

**Name: Neo Zulkarnain Masthur**

**NIM: 1204220051**

**Class: IS-05-02**

## 1. Membuat dan Mendefinisikan Factory

```
PS D:\Program Files\Xampp\htdocs\uses\Modul-9> php artisan make:factory PositionFactory

INFO Factory [D:\Program Files\Xampp\htdocs\uses\Modul-9\database\factories\PositionFactory.php] created successfully.

PS D:\Program Files\Xampp\htdocs\uses\Modul-9> php artisan make:factory EmployeeFactory

INFO Factory [D:\Program Files\Xampp\htdocs\uses\Modul-9\database\factories\EmployeeFactory.php] created successfully.

PS D:\Program Files\Xampp\htdocs\uses\Modul-9>
```

*Gambar 1 Database Position and Factory factory files created*

```
PositionFactory.php X EmployeeFactory.php
database > factories > PositionFactory.php > PositionFactory > definition
1  <?php
2
3  namespace Database\Factories;
4
5  use Illuminate\Database\Eloquent\Factories\Factory;
6
7  /**
8   * @extends \Illuminate\Database\Eloquent\Factories\Factory<\App\Models\Position>
9   */
10 class PositionFactory extends Factory
11 {
12     /**
13      * Define the model's default state.
14      *
15      * @return array<string, mixed>
16      */
17     public function definition(): array
18     {
19         return [
20             'code' => fake()->stateAbbr(),
21             'name' => fake()->jobTitle(),
22             'description' => fake()->sentence(),
23         ];
24     }
25 }
```

*Gambar 2 Code given for output of random job positions*

```

PositionFactory.php X EmployeeFactory.php X
database > factories > EmployeeFactory.php > EmployeeFactory > definition
1  <?php
2
3  namespace Database\Factories;
4
5  use Illuminate\Database\Eloquent\Factories\Factory;
6
7  /**
8   * @extends \Illuminate\Database\Eloquent\Factories\Factory<\App\Models\Employee>
9   */
10 class EmployeeFactory extends Factory
11 {
12     /**
13      * Define the model's default state.
14      *
15      * @return array<string, mixed>
16      */
17     public function definition(): array
18     {
19         return [
20             'firstname' => fake()->firstName(),
21             'lastname' => fake()->lastName(),
22             'email' => fake()->email(),
23             'age' => fake()->numberBetween(25, 50),
24             'position_id' => Position::factory()
25         ];
26     }
27 }

```

*Gambar 3 Code give for output of random employee data with positions*

## 2. Menjalankan Factory via Seeder

```
PositionSeeder.php X EmployeeSeeder.php
database > seeders > PositionSeeder.php > PositionSeeder > run
1  <?php
2
3  namespace Database\Seeders;
4
5  use App\Models\Position;
6  use Illuminate\Database\Console\Seeds\WithoutModelEvents;
7  use Illuminate\Database\Seeder;
8  use Illuminate\Support\Facades\DB;
9
10 class PositionSeeder extends Seeder
11 {
12     /**
13      * Run the database seeds.
14      */
15     public function run(): void
16     {
17         Position::factory()->count(5)->create();
18     }
19 }
```

Gambar 4 PositionSeeder code changed to created 5 random positions

```
EmployeeSeeder.php X
database > seeders > EmployeeSeeder.php > EmployeeSeeder > run
1  <?php
2
3  namespace Database\Seeders;
4
5  use App\Models\Employee;
6  use Illuminate\Database\Console\Seeds\WithoutModelEvents;
7  use Illuminate\Database\Seeder;
8  use Illuminate\Support\Facades\DB;
9
10 class EmployeeSeeder extends Seeder
11 {
12     /**
13      * Run the database seeds.
14      */
15     public function run(): void
16     {
17         Employee::factory()->count(10)->create();
18     }
19 }
```

Gambar 5 EmployeePosition code changed to create 10 random employee data

```
PS D:\Progam Files\Xampp\htdocs\uses\Modul-9> php artisan db:seed

INFO Seeding database.

Database\Seeders\PositionSeeder ..... RUNNING
Database\Seeders\PositionSeeder ..... 59 ms DONE

Database\Seeders\EmployeeSeeder ..... RUNNING
Database\Seeders\EmployeeSeeder ..... 80 ms DONE

PS D:\Progam Files\Xampp\htdocs\uses\Modul-9> 
```



























Gambar 6 Seeder activated into Database

←T→	id	code	name	description	created_at	updated_at
<input type="checkbox"/> Edit Copy Delete	1	FE	Front End Developer	Front End Developer	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	2	BE	Back End Developer	Back End Developer	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	3	SA	System Analist	System Analist	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	4	CA	Police and Sheriffs Patrol Officer	Libero rem sed consequat.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	5	MO	User Experience Manager	Sed impedit officiis illo eveniet.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	6	GA	Gaming Cage Worker	Est neque beatae omnis sed.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	7	NJ	Home Appliance Installer	Ut quo non rerum similique quia quod in.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	8	CO	Well and Core Drill Operator	Ut reprehenderit nisi dolore ut iure.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	9	HI	Anthropology Teacher	Reiciendis molestiae repudiandae voluptatum error ...	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	10	MI	Sawing Machine Operator	Rerum et quae corrupti voluptas harum.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	11	OK	Credit Checkers Clerk	Sit quam totam esse.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	12	MS	Agricultural Crop Farm Manager	Vel et aut laborum adipisci ea ex eaque.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	13	SD	Employment Interviewer	Exercitationem quia eligendi sit quidem.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	14	DC	Textile Dyeing Machine Operator	Cumque delectus optio cum aut autem quo provident.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	15	MN	Veterinarian	Esse facere delectus perferendis harum.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	16	VT	Airfield Operations Specialist	In error itaque qui distinctio.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	17	KS	Electrical and Electronic Inspector and Tester	Aut quibusdam labore quidem placeat.	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	18	ME	Personnel Recruiter	Nobis consequat optio architecto illo.	2024-06-11 08:49:34	2024-06-11 08:49:34

*Gambar 7 Random positions created*

←T→	id	firstname	lastname	email	age	position_id	created_at	updated_at
<input type="checkbox"/> Edit Copy Delete	1	Purnama	Anaking	purnama.anaking@gmail.com	20	1	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	2	Adzanil	Rachmadhi	adzanil.rachmadhi@gmail.com	25	2	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	3	Berlian	Rahmy	berlian.rahmy@gmail.com	23	3	NULL	NULL
<input type="checkbox"/> Edit Copy Delete	5	Marta	Larkin	aweber@gmail.com	25	9	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	6	Gloria	Altenwerth	wschamberger@west.com	35	10	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	7	Opal	O'Connell	little.ellis@kihn.com	43	11	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	8	Gerry	Hegmann	stefan00@mueller.com	34	12	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	9	Bailey	Hermann	bernita36@mante.com	35	13	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	10	Sid	Cummings	orpha26@braun.biz	30	14	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	11	Onie	Keeling	proberts@cole.info	42	15	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	12	Paige	Walsh	neal53@carter.com	30	16	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	13	Leon	Kub	afadel@goodwin.com	41	17	2024-06-11 08:49:34	2024-06-11 08:49:34
<input type="checkbox"/> Edit Copy Delete	14	Ola	Hoppe	dave74@gmail.com	42	18	2024-06-11 08:49:34	2024-06-11 08:49:34

*Gambar 8 Random employee data created*

Data Master <a href="#">Home</a> <a href="#">Employee List</a> <span>My Profile</span>					
Employee List					Create Employee
First Name	Last Name	Email	Age	Position	
Purnama	Anaking	purnama.anaking@gmail.com	20	Front End Developer	 
Adzanil	Rachmadhi	adzanil.rachmadhi@gmail.com	25	Back End Developer	 
Berlian	Rahmy	berlian.rahmy@gmail.com	23	System Analyst	 
Marta	Larkin	aweber@gmail.com	25	Anthropology Teacher	 
Gloria	Altenwerth	wschamberger@west.com	35	Sawing Machine Operator	 
Opal	O'Connell	little.ellis@kihn.com	43	Credit Checkers Clerk	 
Gerry	Hegmann	stefan00@mueller.com	34	Agricultural Crop Farm Manager	 
Bailey	Hermann	bernita36@mante.com	35	Employment Interviewer	 
Sid	Cummings	orpha26@braun.biz	30	Textile Dyeing Machine Operator	 
Onie	Keeling	proberts@cole.info	42	Veterinarian	 
Paige	Walsh	neal53@carter.com	30	Airfield Operations Specialist	 
Leon	Kub	afadel@goodwin.com	41	Electrical and Electronic Inspector and Tester	 
Ola	Hoppe	dave74@gmail.com	42	Personnel Recruiter	 

Gambar 9 Employee Data page after Database updated

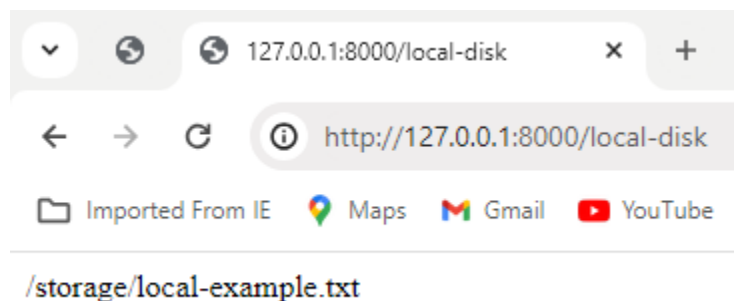
### 3&4. Meletakkan File pada Local dan Public Disk

```

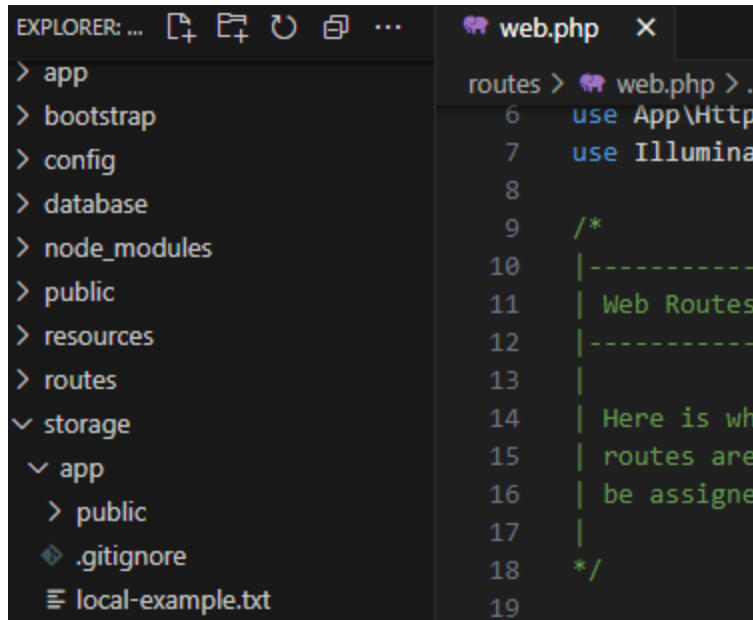
27 Route::get('/local-disk', function () {
28     Storage::disk('local')->put('local-example.txt', 'This is local example content');
29     return asset('storage/local-example.txt');
30 });
31
32 Route::get('/public-disk', function () {
33     Storage::disk('public')->put('public-example.txt', 'This is public example content');
34     return asset('storage/public-example.txt');
35 });

```

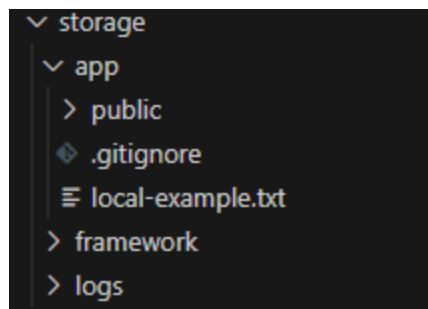
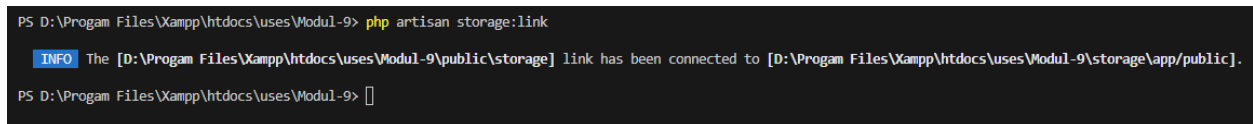
Gambar 10 Local and Public route added



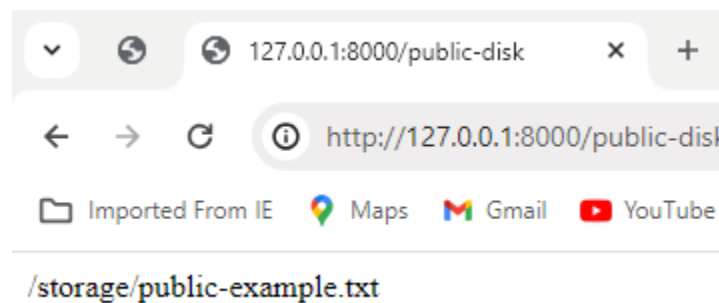
Gambar 11 Result of using "local-disk" route



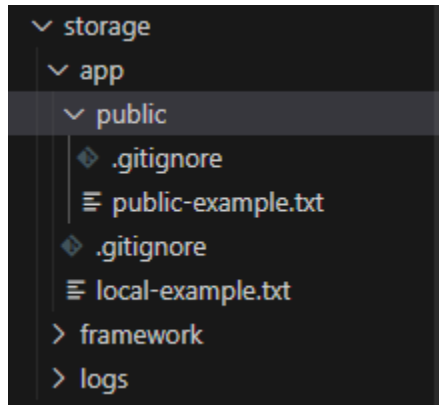
Gambar 12 File appearing after running "local-disk" route



Gambar 13 Gambar 12 Public folder created in resource folder

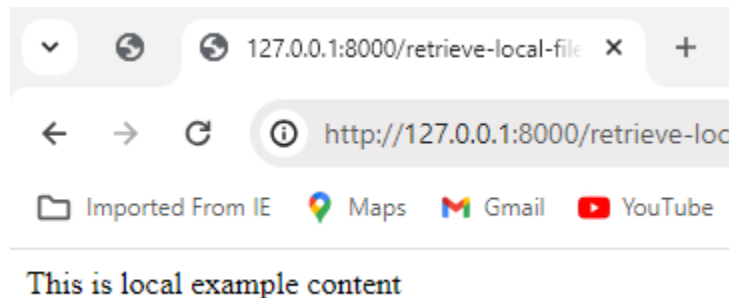


Gambar 14 Result of using "public-disk" route

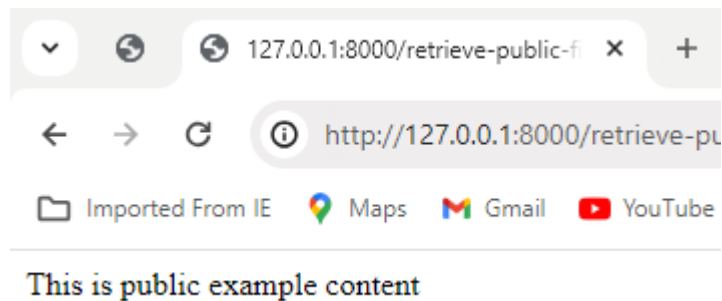


Gambar 15 File appearing after running "public-disk" route

## 5. Menampilkan Isi File



Gambar 16 using the "retrieve-local-file", it shows the text inside the "local-example.txt" file



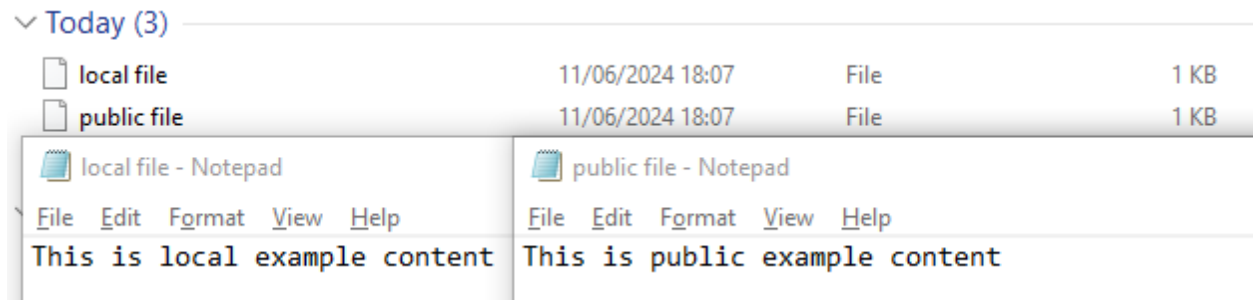
Gambar 17 using the "retrieve-public-file", it shows the text inside the "public-example.txt" file

## 6. Mendownload File

```
55 Route::get('/download-local-file', function () {  
56     return Storage::download('local-example.txt', 'local file');  
57 });  
58  
59 Route::get('/download-public-file', function () {  
60     return Storage::download('public/public-example.txt', 'public file');  
61 });
```

Gambar 18 Route for file download locally and publicly





*Gambar 19 Result of running said routes*

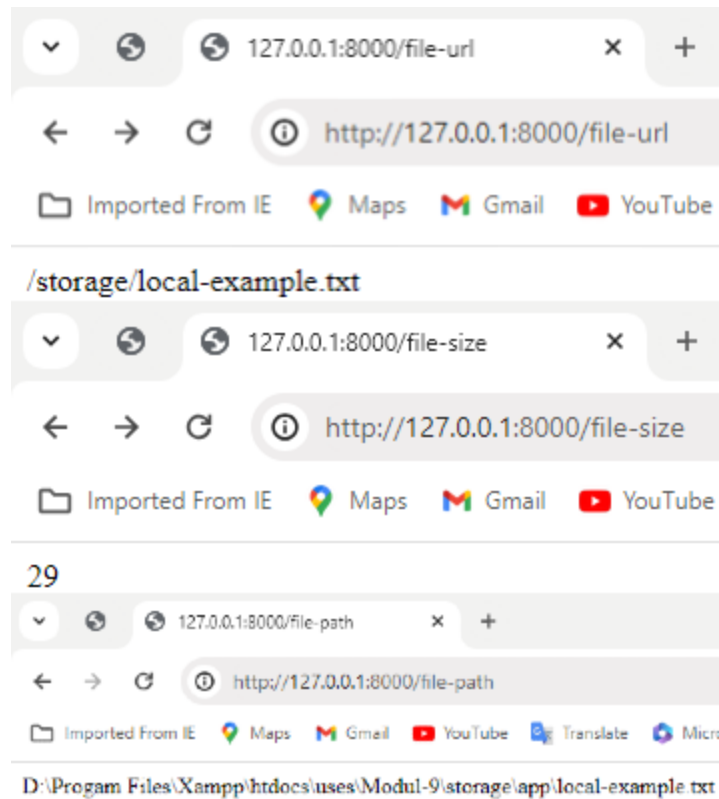
## 7. Menampilkan URL, Path dan Size dari File

```

63 Route::get('/file-url', function () {
64     // Just prepend "/storage" to the given path and return a relative URL
65     $url = Storage::url('local-example.txt');
66     return $url;
67 });
68 Route::get('/file-size', function () {
69     $size = Storage::size('local-example.txt');
70     return $size;
71 });
72 Route::get('/file-path', function () {
73     $path = Storage::path('local-example.txt');
74     return $path;
75 });

```

*Gambar 20 Routes for showing file URL, Size, and Location*



*Gambar 21 Result for running each Routes*

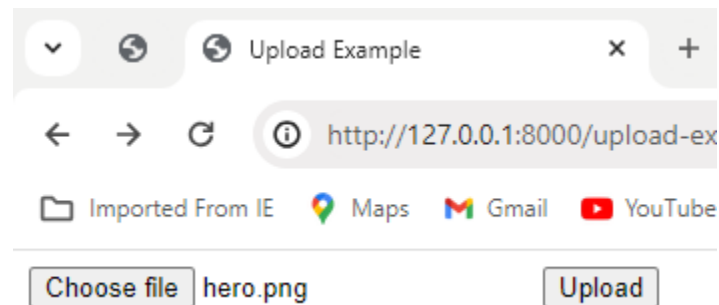
## 8. Menyimpan File via Form

```
78 Route::get('/upload-example', function () {  
79     return view('upload_example');  
80 });  
81 Route::post('/upload-example', function (Request $request) {  
82     $path = $request->file('avatar')->store('public');  
83     return $path;  
84 }->name('upload-example');
```

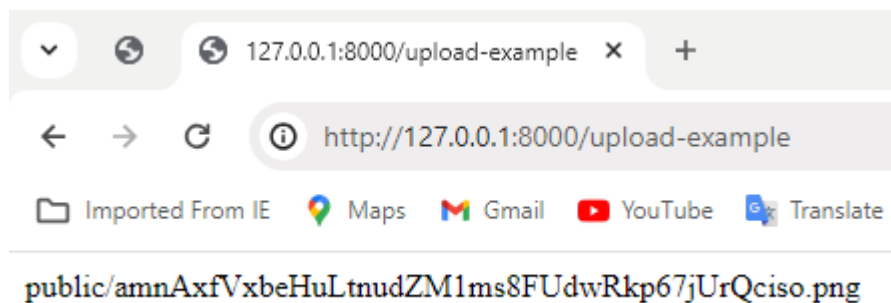
*Gambar 22 Route for displaying upload page and it's pathing*

```
web.php  upload_example.blade.php X
resources > views > upload_example.blade.php > html
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <meta http-equiv="X-UA-Compatible" content="ie=edge">
8      <title>Upload Example</title>
9  </head>
10
11 <body>
12     <form action="{{ route('upload-example') }}" method="POST" enctype="multipart/form-data">
13         @csrf
14         <input type="file" name="avatar">
15         <button type="submit">Upload</button>
16     </form>
17 </body>
18
19 </html>
```

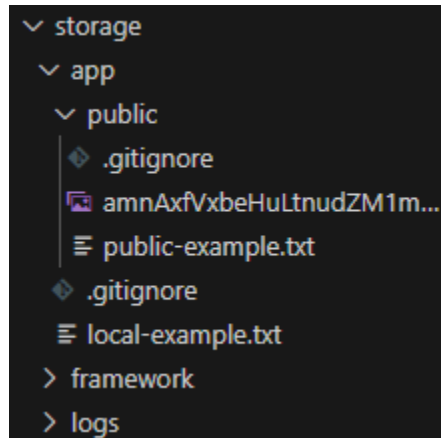
Gambar 23 Code for Upload Page



Gambar 24 Upload page display with a chosen file



Gambar 25 Result after picking a file and uploading it

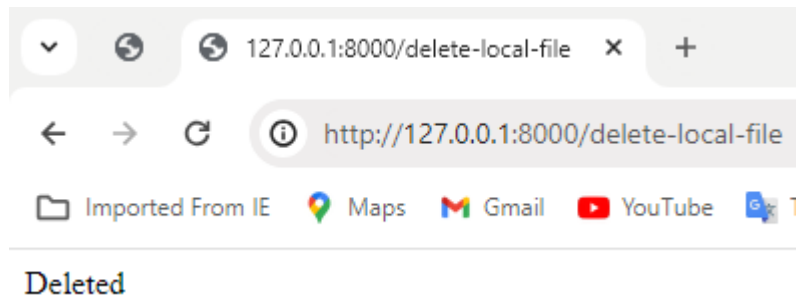


*Gambar 26 Uploaded file kept into public folder with data intact*

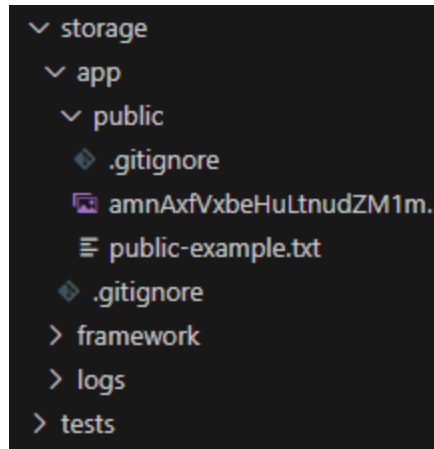
## 9. Menghapus File pada Storage

```
86 Route::get('/delete-local-file', function (Request $request) {
87     Storage::disk('local')->delete('local-example.txt');
88     return 'Deleted';
89 });
90 Route::get('/delete-public-file', function (Request $request) {
91     Storage::disk('public')->delete('public-example.txt');
92     return 'Deleted';
93 });
```

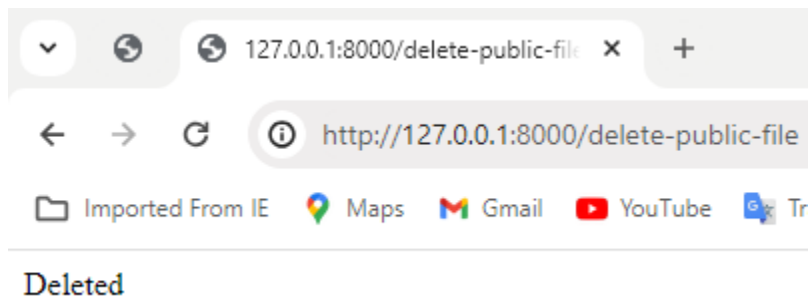
*Gambar 27 Routes added for file deletion*



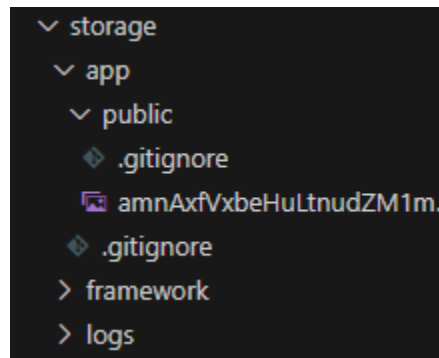
*Gambar 28 Local file deleted using 'delete-local-file' URL*



Gambar 29 Result of deletion

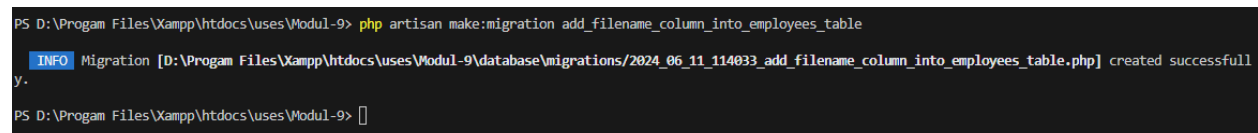


Gambar 30 Local file deleted using 'delete-public-file' URL



Gambar 31 Result of deletion

## 10. Menerapkan Fitur Upload pada Web Data Master Employee



Gambar 32 Migration file created

```
web.php 2024_06_11_114033_add_filename_column_into_employees_table.php X
database > migrations > 2024_06_11_114033_add_filename_column_into_employees_table.php > class
1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 return new class extends Migration
8 {
9     /**
10      * Run the migrations.
11      */
12     public function up(): void
13     {
14         Schema::table('employees', function (Blueprint $table) {
15             $table->string('original_filename')->after('position_id')->nullable();
16             $table->string('encrypted_filename')->after('original_filename')->nullable();
17         });
18     }
19
20     /**
21      * Reverse the migrations.
22      */
23     public function down(): void
24     {
25         Schema::table('employees', function (Blueprint $table) {
26             $table->dropColumn('encrypted_filename');
27             $table->dropColumn('original_filename');
28         });
29     }
30 };
31
```

Gambar 33 Code given for migrate file

```
PS D:\Progam Files\Xampp\htdocs\uses\Modul-9> php artisan migrate
INFO Running migrations.
2024_06_11_114033_add_filename_column_into_employees_table ..... 112ms DONE
PS D:\Progam Files\Xampp\htdocs\uses\Modul-9> 
```

Gambar 34 File migrated

```
web.php create.blade.php X
resources > views > employee > create.blade.php > div.container-sm.mt-5 > form
1 @extends('layouts.app')
2 @section('content')
3     <div class="container-sm mt-5">
4         <form action="{{ route('employees.store') }}" method="POST" enctype="multipart/form-data">
```

Gambar 35 enctype method added



## Create Employee

First Name

Last Name

Email

Age

Position



⬅ Cancel

✔ Save

*Gambar 36 Display before added code*



## Create Employee

First Name

Last Name

Email

Age

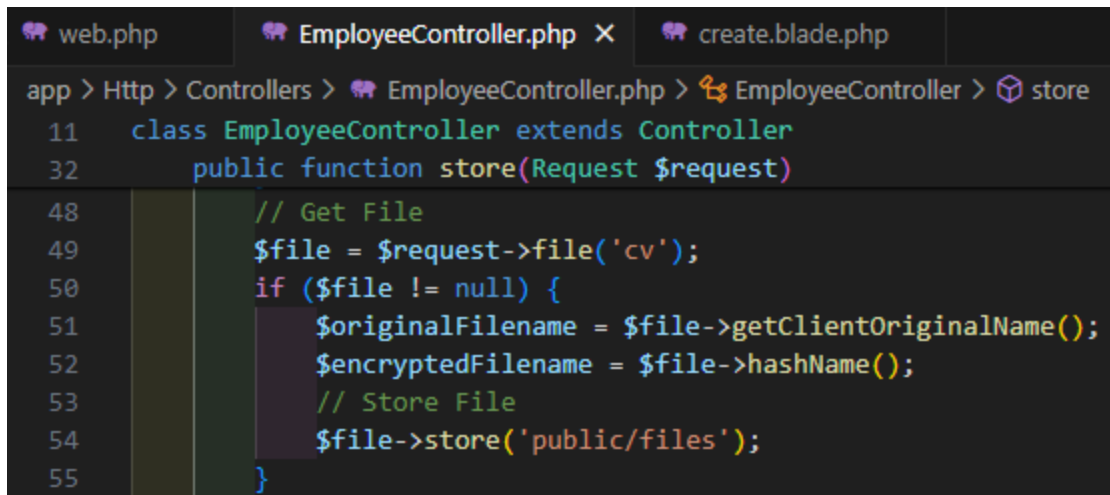
Position



Curriculum Vitae (CV)

*Gambar 37 Display after added code*





```
web.php EmployeeController.php X create.blade.php
app > Http > Controllers > EmployeeController.php > EmployeeController > store
11 class EmployeeController extends Controller
32     public function store(Request $request)
48         // Get File
49         $file = $request->file('cv');
50         if ($file != null) {
51             $originalFilename = $file->getClientOriginalName();
52             $encryptedFilename = $file->hashName();
53             // Store File
54             $file->store('public/files');
55         }
```

*Gambar 38 Added code to EmployeeController on store() method for file upload*



## Create Employee

First Name

Neo

Last Name

Zulkarnain

Email

neo.zulkarnain@gmail.com

Age

20

Position

ME - Personnel Recruiter



Curriculum Vitae (CV)

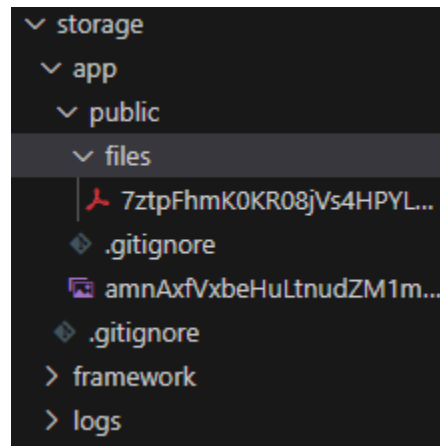
Choose file

neo\_zulkarnain\_cv.pdf

⬅ Cancel

✔ Save

Gambar 39 Contents of Employee Data before saving



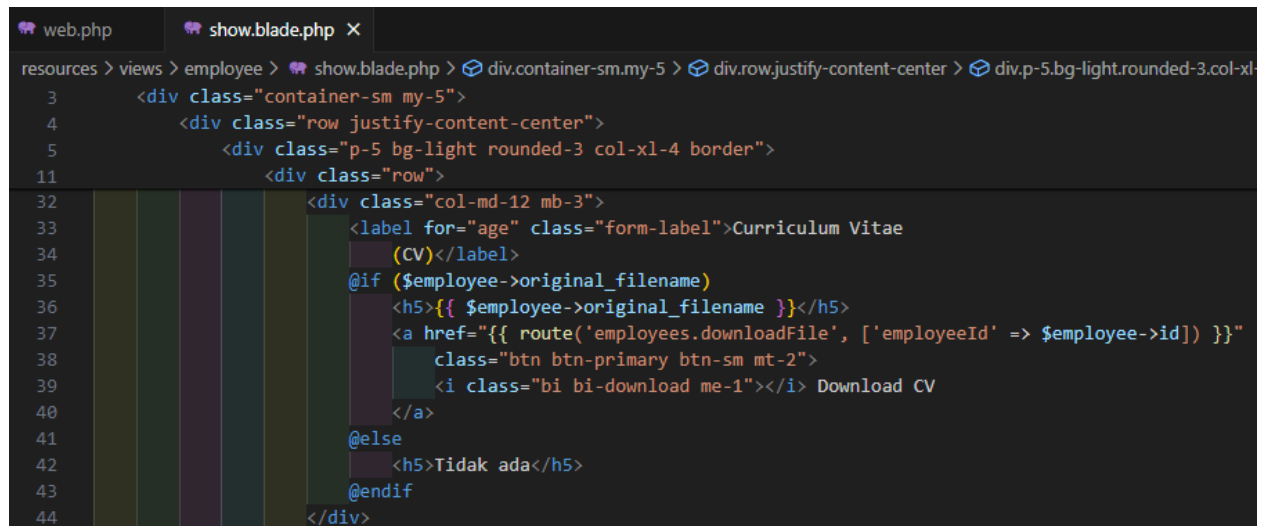
Gambar 40 Successfully added

```

95 Route::get('download-file/{employeeId}', [
96     EmployeeController::class, 'downloadFile'
97 ])->name('employees.downloadFile');

```

Gambar 41 Route for ability to download a file



Gambar 42 Download function added onto Employee Data Profile



## Detail Employee

First Name	Last Name	Age
Neo	Zulkarnain	20

Email  
**neo.zulkarnain@gmail.com**

Position  
**Personnel Recruiter**

Curriculum Vitae (CV)  
**neo\_zulkarnain\_cv.pdf**

[Download CV](#)

[← Back](#)

Gambar 43 Display after adding download function

Today (4)

neo\_zulkarnain\_cv

11/06/2024 19:30 Avast HTML Document

195 KB

Gambar 44 Successfully downloaded

## 11. Latihan



## Edit Employee

First Name

Neo

LastName

Zulkarnain

Email

neo.zulkarnain@gmail.com

Age

20

Position

ME - Personnel Recruiter



Curriculum Vitae (CV)

Choose file

No file chosen

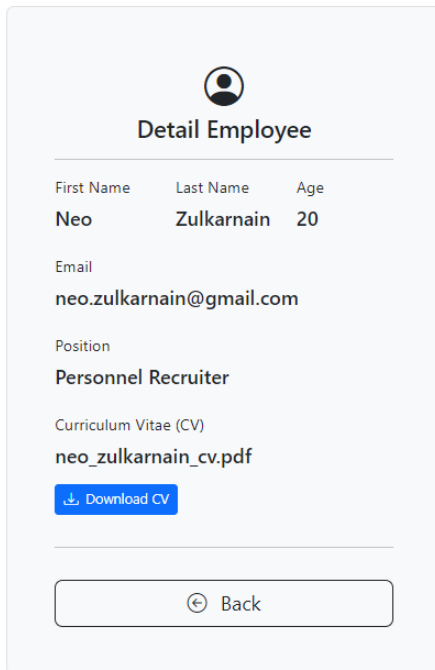
Current CV: neo\_zulkarnain\_cv.pdf

☐ Delete current CV

⬅ Cancel

✔ Edit

*Gambar 45 New options for changing the CV file and it's between deleting and changing the file*



The screenshot shows a web form titled "Detail Employee" with a user icon. It contains a table for employee details, followed by fields for Email, Position, and Curriculum Vitae (CV). The CV field displays a file named "neo\_zulkarnain\_cv.pdf" and a blue "Download CV" button. At the bottom is a "Back" button with a left arrow icon.

First Name	Last Name	Age
Neo	Zulkarnain	20

Email  
neo.zulkarnain@gmail.com

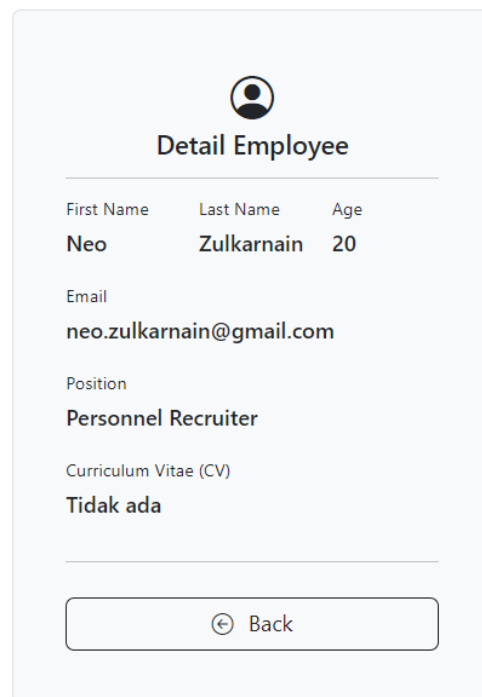
Position  
Personnel Recruiter

Curriculum Vitae (CV)  
neo\_zulkarnain\_cv.pdf

[Download CV](#)

[← Back](#)

Gambar 46 Before deletion of CV file



The screenshot shows the same "Detail Employee" form, but the CV field now displays "Tidak ada" (None), indicating the file has been successfully deleted. The "Download CV" button is no longer present.

First Name	Last Name	Age
Neo	Zulkarnain	20

Email  
neo.zulkarnain@gmail.com

Position  
Personnel Recruiter

Curriculum Vitae (CV)  
Tidak ada

[← Back](#)

Gambar 47 CV deleted by checking the check box and then save

## 12. Tugas

<https://github.com/ShrikeHolder/Modul-9>