



Nama : Muhammad Bagus Indrawan
Kelas : TI-2H/20
NIM : 2241720217

Mata Kuliah : Pemrograman Web Lanjut (PWL)
Program Studi : D4 – Teknik Informatika / D4 – Sistem Informasi Bisnis
Semester : 4 (empat) / 6 (enam)
Pertemuan ke- : 5 (satu)

JOBSHEET 05

Blade View, Web Templating(AdminLTE), Datatables

View merupakan tempat bagi kita untuk meletakkan kode-kode HTML. Sehingga dalam laravel tidak menggunakan lagi file berformat .html. Kita masih menggunakan tagging HTML, tapi kita perlu handle tampilan dengan lebih canggih. Menampilkan data yang diberikan oleh controller. Untuk itu kita akan menggunakan templating engine, yaitu Blade.

Blade merupakan templating engine bawaan Laravel. Berguna untuk mempermudah dalam menulis kode tampilan. Dan juga memberikan fitur tambahan untuk memanipulasi data di view yang dilempar dari controller. Blade juga memungkinkan penggunaan plain PHP pada kode View.

Sesuai dengan **studi Kasus PWL.pdf**.

Jadi project Laravel 10 kita masih sama dengan menggunakan repositori **PWL_POS**.

*Project **PWL_POS** akan kita gunakan sampai pertemuan 12 nanti, sebagai project yang akan kita pelajari*



Praktikum 1 – Integrasi Laravel dengan AdminLTE3

Cara paling cepat dan clean dapat dilakukan menggunakan repository dalam tautan berikut
<https://github.com/jeroennoten/Laravel-AdminLTE/wiki/Installation>.

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

```
composer require jeroennoten/laravel-adminlte
```

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

```
adminlte:install
```

Perintah di atas akan meng-install:

- **AdminLTE distribution files** dan dependensinya (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`

```
{ } composer.json > ...  
1 {  
2   "name": "laravel/laravel",  
3   "type": "project",  
4   "description": "The skeleton application for the Laravel framework.",  
5   "keywords": ["laravel", "framework"],  
6   "license": "MIT",  
7   "require": {  
8     "php": "^8.1",  
9     "guzzlehttp/guzzle": "^7.2",  
10    "jeroennoten/laravel-adminlte": "^3.9",  
11    "laravel/framework": "^10.10",  
12    "laravel/sanctum": "^3.3",  
13    "laravel/tinker": "^2.8"  
14  },  
15  "require-dev": {  
16    "fakerphp/faker": "^1.9.1",  
17    "laravel/pint": "^1.0",  
18    "laravel/sail": "^1.18",  
19    "mockery/mockery": "^1.4.4",  
20    "nunomaduro/collision": "^7.0",  
21    "phpunit/phpunit": "^10.1",  
22    "spatie/laravel-ignition": "^2.0"  
23  },  
24  "autoload": {  
25    "psr-4": {
```



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

{{-- Create a common footer --}}

@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



```
resources > views > welcome.blade.php > ...
1  @extends('layouts.app')
2
3  {{-- Customize layout sections --}}
4
5  @section('subtitle', 'Welcome')
6  @section('content_header_title', 'Home')
7  @section('content_header_subtitle', 'Welcome')
8
9  {{-- Content body: main page content --}}
10
11 @section('content_body')
12 |   <p>Welcome to this beautiful admin panel.</p>
13 @stop
14
15 {{-- Push extra CSS --}}
16
17 @push('css')
18 |   {{-- Add here extra stylesheets --}}
19 |   {{-- <link rel="stylesheet" href="/css/admin_custom.css"> --}}
20 @endpush
21
22 {{-- Push extra scripts --}}
23
24 @push('js')
25 |   <script> console.log("Hi, I'm using the Laravel-AdminLTE package!"); </script>
26 @endpush
```

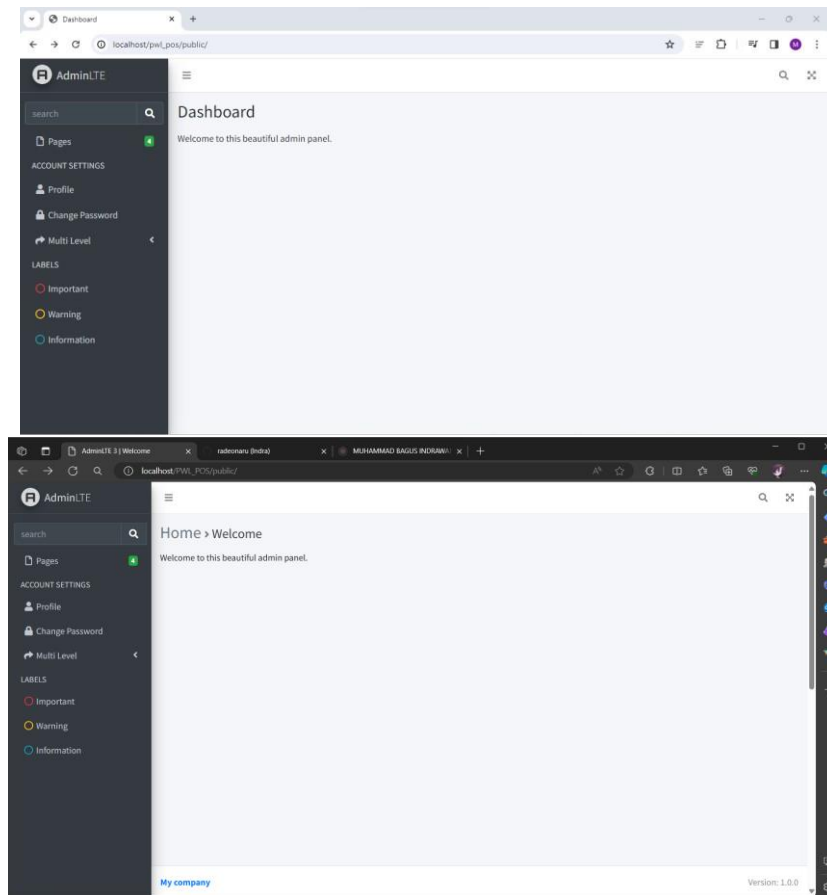
Untuk menggunakan blade template ini cukup dengan extend **AdminLTE layout** dengan cara `@extends('adminlte::page')`.

Template yields di beberapa bagian terklasifikasi dalam 2 yield:

- **main:** Biasa digunakan untuk extending the layout.
- **misc:** untuk kasus yang tidak biasa, atau hanya situasi tertentu.

Dokumentasi lebih detail terdapat di link berikut: <https://github.com/jeroennoten/Laravel-AdminLTE/wiki/Usage>

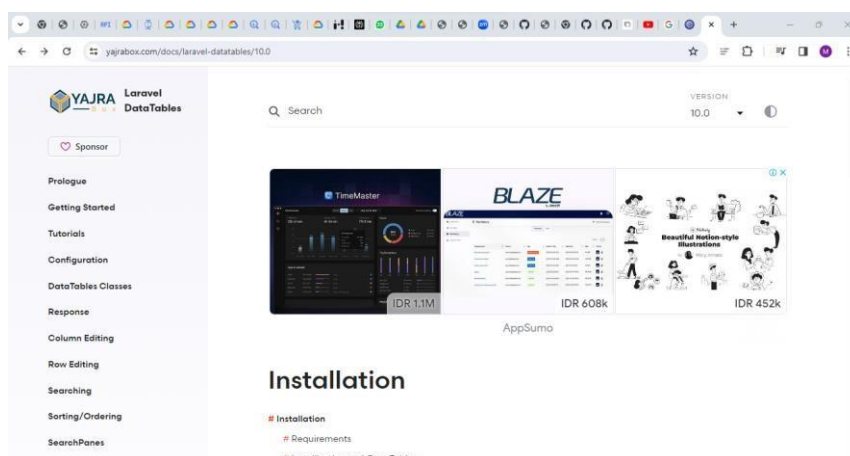
5. Kembali ke browser, menuju ke halaman awal.



Praktikum 2 – Integrasi dengan DataTables

Datatables biasa digunakan untuk menampilkan list data dengan banyak fitur seperti searching, pagination, sorting dan lain-lainnya. Dalam laravel biasa digunakan yajra datatables karena didesain agar sesuai dengan laravel.

<https://yajrabox.com/docs/laravel-datatables/10.0>





Berikut langkah-langkah praktikum

1. Install Laravel DataTables `composer require laravel/ui --dev`

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

2. Pastikan nodejs sudah terinstall, dengan perintah `npm -v`.

Jika belum install nodejs package manager dari <https://nodejs.org/dist/v20.11.1/node-v20.11.1-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`

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

6. Jalankan dengan `npm run dev`



```
C:\laragon\www\pwl_pos>npm run dev

> dev
> vite

VITE v5.1.5 ready in 250 ms

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

LARAVEL v10.46.0 plugin v1.0.2

→ APP_URL: http://localhost
```

```
VITE v5.1.6 ready in 437 ms

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

LARAVEL v11.0.7 plugin v1.0.2

→ APP_URL: http://localhost
```

7. Selanjutnya kita akan buat datatables untuk kategori `php artisan datatables:make Kategori`

```
C:\laragon\www\pwl_pos>php artisan datatables:make Kategori

INFO DataTable [C:\laragon\www\pwl_pos\app\DataTables\KategoriDataTable.php] created successfully.
```

8. Kita edit KategoriDatable untuk mengatur kolom apasaja yang ingin ditampilkan
Sesuaikan dengan kodeprogram berikut

```
<?php
```



```
namespace App\DataTables;
    use App\Models\KategoriModel; use
Illuminate\Database\Eloquent\Builder as QueryBuilder; use
Yajra\DataTables\EloquentDataTable; use
Yajra\DataTables\Html\Builder as HtmlBuilder; use
Yajra\DataTables\Html/Button; use
Yajra\DataTables\Html\Column; use
Yajra\DataTables\Html\Editor\Editor; use
Yajra\DataTables\Html\Editor\Fields; use
Yajra\DataTables\Services\DataTable;
    class KategoriDataTable extends
DataTable
    {
        /**
        * Build the DataTable class.
        *
        * @param QueryBuilder $query Results from query() method.
        */        public function dataTable(QueryBuilder $query):
EloquentDataTable
        {
            return (new
EloquentDataTable($query))
/*
                ->addColumn('action', 'kategori.action') */
                ->setRowId('id');
        }

        /**
        * Get the query source of dataTable.
        */        public function query(KategoriModel $model):
QueryBuilder
        {
            return $model-
>newQuery();
        }

        /**
        * Optional method if you want to use the html builder.
        */        public function html():
HtmlBuilder
        {
            return $this-
>builder()
                ->setTableId('kategori-table')
                ->columns($this->getColumns())
                ->minifiedAjax()
                //->dom('Bfrtip')
                ->orderBy(1)
                ->selectStyleSingle()
                ->buttons([
```



```
        Button::make('excel'),
        Button::make('csv'),
        Button::make('pdf'),
        Button::make('print'),
        Button::make('reset'),
        Button::make('reload')
    ]);
}

/**
 * Get the dataTable columns definition.
 */
public function
getColumns(): array
{
    return [
        /*      Column::computed('action')
            ->exportable(false)
            ->printable(false)
            ->width(60)
            ->addClass('text-center'), */
        Column::make('kategori_id'),
        Column::make('kategori_kode'),
        Column::make('kategori_nama'),
        Column::make('created_at'),
        Column::make('updated_at'),
    ];
}

/**
 * Get the filename for export.
 */
protected function
filename(): string
{
    return 'Kategori_' .
date('YmdHis');
}
}
```

9. Ubah kategori model, sesuaikan seperti berikut



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

10. Ubah Kategori Controller sesuaikan dengan skrip berikut:

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

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">Manage 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
23
24
```

12. Pastikan route untuk kategori sudah tersedia

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

13. Sesuaikan app layout seperti kode berikut

```
@extends('adminlte::page')

{{-- Extend and customize the browser title --}}

@section('title')
    {{ config('adminlte.title') }}
    @hasSection('subtitle') | @yield('subtitle') @endif
@stop

@vite('resources/js/app.js')

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

@section('content_header')
    @hasSection('content_header_title')
        <h1 class="text-muted">
            @yield('content_header_title')
    @endif
@stop
```



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<http://www.polinema.ac.id>

@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

{{-- Create a common footer --}}

@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 src="https://cdn.datatables.net/2.0.2/js/dataTables.js"></script>

@endpush

@stack('scripts')
```



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

```
@push('css')
```




```
<link rel="stylesheet"
href="https://cdn.datatables.net/2.0.2/css/dataTables.dataTables.css" />
<style type="text/css">

    {{-- You can add AdminLTE customizations here --}}
    /*
    .card-header {
border-bottom: none;
    }
    .card-title {
font-weight: 600;
    }
    */

</style>

@endpush
```

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         Builder::useVite();
24     }
25 }
```

15. Isikan beberapa data ke table kategori

pwl_pos.m_kategori: 2 rows total (approximately)

kategori_id	kategori_kode	kategori_nama	created_at	updated_at
1	CML	Cemilan	2024-03-09 10:30:00	(NULL)
2	MNR	Minuman Ringan	2024-03-09 13:01:39	(NULL)

```
PS C:\laragon\www\PWL_POS> php artisan db:seed --class=KategoriSeeder
```

INFO Seeding database.



				kategori_id	kategori_kode	kategori_nama	created_at	updated_at
<input type="checkbox"/>	Ubah	Salin	Hapus	1	KTG001	Elektronik	NULL	NULL
<input type="checkbox"/>	Ubah	Salin	Hapus	2	KTG002	Pakaian	NULL	NULL
<input type="checkbox"/>	Ubah	Salin	Hapus	3	KTG003	Alat Rumah Tangga	NULL	NULL
<input type="checkbox"/>	Ubah	Salin	Hapus	4	KTG004	Otomotif	NULL	NULL
<input type="checkbox"/>	Ubah	Salin	Hapus	5	KTG005	Olahraga	NULL	NULL
<input type="checkbox"/>	Ubah	Salin	Hapus	6	CML	Cemilan	NULL	NULL
<input type="checkbox"/>	Ubah	Salin	Hapus	7	MNR	Minuman Ringan	NULL	NULL

16. Databbles sudah dapat di load di `/kategori`

AdminLTE 3 | Kategori

localhost/pwl_pos/public/kategori

Home > Kagegori

Manage Kategori

Show 10 entries

Search:

Kategori Id	Kategori Kode	Kategori Nama	Created At	Updated At
1	CML	Cemilan	2024-03-09T10:30:00.000000Z	
2	MNR	Minuman Ringan	2024-03-09T13:01:39.000000Z	

Showing 1 to 2 of 2 entries

Previous 1 Next



Kategori Id	Kategori Kode	Kategori Nama	Created At	Updated At
1	KTG001	Elektronik		
2	KTG002	Pakaian		
3	KTG003	Alat Rumah Tangga		
4	KTG004	Otomotif		
5	KTG005	Olahraga		
6	CML	Cemilan		
7	MNR	Minuman Ringan		

Praktikum 3 – Membuat form kemudian menyimpan data dalam database

Langkah-langkahnya adalah sebagai berikut:

1. Menyesuaikan routing, tambahkan dua routing berikut

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

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



```
resources > views > kategori > create.blade.php > ...
1  @extends('layouts.app')
2  {{-- Customize layout sections --}}
3  @section('subtitle', 'Kategori')
4  @section('content_header_title', 'Kategori')
5  @section('content_header_subtitle', 'Create')
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">Buat kategori baru</h3>
12             </div>
13
14             <form method="post" action=" ../kategori">
15                 <div class="card-body">
16                     <div class="form-group">
17                         <label for="kodeKategori">Kode Kategori</label>
18                         <input type="text" class="form-control" id="kodeKategori" name="kodeKategori" placeholder="Kode Kategori">
19                     </div>
20                     <div class="form-group">
21                         <label for="namaKategori">Nama Kategori</label>
22                         <input type="text" class="form-control" id="namaKategori" name="namaKategori" placeholder="Nama Kategori">
23                     </div>
24                 </div>
25
26                 <div class="card-footer">
27                     <button type="submit" class="btn btn-primary">Submit</button>
28                 </div>
29             </form>
30         </div>
31     </div>
32 @endsection
```

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

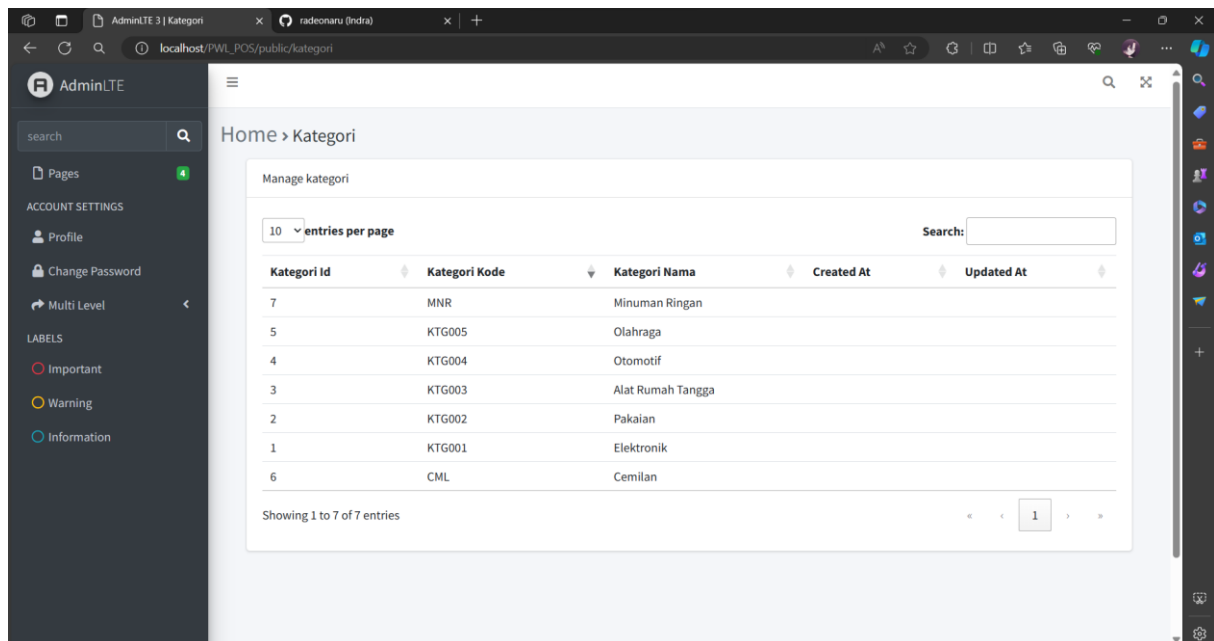
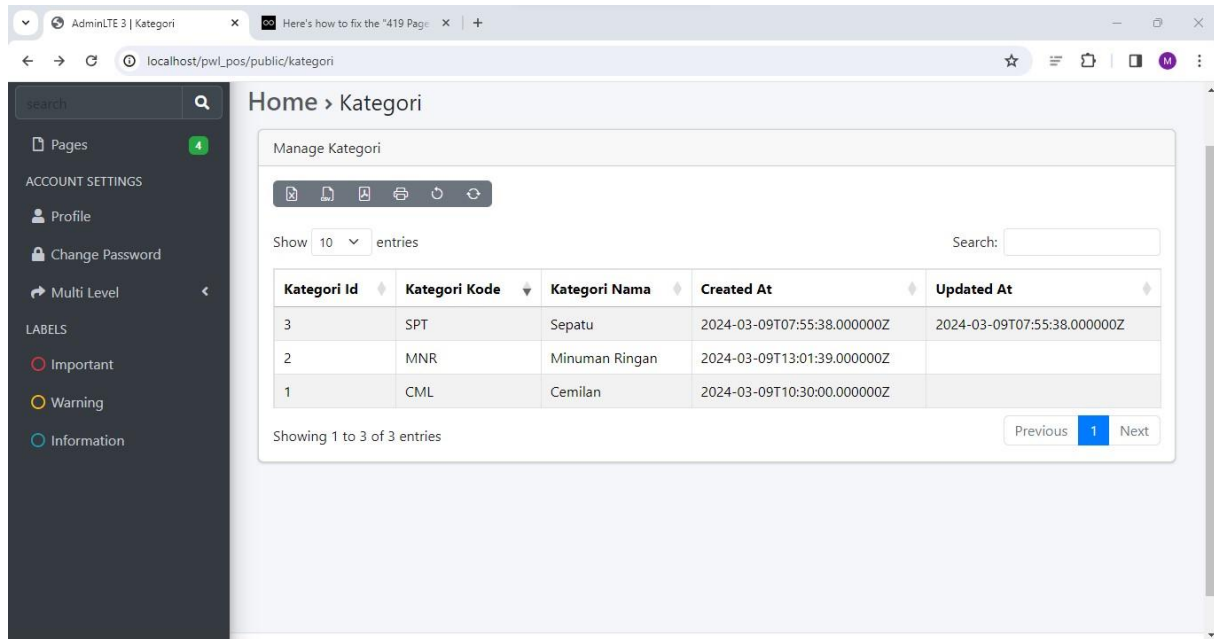
```
app > Http > Middleware > VerifyCsrfToken.php > VerifyCsrfToken > $except
1  <?php
2
3  namespace App\Http\Middleware;
4
5  use Illuminate\Foundation\Http\Middleware\VerifyCsrfToken as Middleware;
6
7  class VerifyCsrfToken extends Middleware
8  {
9      /**
10       * The URIs that should be excluded from CSRF verification.
11       *
12       * @var array<int, string>
13       */
14       protected $except = [
15           '/kategori'
16       ];
17  }
18
```



5. Akses kategori/create

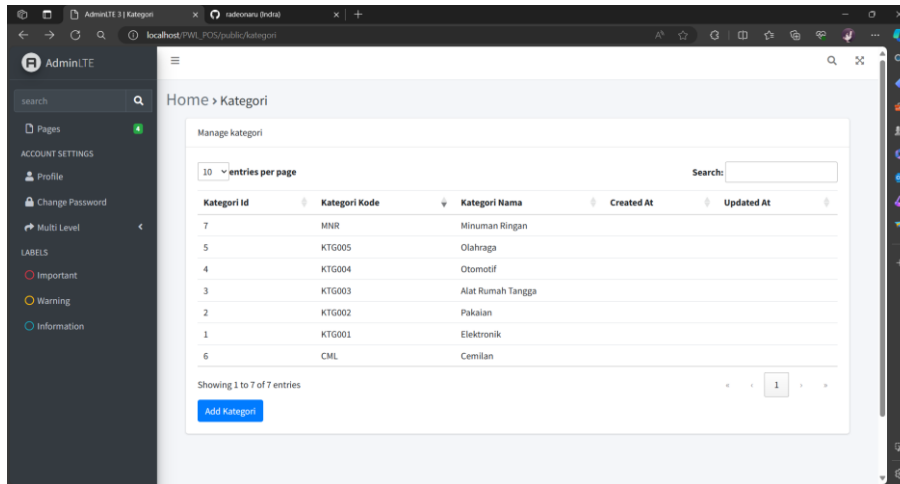
The image displays two screenshots of the AdminLTE 3 'Kategori > Create' page. The top screenshot shows the form with a blue header 'Buat kategori baru' and a 'Submit' button. The bottom screenshot shows the form with a blue header 'Buat Kategori' and a 'Submit' button. Both screenshots show the same form fields: 'Kode Kategori' (with a placeholder 'untuk makanan, contoh: MKN') and 'Nama Kategori' (with a placeholder 'Nama').

6. Halaman kategori

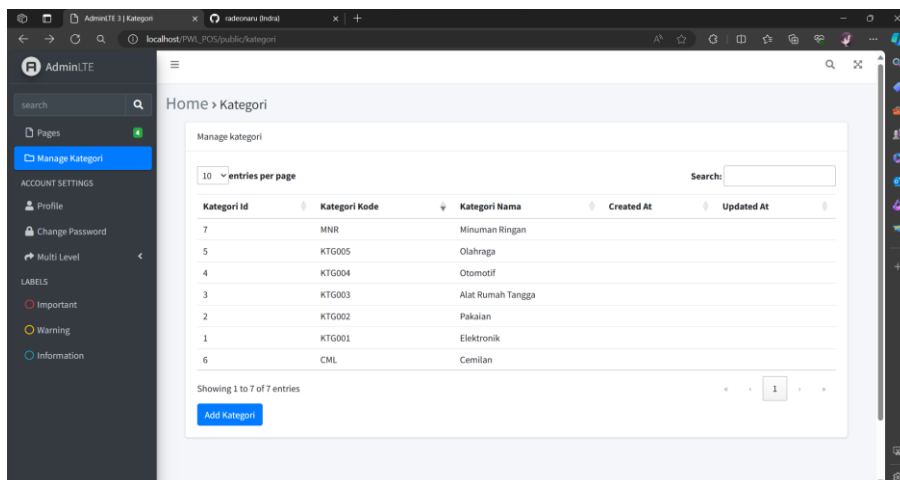


Tugas Praktikum

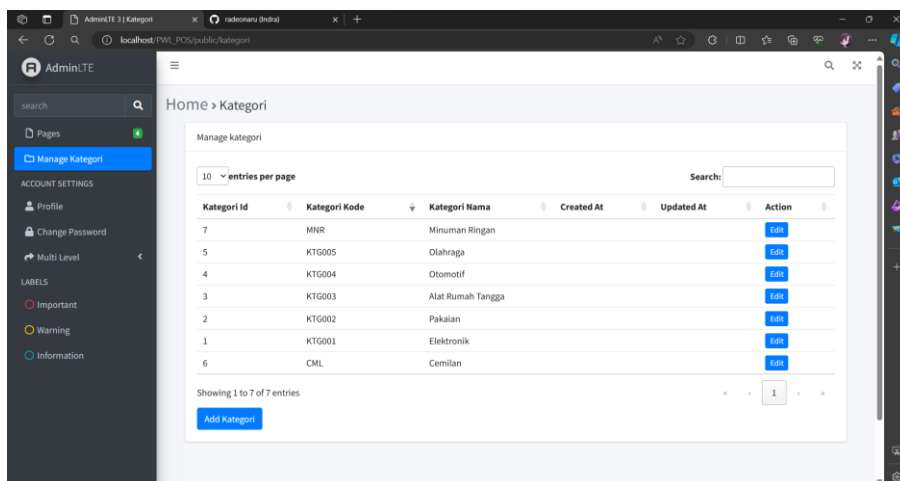
1. Tambahkan button Add di dalam manage kategori, yang mengarah ke create kategori baru



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



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





AdminLTE 3 | Edit Kategori

localhost/PWL_POS/public/kategori/edit/7

AdminLTE

search

Pages

Manage Kategori

ACCOUNT SETTINGS

Profile

Change Password

Multi Level

LABELS

Important

Warning

Information

Kategori > Edit

Edit Kategori

Kategori ID

7

Kategori Kode

DRK

Kategori Nama

Minuman Ringan

Kembali Update

AdminLTE 3 | Kategori

localhost/PWL_POS/public/kategori

AdminLTE

search

Pages

Manage Kategori

ACCOUNT SETTINGS

Profile

Change Password

Multi Level

LABELS

Important

Warning

Information

Manage kategori

10 entries per page

Search:

Kategori Id	Kategori Kode	Kategori Nama	Created At	Updated At	Action
5	KTG005	Olahraga			Edit
4	KTG004	Otomotif			Edit
3	KTG003	Alat Rumah Tangga			Edit
2	KTG002	Pakaian			Edit
1	KTG001	Elektronik			Edit
7	DRK	Minuman Ringan		2024-03-27T02:44:59.000000Z	Edit
6	CML	Cemilan			Edit

Showing 1 to 7 of 7 entries

Add Kategori

My company

Version: 1.0.0

4. Tambahkan action delete di datatables serta controllernya

AdminLTE 3 | Kategori

localhost/PWL_POS/public/kategori

AdminLTE

search

Pages

Manage Kategori

ACCOUNT SETTINGS

Profile

Change Password

Multi Level

LABELS

Important

Warning

Information

Home > Kategori

Manage kategori

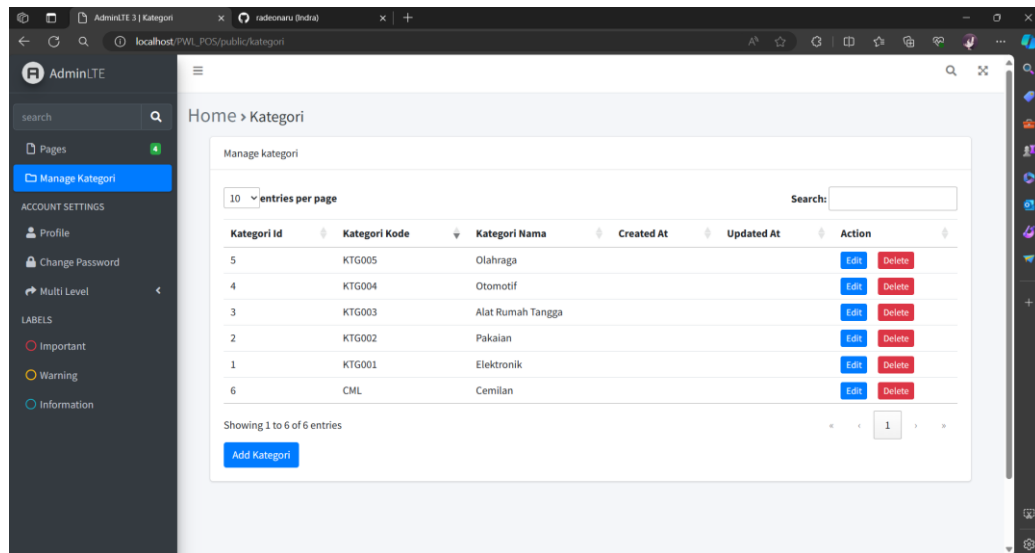
10 entries per page

Search:

Kategori Id	Kategori Kode	Kategori Nama	Created At	Updated At	Action
5	KTG005	Olahraga			Edit Delete
4	KTG004	Otomotif			Edit Delete
3	KTG003	Alat Rumah Tangga			Edit Delete
2	KTG002	Pakaian			Edit Delete
1	KTG001	Elektronik			Edit Delete
7	DRK	Minuman Ringan		2024-03-27T02:44:59.000000Z	Edit Delete
6	CML	Cemilan			Edit Delete

Showing 1 to 7 of 7 entries

Add Kategori



References

1. <https://github.com/jeroennoten/Laravel-AdminLTE/wiki/Usage>
2. <https://yajrabox.com/docs/laravel-datatables/10.0/quick-starter>