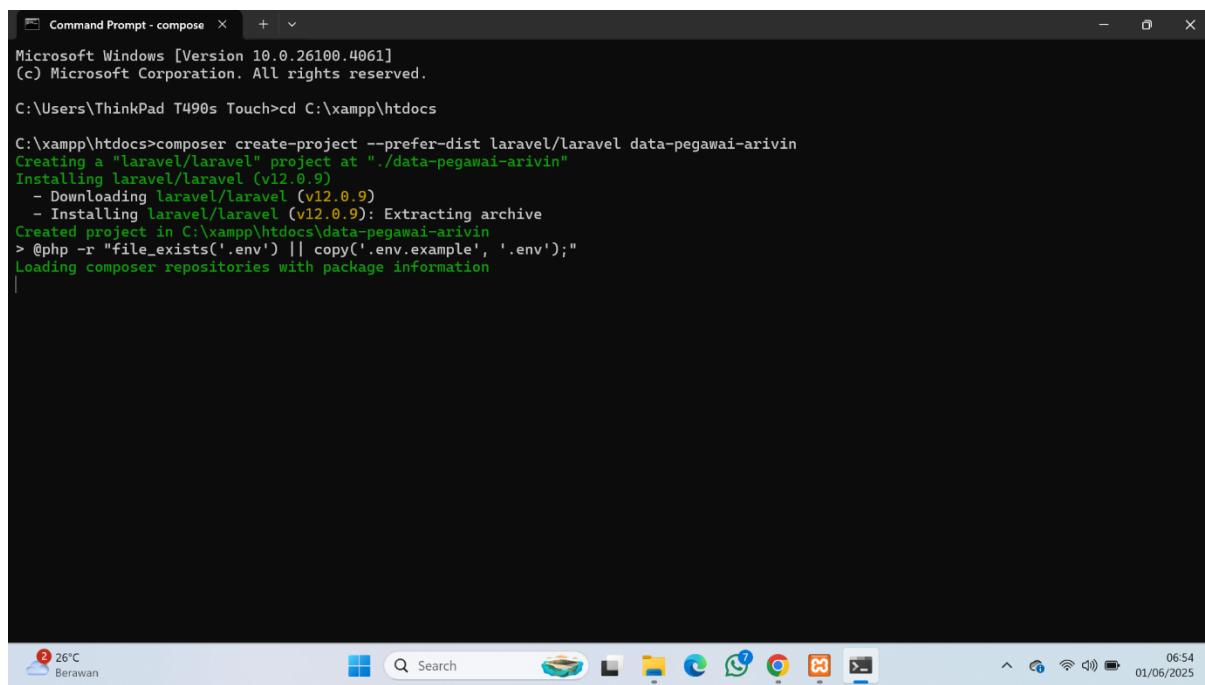


NAMA = NUR ARIVIN UMAR
NIM = G.231.23.0139
MATKUL = P. FRAMEWORK
TUGAS = CRUD DATA PEGAWAI
LINK GITHUB = <https://github.com/arivinumar/data-pegawai>

Langkah langkahnya :

- Buat project laravel



```
Command Prompt - compose  X + ▾ Microsoft Windows [Version 10.0.26100.4061]
(c) Microsoft Corporation. All rights reserved.

C:\xampp\htdocs>composer create-project --prefer-dist laravel/laravel data-pegawai-arivin
Creating a "laravel/laravel" project at "./data-pegawai-arivin"
Installing laravel/laravel (v12.0.9)
- Downloading laravel/laravel (v12.0.9)
- Installing laravel/laravel (v12.0.9): Extracting archive
Created project in C:\xampp\htdocs\data-pegawai-arivin
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
|
```

The screenshot shows a Windows Command Prompt window titled "Command Prompt - compose". The command entered is "composer create-project --prefer-dist laravel/laravel data-pegawai-arivin". The output shows the creation of a new Laravel project named "data-pegawai-arivin" in the current directory. The terminal also displays the creation of a ".env" file from ".env.example". The window has a standard Windows title bar and taskbar at the bottom.

```
Command Prompt x + v - o x
laravel/pail ..... DONE
laravel/sail ..... DONE
laravel/tinker ..... DONE
nesbot/carbon ..... DONE
nunomaduro/collision ..... DONE
nunomaduro/termwind ..... DONE

80 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].  

No security vulnerability advisories found.  

> @php artisan key:generate --ansi
[INFO] Application key set successfully.  

> @php -r "file_exists('database/database.sqlite') || touch('database/database.sqlite');"  

> @php artisan migrate --graceful --ansi
[INFO] Preparing database.  

Creating migration table ..... 14.81ms DONE
[INFO] Running migrations.  

0001_01_01_000000_create_users_table ..... 24.39ms DONE
0001_01_01_000001_create_cache_table ..... 7.35ms DONE
0001_01_01_000002_create_jobs_table ..... 19.14ms DONE  

C:\xampp\htdocs>
```



```
Command Prompt - php arti: x + v - o x
[INFO] Application key set successfully.  

> @php -r "file_exists('database/database.sqlite') || touch('database/database.sqlite');"  

> @php artisan migrate --graceful --ansi
[INFO] Preparing database.  

Creating migration table ..... 14.81ms DONE
[INFO] Running migrations.  

0001_01_01_000000_create_users_table ..... 24.39ms DONE
0001_01_01_000001_create_cache_table ..... 7.35ms DONE
0001_01_01_000002_create_jobs_table ..... 19.14ms DONE  

C:\xampp\htdocs>code.  

'code.' is not recognized as an internal or external command,  

operable program or batch file.  

C:\xampp\htdocs>code .  

C:\xampp\htdocs>cd data-pegawai-arivin  

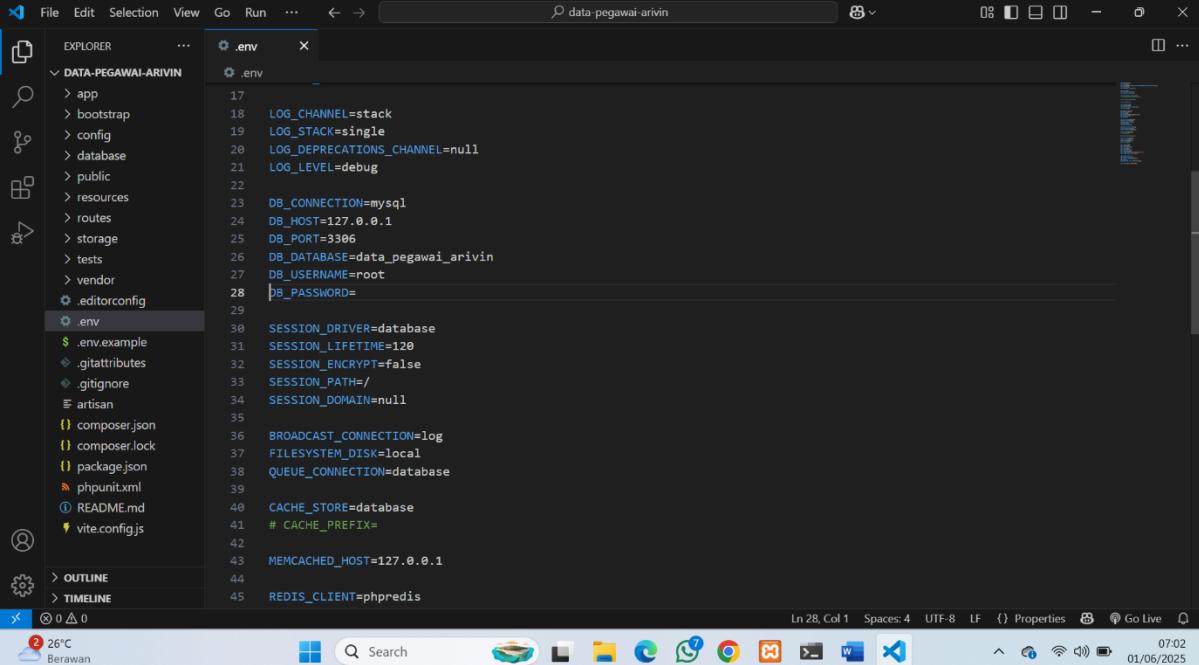
C:\xampp\htdocs\data-pegawai-arivin>code .  

C:\xampp\htdocs\data-pegawai-arivin>php artisan serve
[INFO] Server running on [http://127.0.0.1:8000].
Press Ctrl+C to stop the server  

C:\xampp\htdocs>
```



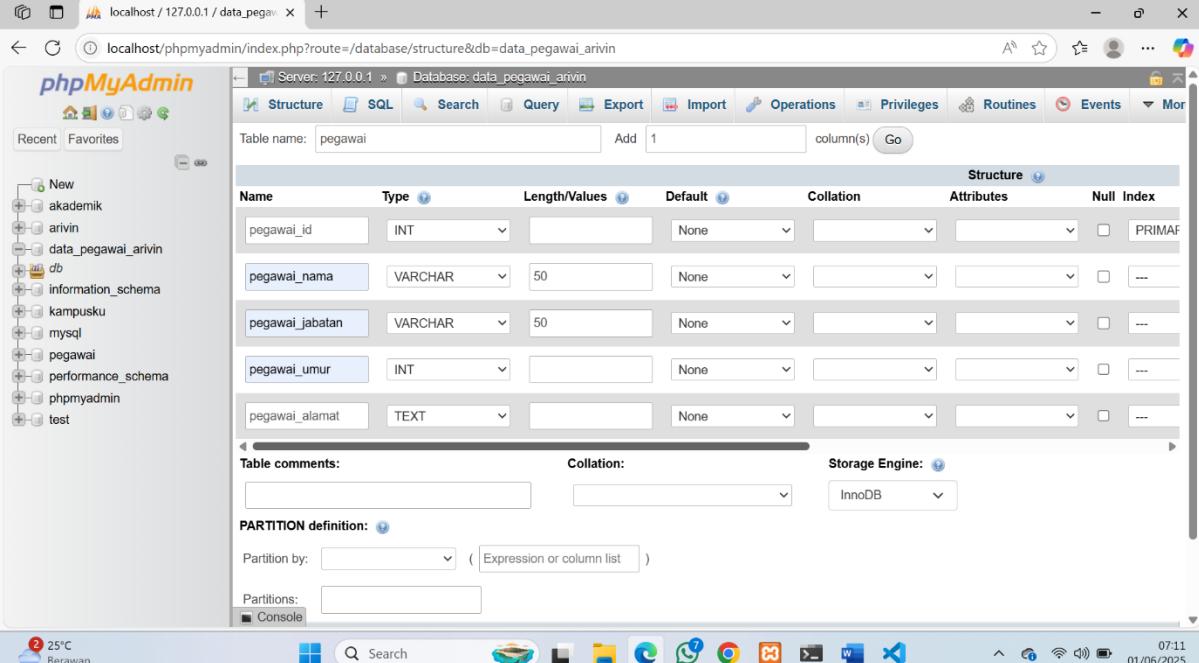
➤ Konfigurasi database



The screenshot shows the VS Code interface with the .env file open in the editor. The file contains environment variables for a MySQL database connection:

```
LOG_CHANNEL=stack
LOG_STACK=single
LOG_DEPRECATIONS_CHANNEL=null
LOG_LEVEL=debug
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=data_pegawai_arivin
DB_USERNAME=root
DB_PASSWORD=
SESSION_DRIVER=database
SESSION_LIFETIME=120
SESSION_ENCRYPT=false
SESSION_PATH=/
SESSION_DOMAIN=null
BROADCAST_CONNECTION=log
FILESYSTEM_DISK=local
QUEUE_CONNECTION=database
CACHE_STORE=database
# CACHE_PREFIX=
MEMCACHED_HOST=127.0.0.1
REDIS_CLIENT=phpredis
```

➤ Buat Table di Mysql



The screenshot shows the phpMyAdmin interface for creating a new table named 'pegawai'. The table structure is defined as follows:

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index
pegawai_id	INT		None			□	PRIMAR
pegawai_nama	VARCHAR	50	None			□	---
pegawai_jabatan	VARCHAR	50	None			□	---
pegawai_umur	INT		None			□	---
pegawai_alamat	TEXT		None			□	---

Table comments: Collation: Storage Engine: InnoDB

PARTITION definition: Partition by: Expression or column list Partitions: Console

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	pegawai_id	int(11)	utf8mb4_general_ci	No	None		AUTO_INCREMENT		Change Drop More
2	pegawai_nama	varchar(50)	utf8mb4_general_ci	No	None				Change Drop More
3	pegawai_jabatan	varchar(50)	utf8mb4_general_ci	No	None				Change Drop More
4	pegawai_umur	int(11)		No	None				Change Drop More
5	pegawai_alamat	text	utf8mb4_general_ci	No	None				Change Drop More

Indexes

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Rename Drop	PRIMARY	BTREE	Yes	No	pegawai_id	0	A	No	

Create an index on columns [Go](#)

➤ Route Pegawai

```

routes > web.php > ...
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4
5 Route::get('/', function () {
6     return view('welcome');
7 });
8
9 Route::get('/Pegawai', [PegawaiController::class, 'index']);

```

➤ PegawaiController.php

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "DATA-PEGAWAI-ARIVIN". The "routes" folder contains "web.php".
- Editor:** The "web.php" file is open, displaying the following PHP code:

```
use Illuminate\Support\Facades\Route;
Route::get('/', function () {
    return view('welcome');
});
Route::get('/Pegawai', [PegawaiController::class, 'index']);
```

- Terminal:** The terminal shows the command `php artisan make:controller PegawaiController` being run, with the response: `[INFO] Controller [C:\xampp\htdocs\data-pegawai-arivin\app\Http\Controllers\PegawaiController.php] created successfully.`
- Bottom Status Bar:** Shows the current file is "web.php", line 9, column 61, with 4 spaces, using UTF-8 encoding, and the date/time as 07:21 on 01/06/2025.

➤ Migrate

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "DATA-PEGAWAI-ARIVIN". The "routes" folder contains "web.php".
- Editor:** The "web.php" file is open, displaying the same code as before.
- Terminal:** The terminal shows the command `php artisan migrate` being run, with the output:

```
INFO Preparing database.
Creating migration table ..... 18.78ms DONE
INFO Running migrations.
0001_01_01_00000_create_users_table 63.95ms DONE
0001_01_01_00001_create_cache_table Open folder in new window (ctrl + click) 16.24ms DONE
0001_01_01_00002_create_jobs_table 56.90ms DONE
```

- Bottom Status Bar:** Shows the current file is "web.php", line 9, column 61, with 4 spaces, using UTF-8 encoding, and the date/time as 07:22 on 01/06/2025.

The screenshot shows a Microsoft Visual Studio Code (VS Code) interface. The main area displays the code for a PHP controller:

```
<?php  
namespace App\Http\Controllers;  
use Illuminate\Http\Request;  
use Illuminate\Support\Facades\DB;  
class PegawaiController extends Controller  
{  
    public function index()  
    {  
        $pegawai = DB::table('pegawai')->get();  
        return view('index',['pegawai' => $pegawai]);  
    }  
}
```

The left sidebar (Explorer) shows the project structure:

- DATA-PEGAWAI-ARVIN
- app
 - Http \ Controllers
 - Controller.php
 - PegawaiController.php
 - Models
 - Providers
 - bootstrap
 - config
 - database
 - public
 - resources
 - routes
 - console.php
 - web.php
 - storage
 - tests
 - vendor
 - .editorconfig
 - .env
 - .env.example
 - .gitattributes
 - .gitignore
 - artisan
- PROBLEMS
- OUTPUT
- DEBUG CONSOLE
- TERMINAL
- PORTS

No problems have been detected in the workspace.

Bottom status bar: Ln 13, Col 54 | Spaces: 4 | UTF-8 | LF | PHP | Go Live | 07:36 | 01/06/2025

➤ Index.blade.php

The screenshot shows a Windows desktop environment with a code editor and a web browser.

Code Editor (VS Code):

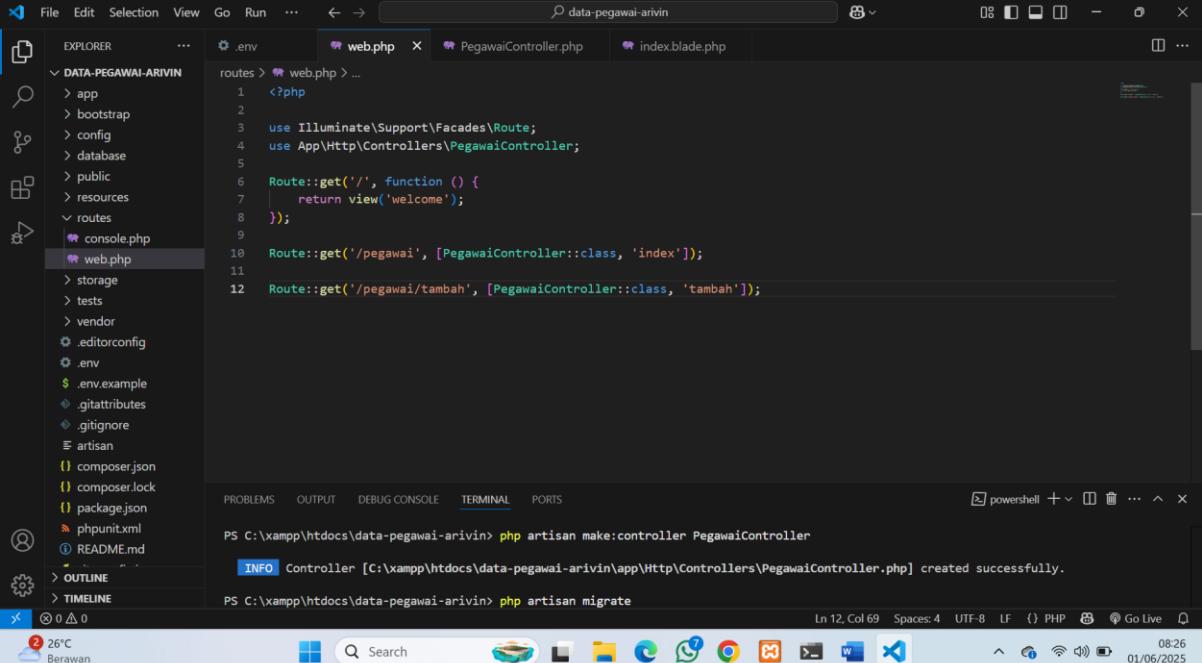
- File Path:** resources > views > index.blade.php
- Code Content:**

```
resources > views > index.blade.php > html
1 <html lang="en">
2   <body>
3     <br/>
4     <table border="1">
5       <tr>
6         <th>Nama</th>
7         <th>Jabatan</th>
8         <th>Umur</th>
9         <th>Alamat</th>
10        <th>Opsi</th>
11      </tr>
12      @foreach($pegawai as $p)
13      <tr>
14        <td>{{ $p->pegawai_nama }}</td>
15        <td>{{ $p->pegawai_jabatan }}</td>
16        <td>{{ $p->pegawai_umur }}</td>
17        <td>{{ $p->pegawai_alamat }}</td>
18        <td>
19          <a href="/pegawai/edit/{{ $p->pegawai_id }}>Edit</a>
20          <a href="/pegawai/hapus/{{ $p->pegawai_id }}>Hapus</a>
21        </td>
22      </tr>
23    @endforeach
24  </table>
25 </body>
26 </html>
```

Browser Window:

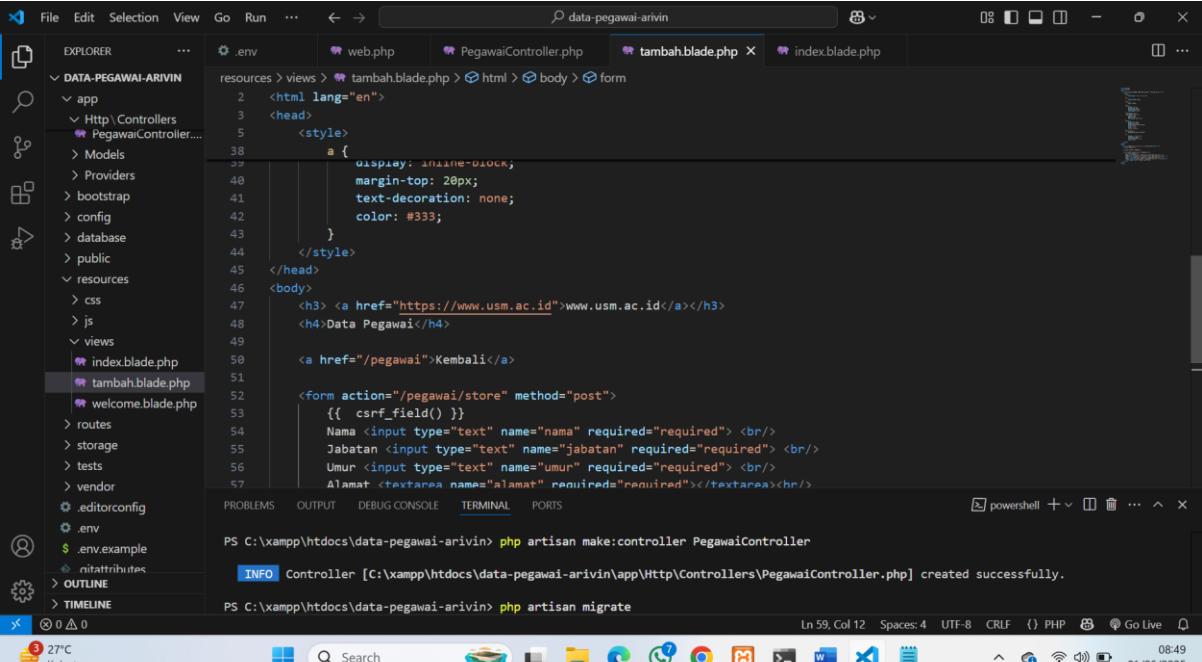
- Title Bar:** Tutorial Membuat CRUD Pada Lar
- Address Bar:** localhost:8000/pegawai
- Content:** Data Pegawai
- Links:** +Tambah Pegawai Baru
- Table Headers:** Nama, Jabatan, Umur, Alamat, Opsi

➤ Route tambah



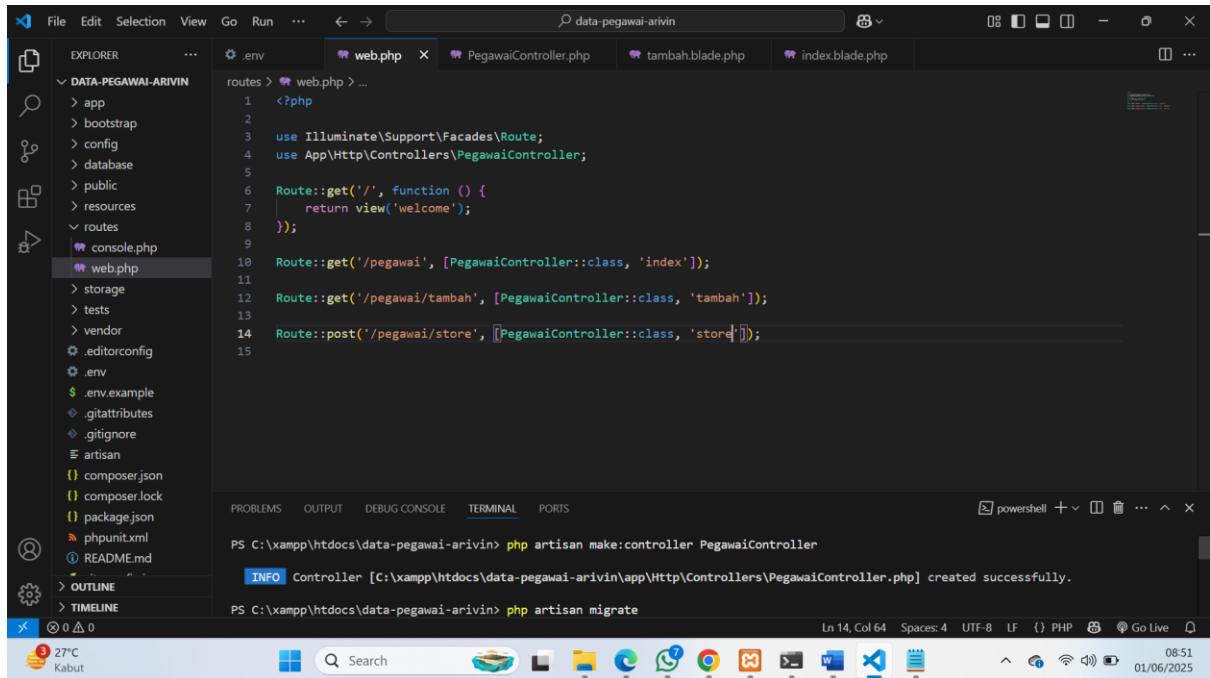
```
routes > web.php > ...
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use App\Http\Controllers\PegawaiController;
5
6  Route::get('/', function () {
7      return view('welcome');
8  });
9
10 Route::get('/pegawai', [PegawaiController::class, 'index']);
11
12 Route::get('/pegawai/tambah', [PegawaiController::class, 'tambah']);
```

➤ Tambah.blade.php



```
resources > views > tambah.blade.php > html > body > form
2   <html lang="en">
3     <head>
4       <style>
5         a {
6             display: inline-block;
7             margin-top: 20px;
8             text-decoration: none;
9             color: #333;
10        }
11       </style>
12     </head>
13     <body>
14       <h3> <a href="https://www.usm.ac.id">www.usm.ac.id</a></h3>
15       <h4>Data Pegawai</h4>
16
17       <a href="/pegawai">Kembali</a>
18
19       <form action="/pegawai/store" method="post">
20         {{ csrf_field() }}
21         Nama <input type="text" name="nama" required="required"> <br/>
22         Jabatan <input type="text" name="jabatan" required="required"> <br/>
23         Umur <input type="text" name="umur" required="required"> <br/>
24         Alamat <textarea name="alamat" required="required"></textarea><br/>
```

➤ Route store



The screenshot shows the Visual Studio Code interface with the following details:

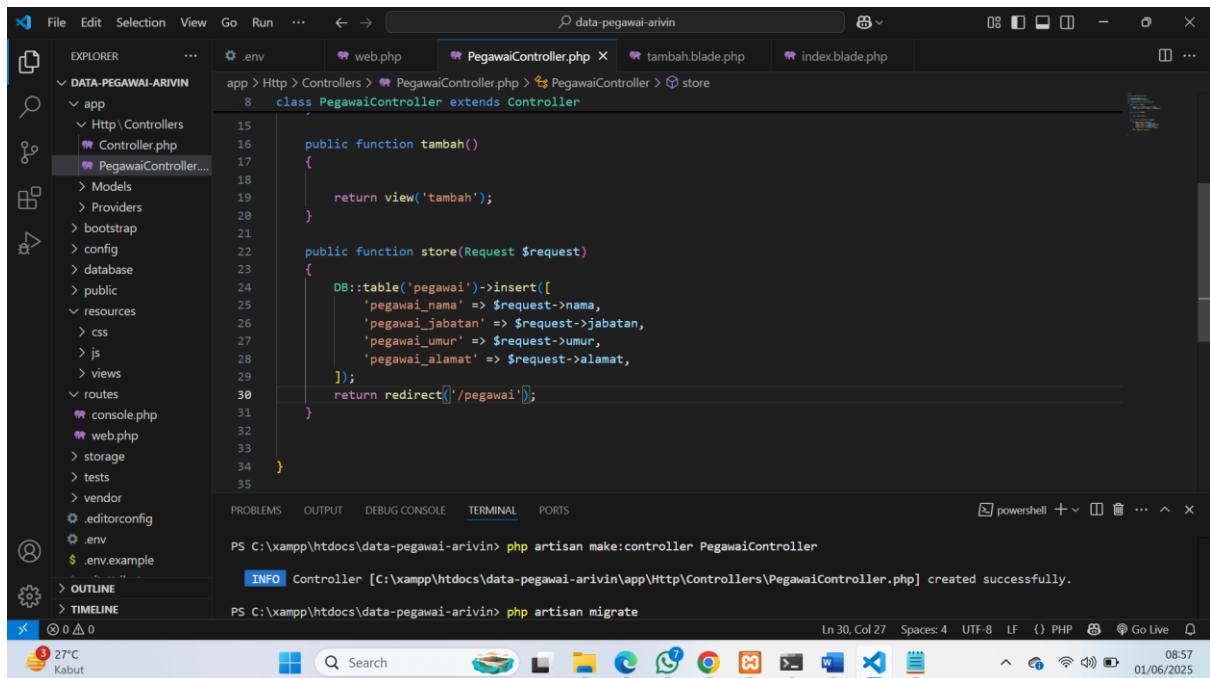
- File Explorer:** Shows the project structure under "DATA-PEGAWAI-ARIVIN". The "routes" folder contains "web.php". Other files like ".env", "app", "bootstrap", "config", "database", "public", "resources", "console.php", ".editorconfig", ".env.example", ".gitattributes", ".gitignore", "artisan", "composer.json", "composer.lock", "package.json", "phpunit.xml", and "README.md" are also listed.
- Code Editor:** The "routes/web.php" file is open. It contains PHP code defining routes for a Laravel application. The code includes:

```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\PegawaiController;
5
6 Route::get('/', function () {
7     return view('welcome');
8 });
9
10 Route::get('/pegawai', [PegawaiController::class, 'index']);
11
12 Route::get('/pegawai/tambah', [PegawaiController::class, 'tambah']);
13
14 Route::post('/pegawai/store', [PegawaiController::class, 'store']);
```
- Terminal:** The terminal shows command-line output from running artisan commands:

```
PS C:\xampp\htdocs\data-pegawai-arivin> php artisan make:controller PegawaiController
[INFO] Controller [C:\xampp\htdocs\data-pegawai-arivin\app\Http\Controllers\PegawaiController.php] created successfully.

PS C:\xampp\htdocs\data-pegawai-arivin> php artisan migrate
```
- System Tray:** Shows a weather icon for "Kabut" at 27°C.

➤ Methode store



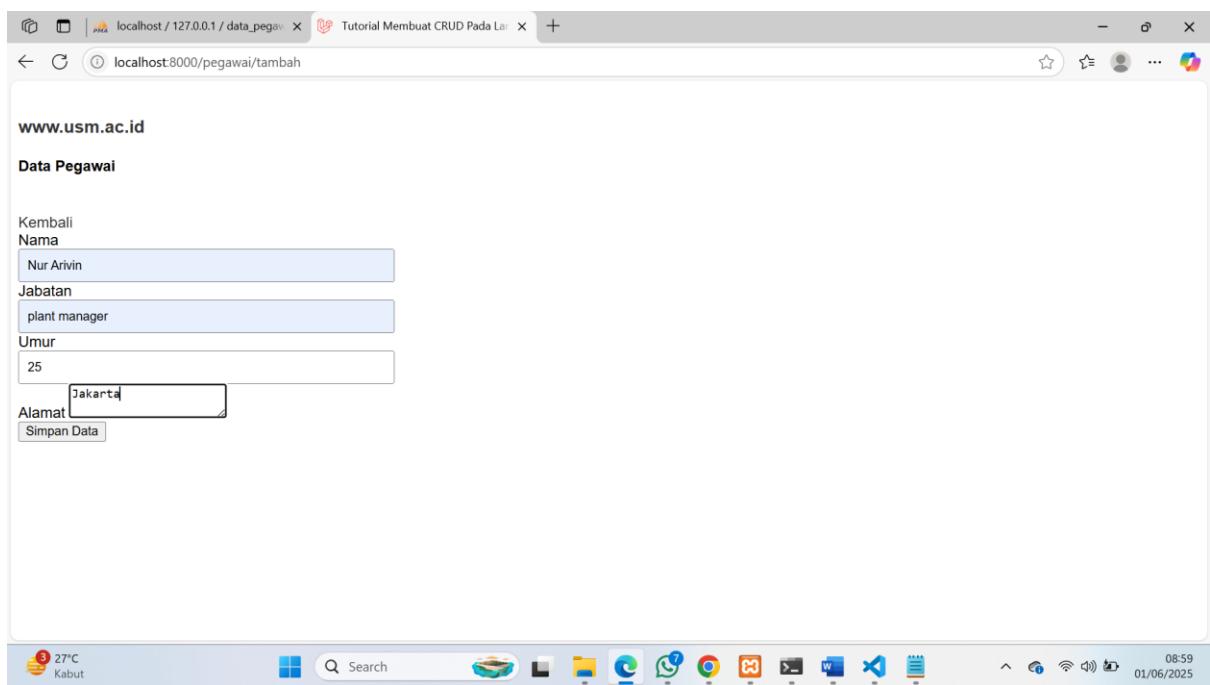
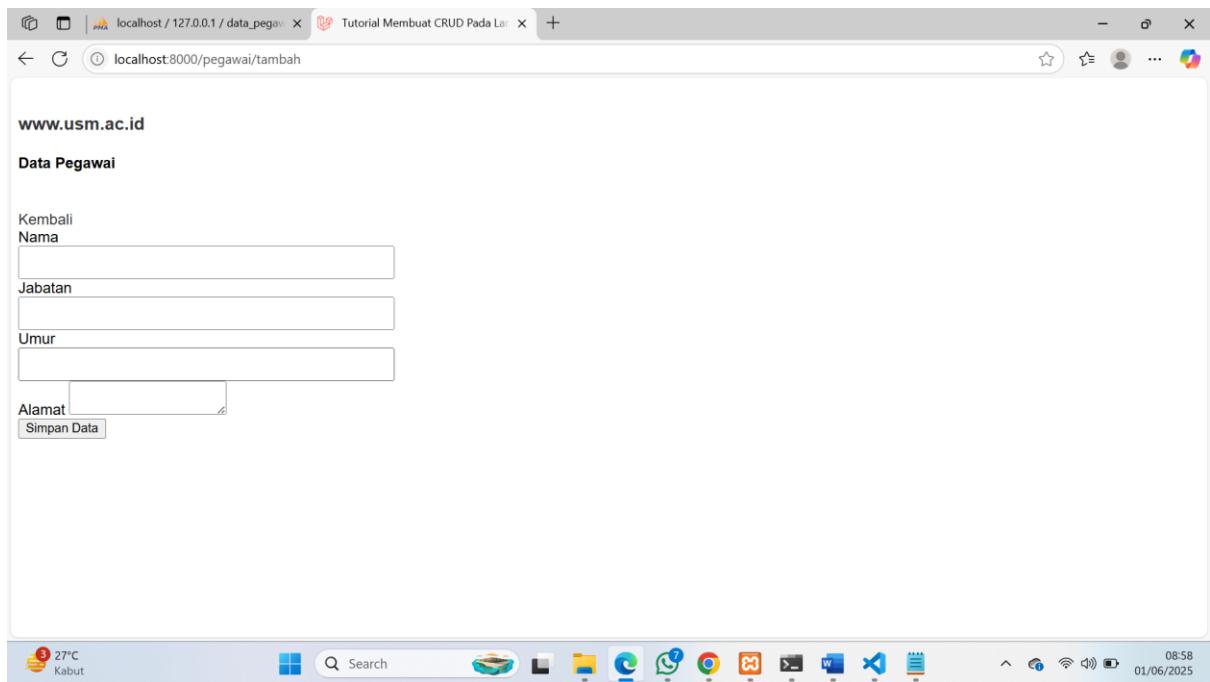
The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "DATA-PEGAWAI-ARIVIN". The "Http\Controllers" folder contains "Controller.php" and "PegawaiController.php". Other files like "Models", "Providers", "bootstrap", "config", "database", "public", "resources", "css", "js", "views", "routes", "console.php", "web.php", "storage", "tests", "vendor", ".editorconfig", ".env", and ".env.example" are listed.
- Code Editor:** The "Http\Controllers\PegawaiController.php" file is open. It contains PHP code for a controller class:

```
15 class PegawaiController extends Controller
16
17     public function tambah()
18     {
19
20         return view('tambah');
21     }
22
23     public function store(Request $request)
24     {
25
26         DB::table('pegawai')->insert([
27             'pegawai_nama' => $request->nama,
28             'pegawai_jabatan' => $request->jabatan,
29             'pegawai_umur' => $request->umur,
30             'pegawai_alamat' => $request->alamat,
31         ]);
32
33         return redirect('/pegawai');
34     }
35 }
```
- Terminal:** The terminal shows command-line output from running artisan commands:

```
PS C:\xampp\htdocs\data-pegawai-arivin> php artisan make:controller PegawaiController
[INFO] Controller [C:\xampp\htdocs\data-pegawai-arivin\app\Http\Controllers\PegawaiController.php] created successfully.

PS C:\xampp\htdocs\data-pegawai-arivin> php artisan migrate
```
- System Tray:** Shows a weather icon for "Kabut" at 27°C.



Data Pegawai

+ Tambah Pegawai Baru

Nama	Jabatan	Umur	Alamat	Opsi
Nur Arivin	plant manager	25	Jakarta	Edit Hapus

27°C Kabut

Search

01/06/2025

➤ Route edit

```
routes > web.php > ...
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use App\Http\Controllers\PegawaiController;
5
6  Route::get('/', function () {
7      return view('welcome');
8 });
9
10 Route::get('/pegawai', [PegawaiController::class, 'index']);
11
12 Route::get('/pegawai/tambah', [PegawaiController::class, 'tambah']);
13
14 Route::post('/pegawai/store', [PegawaiController::class, 'store']);
15
16 Route::get('/pegawai/edit{id}', [PegawaiController::class, 'edit']);
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\xampp\htdocs\data-pegawai-arivin>

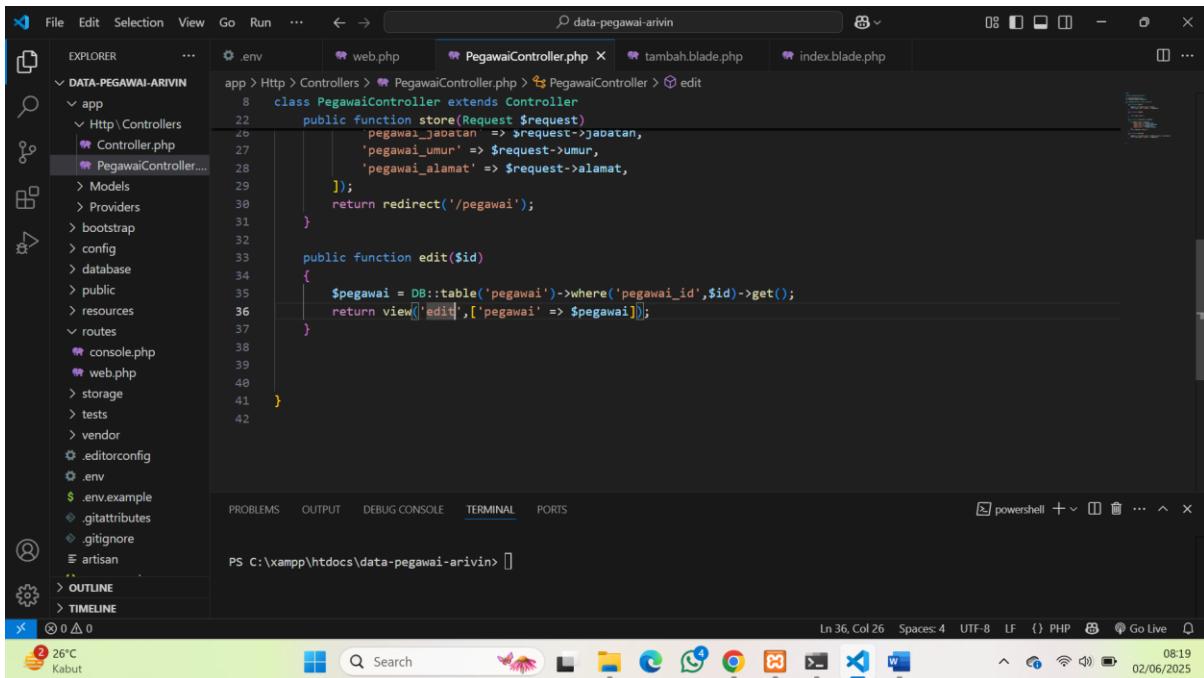
Ln 16, Col 66 Spaces: 4 UTF-8 LF PHP Go Live

26°C Kabut

08:16

02/06/2025

➤ Methode edit

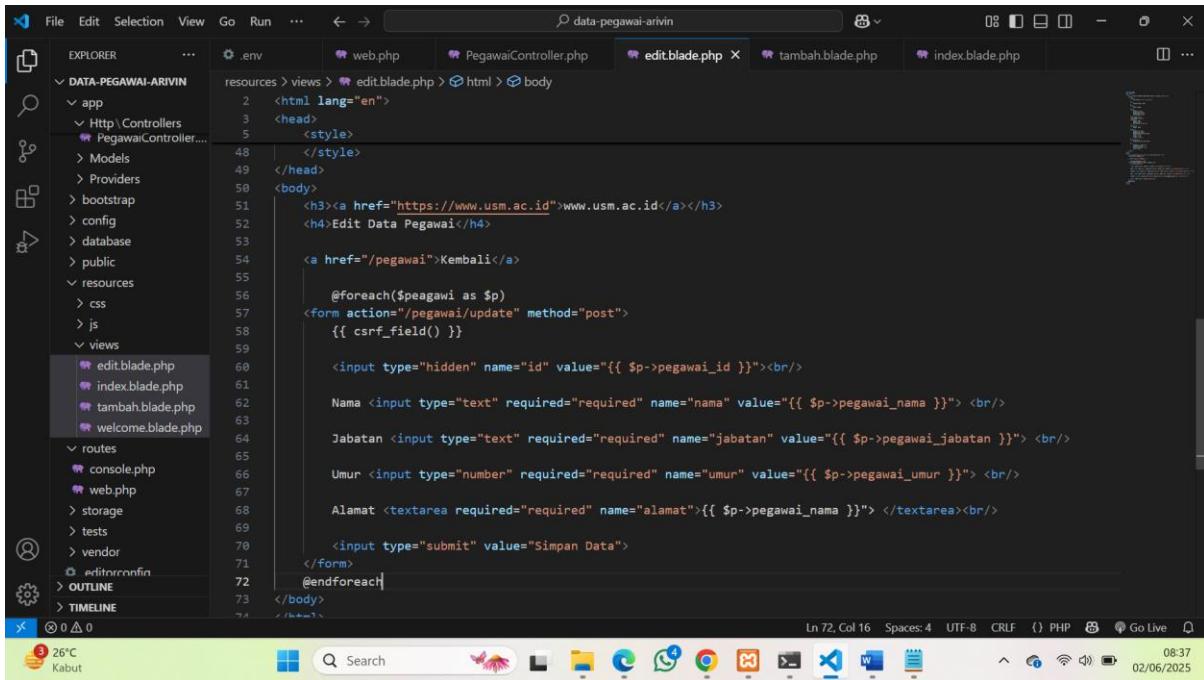


The screenshot shows the Visual Studio Code interface with the file `PegawaiController.php` open in the editor. The code defines a `PegawaiController` class extending `Controller`. It contains two methods: `store` and `edit`. The `store` method handles form submissions, setting `pegawai_jabatan`, `pegawai_umur`, and `pegawai_alamat` from the request to the database. The `edit` method retrieves a `pegawai` record by `pegawai_id` and returns a view named `edit` with the record pre-filled.

```
class PegawaiController extends Controller
{
    public function store(Request $request)
    {
        $pegawai_jabatan => $request->jabatan,
        'pegawai_umur' => $request->umur,
        'pegawai_alamat' => $request->alamat,
    ];
    return redirect('/pegawai');
}

public function edit($id)
{
    $pegawai = DB::table('pegawai')->where('pegawai_id',$id)->get();
    return view('edit',['pegawai' => $pegawai]);
}
```

➤ Edit.blade.php



The screenshot shows the Visual Studio Code interface with the file `edit.blade.php` open in the editor. The code is a Blade template for editing a `pegawai`. It includes a header with a backlink to the homepage and a link to go back. The main part of the template displays a form for updating the `pegawai` record. It uses a `foreach` loop to iterate over the `$pegawai` array, which contains the current record. The form has fields for `Nama`, `Jabatan`, `Umur`, and `Alamat`, each with its value set to the corresponding database field. A submit button is at the bottom of the form.

```
<html lang="en">
<head>
    <style>
        </style>
</head>
<body>
    <h3><a href="https://www.usm.ac.id">www.usm.ac.id</a></h3>
    <h4>Edit Data Pegawai</h4>

    <a href="/pegawai">Kembali</a>

    @foreach($pegawai as $p)
    <form action="/pegawai/update" method="post">
        {{ csrf_field() }}

        <input type="hidden" name="id" value="{{ $p->pegawai_id }}><br/>

        Nama <input type="text" required="required" name="nama" value="{{ $p->pegawai_nama }}><br/>

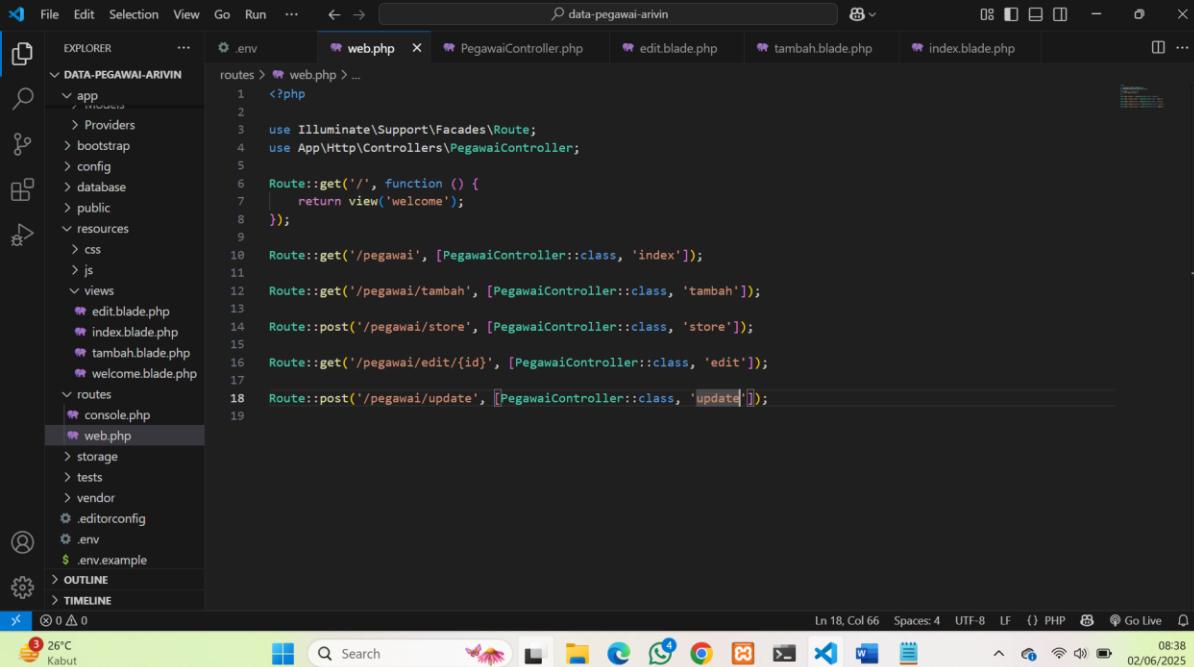
        Jabatan <input type="text" required="required" name="jabatan" value="{{ $p->pegawai_jabatan }}><br/>

        Umur <input type="number" required="required" name="umur" value="{{ $p->pegawai_umur }}><br/>

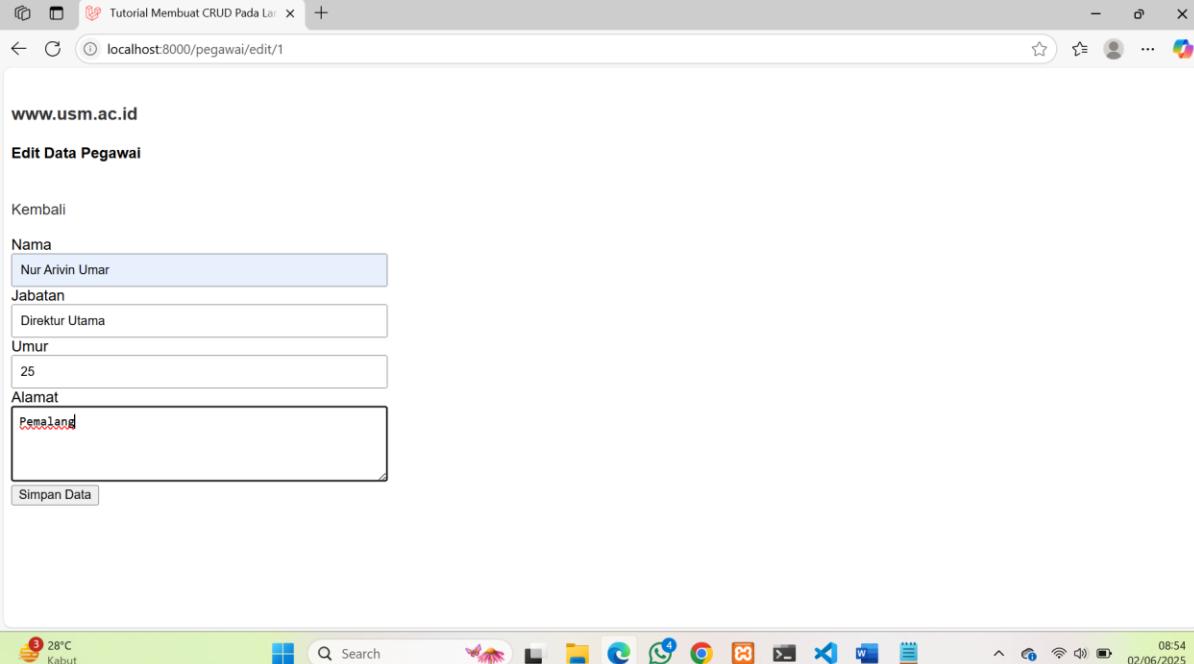
        Alamat <textarea required="required" name="alamat">{{ $p->pegawai_nama }}</textarea><br/>

        <input type="submit" value="Simpan Data">
    </form>
    @endforeach
</body>
```

➤ Route update



```
routes > web.php > ...
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4  use App\Http\Controllers\PegawaiController;
5
6  Route::get('/', function () {
7      return view('welcome');
8  });
9
10 Route::get('/pegawai', [PegawaiController::class, 'index']);
11
12 Route::get('/pegawai/tambah', [PegawaiController::class, 'tambah']);
13
14 Route::post('/pegawai/store', [PegawaiController::class, 'store']);
15
16 Route::get('/pegawai/edit/{id}', [PegawaiController::class, 'edit']);
17
18 Route::post('/pegawai/update', [PegawaiController::class, 'update']);
19
```



localhost:8000/pegawai/edit/1

www.usm.ac.id

Edit Data Pegawai

Kembali

Nama: Nur Arivin Umar

Jabatan: Direktur Utama

Umur: 25

Alamat: Pemalang

Data Pegawai

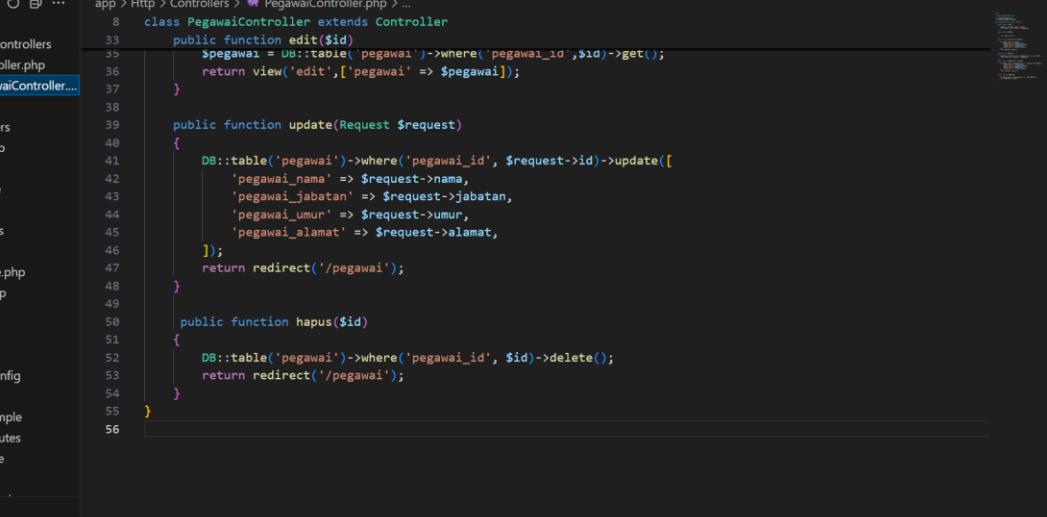
+ Tambah Pegawai Baru

Nama	Jabatan	Umur	Alamat	Opsi
Nur Arivin Umar	Direktur Utama	25	Pemalang	Edit Hapus

➤ Route hapus

```
routes > web.php > ...
4 use App\Http\Controllers\PegawaiController;
5
6 Route::get('/', function () {
7     return view('welcome');
8 });
9
10 Route::get('/pegawai', [PegawaiController::class, 'index']);
11
12 Route::get('/pegawai/tambah', [PegawaiController::class, 'tambah']);
13
14 Route::post('/pegawai/store', [PegawaiController::class, 'store']);
15
16 Route::get('/pegawai/edit{id}', [PegawaiController::class, 'edit']);
17
18 Route::post('/pegawai/update', [PegawaiController::class, 'update']);
19
20 Route::get('/pegawai/hapus{id}', [PegawaiController::class, 'hapus']);
21
```

➤ Methode hapus



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** On the left, it lists the project structure. The `PegawaiController.php` file is currently selected.
- Editor:** The main area displays the `PegawaiController.php` code. The code defines a controller class that handles HTTP requests for managing employees (pegawai). It includes methods for editing, updating, and deleting employee records.
- Bottom Bar:** Includes icons for file operations (New, Open, Save, Find, Replace, Cut, Copy, Paste), tabs for switching between files, and a status bar showing file path (`data-pegawai-arvin`), line/col count (Ln 56, Col 1), spaces/lf count (Spaces: 4, LF), file type (PHP), and a Go Live button.
- Bottom Icons:** Shows system icons for battery level (28°C), signal strength, and date/time (02/06/2025).

```
File Edit Selection View Go Run ... ← → data-pegawai-arvin
EXPLORER .env web.php PegawaiController.php ✘ console.php edit.blade.php tambah.blade.php index.blade.php
DATA... app Http Controllers PegawaiController.php
app > Http > Controllers > PegawaiController.php > ...
8 class PegawaiController extends Controller
33     public function edit($id)
34         $pegawai = DB::table('pegawai')->where('pegawai_id', $id)->get();
35         return view('edit', ['pegawai' => $pegawai]);
36     }
37
38     public function update(Request $request)
39     {
40         DB::table('pegawai')->where('pegawai_id', $request->id)->update([
41             'pegawai_nama' => $request->nama,
42             'pegawai_jabatan' => $request->jabatan,
43             'pegawai_umur' => $request->umur,
44             'pegawai_alamat' => $request->alamat,
45         ]);
46         return redirect('/pegawai');
47     }
48
49     public function hapus($id)
50     {
51         DB::table('pegawai')->where('pegawai_id', $id)->delete();
52         return redirect('/pegawai');
53     }
54
55 }
56
```

Data Pegawai

[+ Tambah Pegawai Baru](#)

Nama	Jabatan	Umur	Alamat	Opsi
Nur Arivin Umar	Direktur Utama	25	Pemalang	Edit Hapus
gita novitasari	SPV	21	KENDAL	Edit Hapus

Screenshot of a web browser window showing a CRUD application for employees.

The title bar reads "Tutorial Membuat CRUD Pada L... x".

The address bar shows "localhost:8000/pegawai".

The main content area displays a table titled "Data Pegawai" with one row:

Nama	Jabatan	Umur	Alamat	Opsi
Nur Arivin Umar	Direktur Utama	25	Pemalang	Edit Hapus

A link "+ Tambah Pegawai Baru" is visible above the table.

The taskbar at the bottom includes icons for weather (26°C Kabut), search, file explorer, email, messaging, browser, code editor, task manager, and system tray.