yang mengarah ke create kategori baru



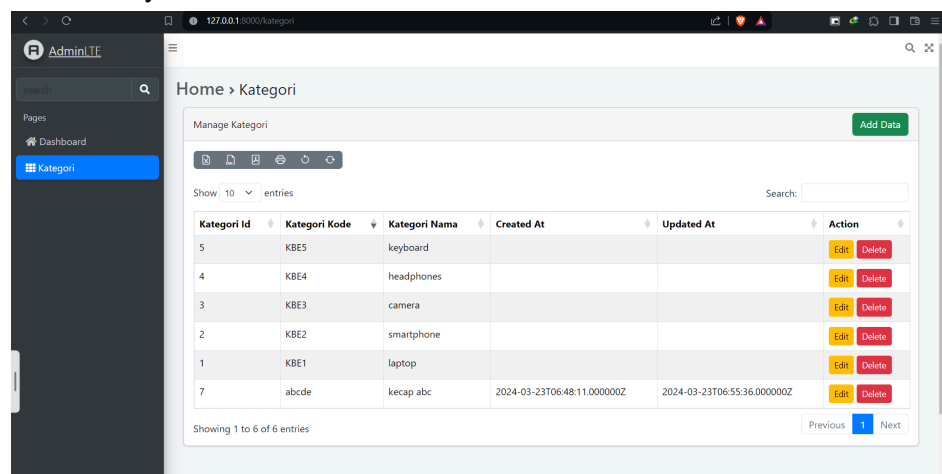
2. Tambahkan menu untuk halaman manage kategori, di daftar menu navbar



```

config > adminlte.php
3   return [
296       'menu' => [
307
308           // Sidebar items:
309           [
310               'type' => 'sidebar-menu-search',
311               'text' => 'search',
312           ],
313           [
314               'text' => 'blog',
315               'url' => 'admin/blog',
316               'can' => 'manage-blog',
317           ],
318           ['header' => 'Pages'],
319           [
320               'text' => 'Dashboard',
321               'url' => '/',
322               'icon' => 'fa fa-home',
323           ],
324           [
325               'text' => 'Kategori',
326               'url' => '/kategori',
327               'icon' => 'fa fa-th',
328           ]
329       ]
330   ]
  
```

3. Tambahkan action edit di datatables dan buat halaman edit serta controllernya




```

app > DataTables > KategoriDataTable.php > KategoriDataTable > dataTable > Closure
16 class KategoriDataTable extends DataTable
23 {
24     public function dataTable(QueryBuilder $query): EloquentDataTable
25     {
26         return (new EloquentDataTable($query))
27             ->addColumn('action', function ($data) {
28                 $editUrl = route('kategori.edit', $data->kategori_id);
29                 $deleteUrl = route('kategori.delete', $data->kategori_id);
30
31                 return "<a href='{$editUrl}' class='btn btn-sm btn-warning'>Edit</a>
32                     <form action='{$deleteUrl}' method='POST' style='display:inline'>
33                         . method_field('DELETE') . csrf_field() . "
34                         <button type='submit' class='btn btn-sm btn-danger' onclick='return confirm(\"Are you sure?\")>Delete</button>
35                     </form>";
36             })
37             ->setRowId('id');

```

The screenshot shows a web browser window with the URL '127.0.0.1:8000/kategori/edit/7'. The page title is 'Kategori > Edit'. On the left is a sidebar with 'AdminLTE' and a search bar. Below the sidebar are links for 'Dashboard' and 'Kategori'. The main content area contains a form titled 'Edit kategori'. The form has two input fields: 'Kode Kategori' with the value 'abcde' and 'Nama Kategori' with the value 'kecap abc'. At the bottom of the form is a blue 'Submit' button.

```

resources > views > kategori > edit.blade.php > div.container > div.card.card-primary > form
1 @extends('layout.app')
2 {{-- Customize layout sections --}}
3 @section('subtitle', 'Kategori')
4 @section('content_header_title', 'Kategori')
5 @section('content_header_subtitle', 'Edit')
6 {{-- Content body: main page content --}}
7 @section('content')
8     <div class="container">
9         <div class="card card-primary">
10             <div class="card-header">
11                 <h3 class="card-title">Edit kategori</h3>
12             </div>
13
14             <form action="{{ route('kategori.save', $data->kategori_id) }}" method="post">
15
16                 {{ csrf_field() }}
17                 {{ method_field('PUT') }}
18
19                 <div class="card-body">
20                     <div class="form-group">
21                         <label for="kodeKategori">Kode Kategori</label>
22                         <input type="text" class="form-control" id="kodeKategori" name="kategori_kode" placeholder=""
23                             value="{{ $data->kategori_kode }}">
24                     </div>
25                     <div class="form-group">
26                         <label for="namaKategori">Nama Kategori</label>
27                         <input type="text" class="form-control" id="kodeKategori" name="kategori_nama" placeholder=""
28                             value="{{ $data->kategori_nama }}">
29                     </div>
30                 </div>
31
32                 <div class="card-footer">
33                     <button type="submit" class="btn btn-primary">Submit</button>
34                 </div>
35             </form>
36         </div>
37     </div>
38 @endsection

```

```

app > Http > Controllers > KategoriController.php > KategoriController > edit_save
10 class KategoriController extends Controller
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31 public function edit($id)
32 {
33     $kategori = KategoriModel::find($id);
34     return view('kategori.edit', ['data' => $kategori]);
35 }
36
37 public function edit_save($id, Request $request)
38 {
39     $item = KategoriModel::find($id);
40     $item->kategori_kode = $request->kategori_kode;
41     $item->kategori_nama = $request->kategori_nama;
42     $item->save();
43     return redirect('/kategori');
44 }

```

```

Route::get('/kategori/edit/{id}', [KategoriController::class, 'edit']->name('kategori.edit'));
Route::put('/kategori/edit_save/{id}', [KategoriController::class, 'edit_save']->name('kategori.save'));

```

Home > Kategori

Manage Kategori Add Data

Show: 10 entries Search:

Kategori Id	Kategori Kode	Kategori Nama	Created At	Updated At	Action
5	KBE5	keyboard			Edit Delete
4	KBE4	headphones			Edit Delete
3	KBE3	camera			Edit Delete
2	KBE2	smartphone			Edit Delete
1	KBE1	laptop			Edit Delete
7	abc	kecap abc	2024-03-23T06:48:11.000000Z	2024-03-23T07:13:44.000000Z	Edit Delete

Showing 1 to 6 of 6 entries Previous 1 Next

4. Tambahkan action delete di datatables serta controllernya

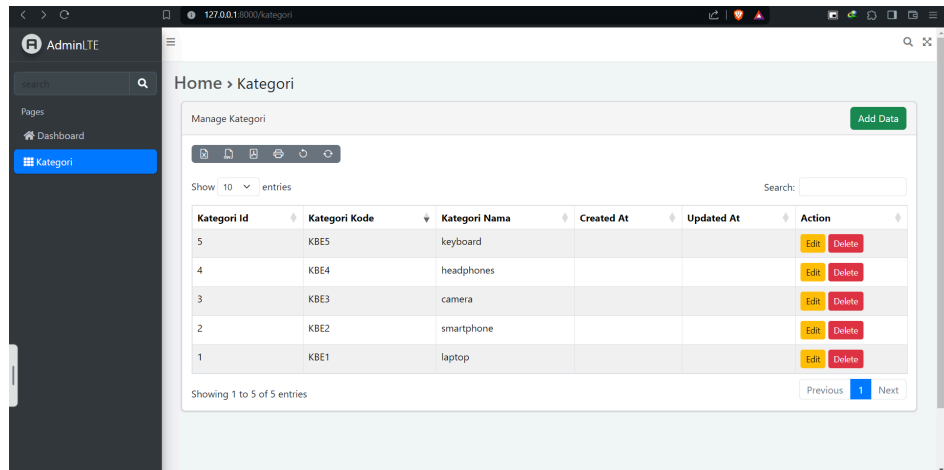
Home > Kategori

Manage Kategori Add Data

Show: 10 entries Search:

Kategori Id	Kategori Kode	Kategori Nama	Created At	Updated At	Action
5	KBE5	keyboard			Edit Delete
4	KBE4	headphones			Edit Delete
3	KBE3	camera			Edit Delete
2	KBE2	smartphone			Edit Delete
1	KBE1	laptop			Edit Delete
7	abc	kecap abc	2024-03-23T06:48:11.000000Z	2024-03-23T07:13:44.000000Z	Edit Delete

Showing 1 to 6 of 6 entries Previous 1 Next



```
app > Http > Controllers > KategoriController.php > KategoriController > edit_save
10 class KategoriController extends Controller
46
47     public function delete($id)
48     {
49         $item = KategoriModel::find($id);
50         if (!$item) {
51             return redirect()->route('/kategori')->with('error', 'Data not found');
52         }
53         $item->delete($id);
54         return redirect()->route('/kategori')->with('success', 'Data delete successfully');
55     }
56 }
```

```
Route::delete('/kategori/delete/{id}', [KategoriController::class, 'delete']->name('kategori.delete'));
```