

ArsipController.php

```
<?php
```

```
namespace App\Http\Controllers;
```

```
use App\Models\Arsip;
```

```
use App\Models\JenisArsip;
```

```
use Carbon\Carbon;
```

```
use Illuminate\Http\Request;
```

```
class ArsipController extends  
Controller
```

```
{
```

```
/**
```

```
 * Display a listing of the resource.
```

```
*/
```

```
function __construct()
```

```
{
```

```
    $this->
```

```
middleware('permission:arsip-list|  
arsip-create| arsip-edit| arsip-delete',  
['only' => ['index', 'show']]);
```

```
    $this->
```

```
middleware('permission:arsip-  
create', ['only' => ['create', 'store']]);
```

```
    $this->
```

```
middleware('permission:arsip-edit',  
['only' => ['edit', 'update']]);
```

```
    $this->
```

```
middleware('permission:arsip-  
delete', ['only' => ['destroy']]);
```

```
}
```

```
function cari_arsip()
```

```
{
```

```
    return view('arsips.cariarsip');
```

```
}
```

```
function
```

```
proses_cari_arsip(Request $request)
```

```
{
```

```
    $input = $request->all();
```

```
    //ambil seluruh data arsip
```

```
    $sarsips = Arsip::all();
```

```
    $jumlah_arsip = sizeof($sarsips);
```

```

//pecahkan text menjadi token

$skatas = explode(' ',
strtolower($input['deskripsi']));

$jumlah_katas = sizeof($skatas);

$susunan_kata = [];

for ($i = 0; $i < $jumlah_katas;
$i++) {

    $kata_susun = "";

    for ($j = 0; $j < $jumlah_katas
- $i; $j++) {

        $kata_susun = $kata_susun
. ' ' . $skatas[$j];

    }

    $kata_susun =
trim($kata_susun);

    $susunan_kata[] =
$kata_susun;

}

$data_arsip_terpilih = [];

$keys = [];

foreach ($susunan_kata as $sk) {

    for ($i = 0; $i <
$jumlah_arsip; $i++) {

        $arsip = $arsips[$i];

        $judul =
strtolower($arsip['judul']);

        if (strpos($judul, $sk) !==
false) {

            if (!in_array($i, $keys)) {

                $keys[] = $i;

                $data_arsip_terpilih[]
= $arsip;

            }

        }

    }

    // dd($data_arsip_terpilih);

    // dd($jumlah_arsip);

    return view('arsips.hasil_cari',
compact('data_arsip_terpilih'));

}

```

```

function index_old()
{
    $Arsip = Arsip::latest()-
>paginate(5);

    return view('arsips.index',
compact('Arsip'))

->with('i', (request()-
>input('page', 1) - 1) * 5);
}

public function index(Request
$request)

{
    if ($request->ajax()) {

        // $data = arsips::select('*');

        $query_data = new Arsip();

        if ($request->sSearch) {

            $search_value = '%' .
$request->sSearch . '%';

            $query_data =
$query_data->where(function
($query) use ($search_value) {

                $query-
>where('jenis_arsip_id', 'like',
$search_value)

```

```

->orWhere('kode_klasifikasi', 'like',
$search_value)

->orWhere('judul',
'like', $search_value)

->orWhere('aktif_hingga', 'like',
$search_value)

->orWhere('inaktif_hingga', 'like',
$search_value)

->orWhere('file', 'like',
$search_value)

->orWhere('catatan',
'like', $search_value);

});

}

$data = $query_data-
>orderBy('judul', 'asc')->get();

return datatables()->of($data)

->addIndexColumn()

->addColumn('jenisarsip',
function (Arsip $arsip) {

    return $arsip->arsip-
>jadwal;

})

```

```

// -
>addColumn('jenisarsip2', function
(Arsip $arsip) {

    // return $arsip->arsip2-
    >kode_klasifikasi;

    // })

    ->addColumn('action',
function ($row) {

    // $btn = '<a
href="javascript:void(0)" class="edit
btn btn-primary btn-sm">View</a>';

    $btn = '<form action="" .
route('arsips.destroy', $row->id) .
"method="POST">

    <a class="btn btn-info"
href="" . route('arsips.show', $row-
>id) . "">Show</a>

    <a class="btn btn-primary"
href="" . route('arsips.edit', $row->id)
. "">Edit</a> . csrf_field() .
method_field('DELETE') . '<button
type="submit" class="btn btn-
danger">Delete</button></form>';

    return $btn;

    })

    ->rawColumns(['action'])

```

```

->make(true);

}

return view('arsips.index');

}

/**

* Show the form for creating a new
resource.

*/

public function create()
{

    $data_arsips = Arsip::all();

    $data_jenisarsips =
JenisArsip::all();

    return view('arsips.create',
compact('data_jenisarsips',
'data_arsips'));

}

/**

* Store a newly created resource in
storage.

*/

```

```

    public function store(Request $request)
    {
        $request->validate([
            'jenis_arsip_id' => 'required',
            'kode_klasifikasi' => 'required',
            'judul' => 'required',
            'aktif_hingga' => 'required',
            'inaktif_hingga' => 'required',
            'file' => 'required',
            'catatan' => 'required',
        ]);

        Arsip::create($request->all());

        return redirect()->route('arsips.index')
            ->with('success', 'arsip created successfully.');
```

```

    }

/**
 * Display the specified resource.
 */

    public function show(Arsip $arsip)
    {
        return view('arsips.show', compact('arsip'));
    }

/**
 * Show the form for editing the specified resource.
 */

    public function edit(Arsip $arsip)
    {
        $data_arsips = Arsip::all();

        return view('arsips.edit', compact('arsip', 'data_arsips'));
    }

/**
 * Update the specified resource in storage.
 */

    public function update(Request $request, Arsip $arsip)

```

```

{
    $request->validate([
        'jenis_arsip_id' => 'required',
        'kode_klasifikasi' =>
'required',

        'judul' => 'required',
        'aktif_hingga' => 'required',
        'inaktif_hingga' => 'required',
        'file' => 'required',
        'catatan' => 'required',
    ]);

    $arsip->update($request->all());

    return redirect()-
>route('arsips.index')
        ->with('success', 'Arsip
updated successfully');
}

/**
 * Remove the specified resource
from storage.
 */

    public function destroy(Arsip
$arsip)
    {
        $arsip->delete();

        return redirect()-
>route('arsips.index')
            ->with('success', 'arsip deleted
successfully');
    }

    public function
cetak_laporanArsip()
    {
        return
view('arsips.laporanArsip');
    }

    // public function
cetak_laporanArsip()

    // {

    // return
view('arsips.laporanArsip');

    // }

```

```

    }

    public function
    proses_cetak_laporanArsip(request
    $request)

    {
        $input = $request->all();

        // $sarsip = Arsip::all();

        // dd($sarsip);

        // dd($input);

        $currentDate = Carbon::now();

        $now = $currentDate-
        >format('Y-m-d');

        if ($input['status'] == 'inaktif') {

            $sarsip =
            Arsip::where('aktif_hingga', '<=',
            $now)->where('inaktif_hingga', '>=',
            $now)->get();

            return
            view('arsips.tampil_laporanInaktif',
            compact('arsip'));

            // dd($sarsip);
        }

        if ($input['status'] == 'aktif') {

            $sarsip =
            Arsip::whereDate('aktif_hingga', '>=',
            $now)->get();

            // dd($sarsip);

            return
            view('arsips.tampil_laporanAktif',
            compact('arsip'));

            // dd($sarsip);
        }

        if ($input['status'] == 'musnah') {

            $sarsip =
            Arsip::whereDate('inaktif_hingga',
            '<=', $now)->get();

            return
            view('arsips.tampil_laporanUsulmus
            nah', compact('arsip'));

            // dd($sarsip);
        }
    }

```

```

        // $arsip =
        Arsip::where('created_at', '>=',
        $input['tanggal_mulai'])-
        >where('created_at', '<=',
        $input['tanggal_selesai']->get();

        // return
        view('arsips.tampil_laporanArsip',
        compact('arsip'));

        // dd($request);
    }
}

```

Jabatancontroller.php

```
<?php
```

```

namespace App\Http\Controllers;

use App\Models\Jabatan;

use Illuminate\Http\Request;

use
Yajra\DataTables\Contracts\DataTabl
e;

class JabatanController extends
Controller
{

```

```

/**
 * Display a listing of the resource.
 */

function __construct()
{
    $this->middleware('permission:jabatan-
list| jabatan-create| jabatan-edit|
jabatan-delete', ['only' => ['index',
'show']]);

    $this->middleware('permission:jabatan-
create', ['only' => ['create', 'store']]);

    $this->middleware('permission:jabatan-
edit', ['only' => ['edit', 'update']]);

    $this->middleware('permission:jabatan-
delete', ['only' => ['destroy']]);
}

function index_old()
{
    $Jabatans = Jabatan::latest()-
>paginate(5);

```

```

        return view('Jabatans.index',
compact('Jabatans'))

        ->with('i',      (request()-
>input('page', 1) - 1) * 5);

    }

    public function index(Request
$request)

    {

        if ($request->ajax()) {

            // $data = jabatan::select('*');

            $query_data = new jabatan();

            if ($request->sSearch) {

                $search_value = '%' .
$request->sSearch . '%';

                $query_data =
$query_data->where(function
($query) use ($search_value) {

                    $query->where('nama',
'like', $search_value)

                    -

                    >orWhere('keterangan',      'like',
$search_value);

                });

            }

```

```

        $data = $query_data-
>orderBy('nama', 'asc')->get();

        return datatables()-of($data)-
>addIndexColumn()-
>addColumn('action', function ($row)
{

            //$btn = '<a
href="javascript:void(0)" class="edit
btn btn-primary btn-sm">View</a>';

            $btn = '<form action="" .
route('jabatans.destroy', $row->id) . ""
method="POST">

                <a class="btn btn-info" href="" .
route('jabatans.show', $row->id) .
"">Show</a>

                <a class="btn btn-primary" href=""
. route('jabatans.edit', $row->id) .
"">Edit</a>' . csrf_field() .
method_field('DELETE') . '

                <button type="submit" class="btn
btn-
danger">Delete</button></form>;

            return $btn;

        })

        ->rawColumns(['action'])

        ->make(true);

```

<pre> } return view('jabatans.index'); } /** * Show the form for creating a new resource. */ public function create() { return view('Jabatans.create'); } /** * Store a newly created resource in storage. */ public function store(Request \$request) { \$request->validate(['nama' => 'required', 'keterangan' => 'required', </pre>	<pre>]); Jabatan::create(\$request->all()); return redirect()- >route('jabatans.index') ->with('success', 'jabatans created successfully.');</pre> <pre> } /** * Display the specified resource. */ public function show(Jabatan \$jabatan) { return view('Jabatans.show', compact('jabatan')); } /** * Show the form for editing the specified resource. */ </pre>
---	--

```

    public function edit(Jabatan
$jabatan)

    {

        return view('Jabatans.edit',
compact('jabatan'));

    }

/**

* Update the specified resource in
storage.

*/

public function update(Request
$request, Jabatan $jabatan)

{

    $request->validate([

        'nama' => 'required',

        'keterangan' => 'required',

    ]);

    $jabatan->update($request-
>all());

    return redirect()-
>route('jabatans.index')

    ->with('success', 'jabatan
updated successfully');

}

/**

* Remove the specified resource
from storage.

*/

public function destroy(Jabatan
$jabatan)

{

    $jabatan->delete();

    return redirect()-
>route('jabatans.index')

    ->with('success', 'Jabatan
deleted successfully');

}

```