

## **Jobsheet 12: File Upload**

Dosen Pengampu Mata Kuliah :

M. Hasyim Ratsanjani, S.Kom., M.Kom.



Dibuat Oleh :

Daffa Yudisa Akbar : TI-2H / 10 : 2241720008

Politeknik Negeri Malang

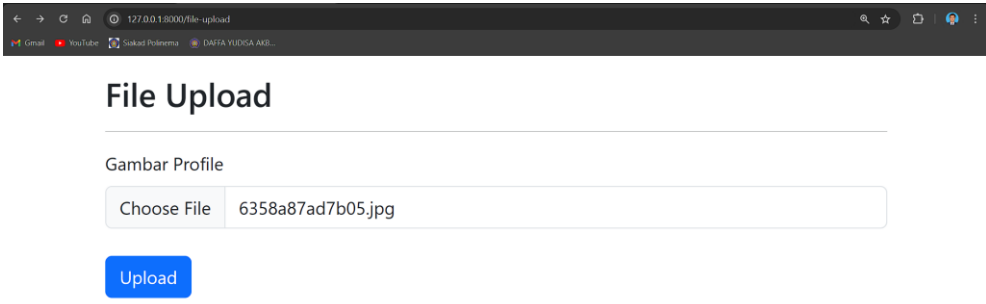
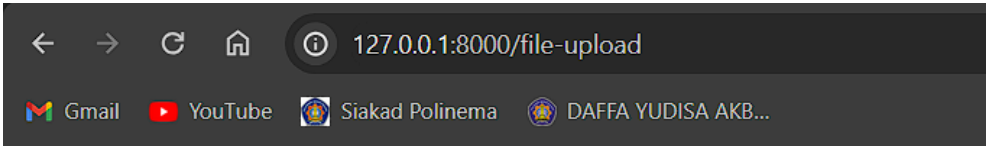
Jurusan Teknologi Informasi

D-IV Teknik Informatika

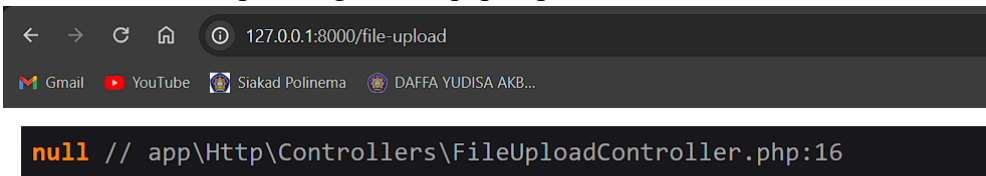
2024

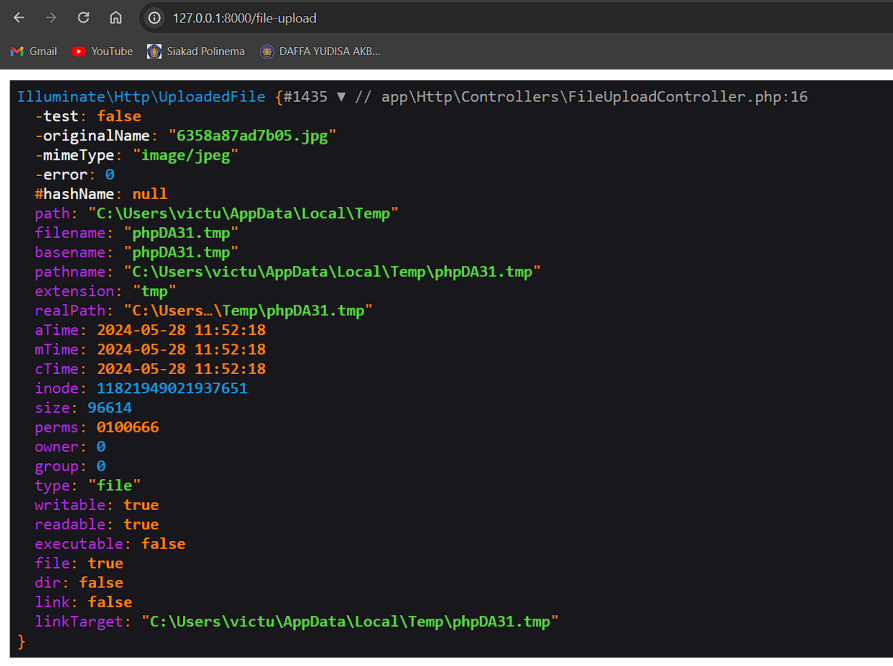
## A. Persiapan Awal

No.	Langkah Praktikum
1.	<p>Membuat beberapa file yang diperlukan, yang pertama controller : php artisan make:controller FileUploadController</p> <pre> PS C:\laragon\www\PWL_POS&gt; php artisan make:controller FileUploadController  INFO Controller [C:\laragon\www\PWL_POS\app\Http\Controllers\FileUploadController.php] created successfully.  PS C:\laragon\www\PWL_POS&gt; </pre>
2.	<p>Buat 2 buah method yaitu method fileUpload() dan prosesFileUpload. Method fileUpload() berisi kode program yang akan memanggil file view file-upload.blade.php. Sedangkan method prosesFileUpload() untuk memproses hasil submit form.</p> <pre> FileUploadController.php U X app &gt; Http &gt; Controllers &gt; FileUploadController.php &gt; PHP Intelephense &gt; FileUploadController  1  &lt;?php 2 3  namespace App\Http\Controllers; 4 5  use Illuminate\Http\Request; 6 7  class FileUploadController extends Controller 8  { 9      public function fileUpload() 10     { 11         return view('file-upload'); 12     } 13 14     public function prosesFileUpload(Request \$request) 15     { 16         return "Pemrosesan File Upload Disini"; 17     } 18 } </pre>
3.	<p>Membuat route pada web.php</p> <pre> 128 129 Route::get('/file-upload', [FileUploadController::class, 'fileUpload']); 130 Route::post('/file-upload', [FileUploadController::class, 'prosesFileUpload']); </pre>
4.	<p>Membuat File View file-upload.blade.php dengan kode sebagai berikut :</p> <pre> resources &gt; views &gt; file-upload.blade.php &gt; html &gt; head &gt; link 1  &lt;!DOCTYPE html&gt; 2  &lt;html lang="en"&gt; 3  &lt;head&gt; 4      &lt;meta charset="UTF-8"&gt; 5      &lt;meta name="viewport" content="width=device-width, initial-scale=1.0"&gt; 6      &lt;meta http-equiv="X-UA-Compatible" content="ie=edge"&gt; 7      &lt;link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet" 8      integrity="sha384-QWTKZyjpPEjISv5WaRU90FeRpok6YctnYmDr5pNlyT2bRjXh0JMhY6hW+ALEwIH" crossorigin="anonymous"&gt; 9      &lt;title&gt;File Upload&lt;/title&gt; 10 &lt;/head&gt; 11 &lt;body&gt; 12     &lt;div class="container mt-3"&gt; 13         &lt;h2&gt;File Upload&lt;/h2&gt; 14         &lt;hr&gt; 15 16         &lt;form action="{{ url('/file-upload') }}" method="POST" enctype="multipart/form-data"&gt; 17             @csrf 18             &lt;div class="mb-3"&gt; 19                 &lt;label for="berkas" class="form-label"&gt;Gambar Profile&lt;/label&gt; 20                 &lt;input type="file" class="form-control" id="berkas" name="berkas"&gt; 21                 @error('berkas') 22                     &lt;div class="text-danger"&gt;{{ \$message }}&lt;/div&gt; 23                 @enderror 24             &lt;/div&gt; 25 26             &lt;button type="submit" class="btn btn-primary my-2"&gt;Upload&lt;/button&gt; 27         &lt;/form&gt; 28     &lt;/div&gt; 29 &lt;/body&gt; 30 &lt;/html&gt; </pre>

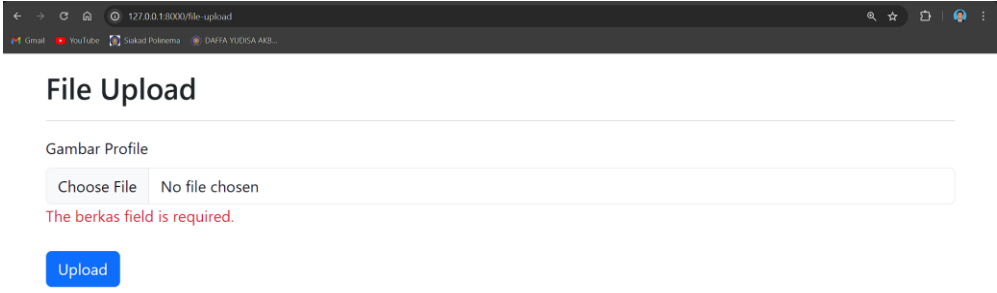
	<ul style="list-style-type: none"> <li>Hasil dari file-upload.blade.php</li> </ul>  <ul style="list-style-type: none"> <li>Ketika tombol upload ditekan</li> </ul>  <p>Pemrosesan File Upload Disini</p>
--	--

## B. Informasi File Upload

No.	Langkah Praktikum
1.	<p>Informasi seputar file yang sudah di upload bisa kita akses melalui Request object, yakni variabel \$request. Silahkan isi kode berikut ke dalam method prosesFileUpload() di FileUploadController:</p> <pre>public function prosesFileUpload(Request \$request) {     dump(\$request-&gt;berkas);     //return "Pemrosesan File Upload Disini"; }</pre>
2.	<p>lakukan 2 percobaan, pertama langsung submit form tanpa mengisi file apapun. Dan kedua, upload file sembarang dan submit. Screen Shot bagaimana hasilnya?</p> <ul style="list-style-type: none"> <li>Submit tanpa mengisi file apapun pada form</li> </ul>  <pre>null // app\Http\Controllers\FileUploadController.php:16</pre> <ul style="list-style-type: none"> <li>Submit menggunakan file</li> </ul>

	 <pre> Illuminate\Http\UploadedFile {#1435 ▾ // app\Http\Controllers\FileUploadController.php:16   -test: false   -originalName: "6358a87ad7b05.jpg"   -mimeType: "image/jpeg"   -error: 0   #hashName: null   path: "C:\Users\victu\AppData\Local\Temp"   filename: "phpDA31.tmp"   basename: "phpDA31.tmp"   pathname: "C:\Users\victu\AppData\Local\Temp\phpDA31.tmp"   extension: "tmp"   realPath: "C:\Users...Temp\phpDA31.tmp"   aTime: 2024-05-28 11:52:18   mTime: 2024-05-28 11:52:18   cTime: 2024-05-28 11:52:18   inode: 11821949021937651   size: 96614   perms: 0100666   owner: 0   group: 0   type: "file"   writable: true   readable: true   executable: false   file: true   dir: false   link: false   linkTarget: "C:\Users\victu\AppData\Local\Temp\phpDA31.tmp" } </pre>
3.	<p>Berikut cara mengambil beberapa info (app/Http/Controllers/FileUploadController.php) dari file yang di upload:</p>  <pre> app &gt; Http &gt; Controllers &gt; FileUploadController.php &gt; PHP Intelephense &gt; FileUploadController 1  &lt;?php 2 3  namespace App\Http\Controllers; 4 5  use Illuminate\Http\Request; 6 7  class FileUploadController extends Controller 8  { 9      public function fileUpload() 10     { 11         return view('file-upload'); 12     } 13 14     public function prosesFileUpload(Request \$request) 15     { 16         // dump(\$request-&gt;berkas); 17         // //return "Pemrosesan File Upload Disini"; 18         if(\$request-&gt;hasFile('berkas')){ 19             echo "path(): ".\$request-&gt;berkas-&gt;path(); 20             echo "&lt;br&gt;"; 21             echo "extension(): ".\$request-&gt;berkas-&gt;extension(); 22             echo "&lt;br&gt;"; 23             echo "getClientOriginalExtension(): ".\$request-&gt;berkas-&gt;getClientOriginalExtension(); 24             echo "&lt;br&gt;"; 25             echo "getMimeType(): ".\$request-&gt;berkas-&gt;getMimeType(); 26             echo "&lt;br&gt;"; 27             echo "getClientOriginalName(): ".\$request-&gt;berkas-&gt;getClientOriginalName(); 28             echo "&lt;br&gt;"; 29             echo "getSize(): ".\$request-&gt;berkas-&gt;getSize(); 30         } else { 31             echo "Tidak Ada Berkas yang Diupload"; 32         } 33     } 34 } 35 </pre>
4.	<p>Lakukan percobaan upload file sehingga akan muncul keterangan sbb:</p>  <pre> path(): C:\Users\victu\AppData\Local\Temp\phpC678.tmp extension(): jpg getClientOriginalExtension(): jpg getMimeType(): image/jpeg getClientOriginalName(): 6358a87ad7b05.jpg getSize(): 96614 </pre>

### C. Validasi File Upload

No.	Langkah Praktikum
1.	<p>Membuat validasi file upload mirip seperti cara membuat validasi inputan form biasa, yakni menggunakan method <code>\$request-&gt;validate()</code>. Berikut contoh penggunaannya: ( <code>app/Http/Controllers/FileUploadController.php</code> )</p> <pre>public function prosesFileUpload(Request \$request) {     \$request-&gt;validate([         'berkas' =&gt; 'required',]);     echo \$request-&gt;berkas-&gt;getClientOriginalName()."Lolos Validasi"; }</pre> 
2.	<p>Syarat validasi selanjutnya adalah membatasi jenis file dan maksimal ukuran file</p> <pre>public function prosesFileUpload(Request \$request) {     \$request-&gt;validate([         'berkas' =&gt; 'required file image max:500',]);     echo \$request-&gt;berkas-&gt;getClientOriginalName()."Lolos Validasi"; }</pre> <pre>;;;;;;;;;;;;;; ; File Uploads ; ;;;;;;;;;;;;;;  ; Whether to allow HTTP file uploads. ; https://php.net/file-uploads file_uploads = On  ; Temporary directory for HTTP uploaded files (will use system default if not ; specified). ; https://php.net/upload-tmp-dir upload_tmp_dir =  ; Maximum allowed size for uploaded files. ; https://php.net/upload-max-filesize upload_max_filesize = 2G  ; Maximum number of files that can be uploaded via a single request max_file_uploads = 20</pre>

### D. Memindah File Upload

N o.	Langkah Praktikum
1.	<p>Cara pertama adalah menggunakan method <code>store()</code> yang bisa diakses dari Request object. Berikut contoh penggunaannya:</p>

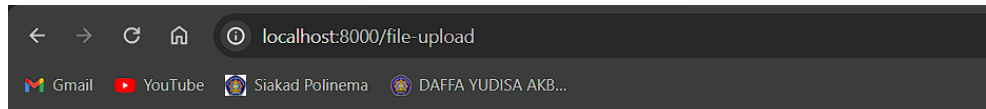
```

public function prosesFileUpload(Request $request)
{
    @param \Illuminate\Http\Request $request

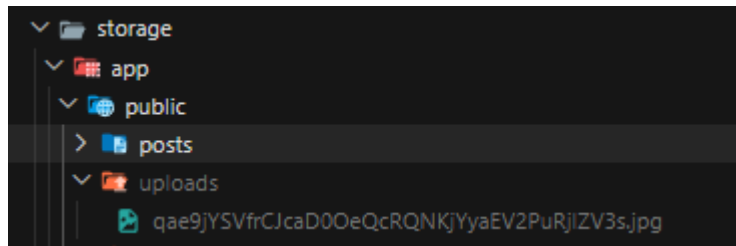
    $request->validate([
        'berkas' => 'required|file|image|max:500',]);
    $path = $request->berkas->store('uploads');
    echo "Proses Upload Berhasil, File Berada di: ".$path;
    //echo $request->berkas->getClientOriginalName()."Lolos Validasi";
}

```

- Setelah mengupload gambar



Proses Upload Berhasil, File Berada di: uploads/qa9jY5VfrCJcaD0OeQcRQNKjYyaEV2PuRjlZV3s.jpg



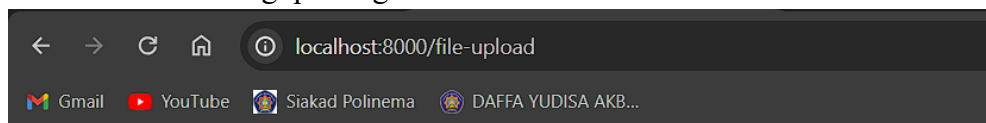
Bagaimana jika kita tetap ingin membuat nama file sendiri? Tidak masalah, Laravel menyediakan method `storeAs()` untuk keperluan ini. Berikut contoh penggunaannya:

```

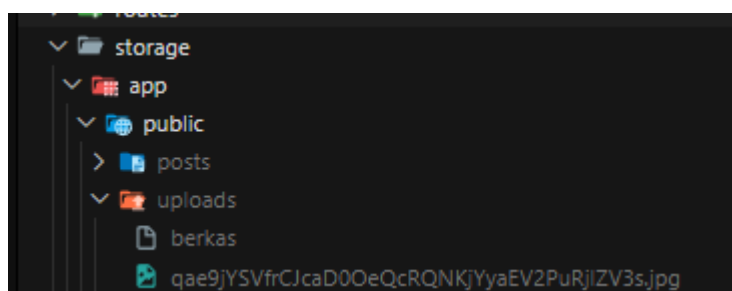
public function prosesFileUpload(Request $request)
{
    $request->validate([
        'berkas' => 'required|file|image|max:5000',]);
    // $path = $request->berkas->store('uploads');
    $path = $request->berkas->storeAs('uploads', 'berkas');
    echo "Proses Upload Berhasil, File Berada di: ".$path;
    //echo $request->berkas->getClientOriginalName()."Lolos Validasi";
}

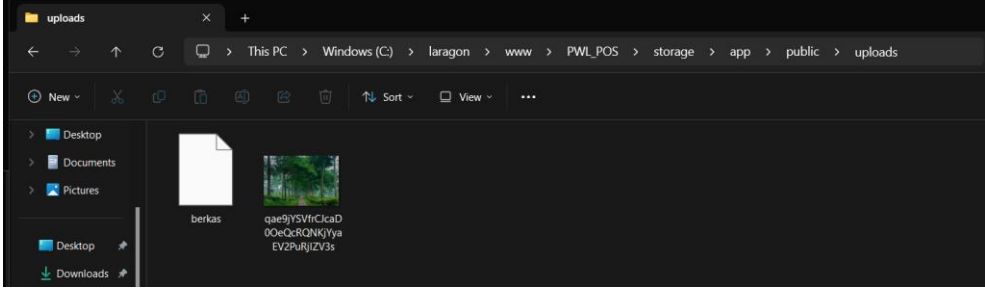
```

- Setelah mengupload gambar

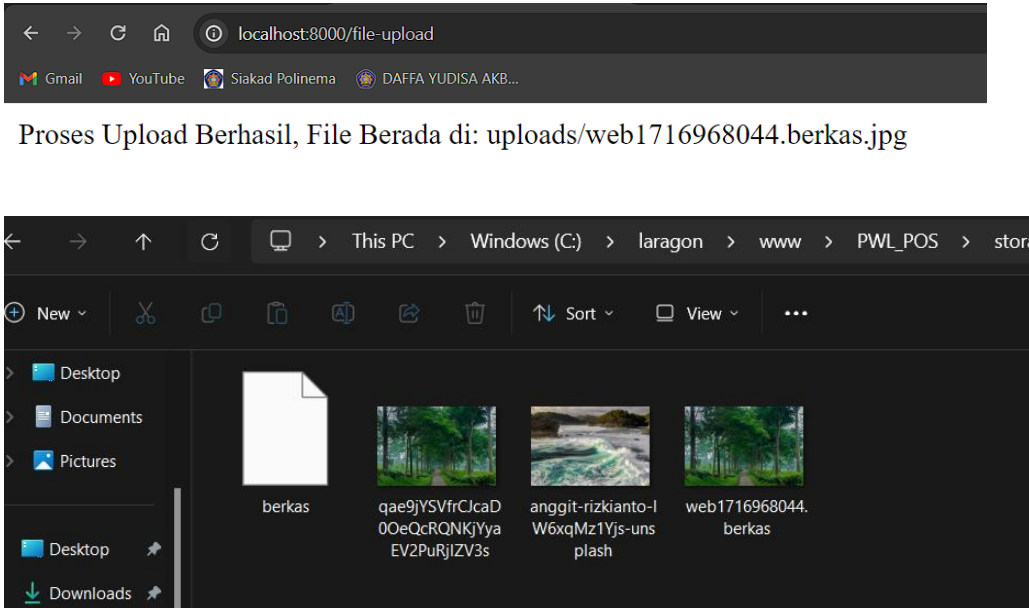


Proses Upload Berhasil, File Berada di: uploads/berkas

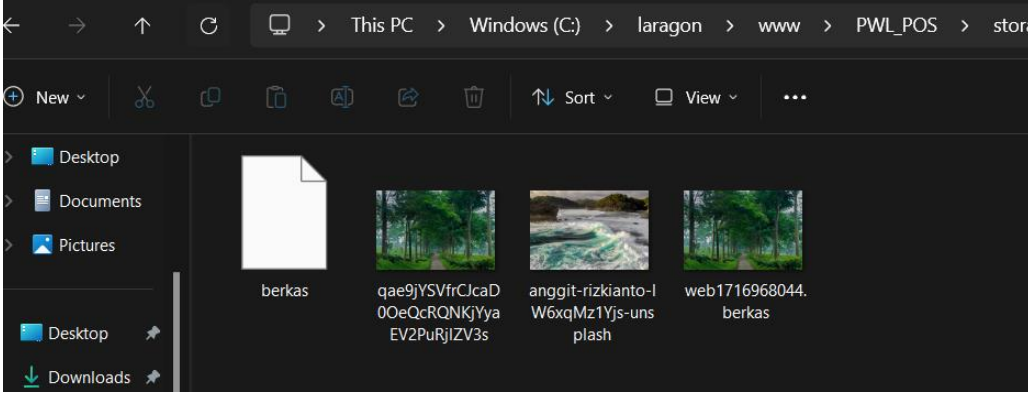


	
2.	<p>Mengambil nama file dari fungsi <code>getClientOriginalName()</code> seperti contoh berikut: (app/Http/Controllers/FileUploadController.php)</p> <pre> public function prosesFileUpload(Request \$request) {     \$request-&gt;validate([         'berkas' =&gt; 'required file image max:5000',]);     //\$path = \$request-&gt;berkas-&gt;store('uploads');     \$namaFile = \$request-&gt;berkas-&gt;getClientOriginalName();     \$path = \$request-&gt;berkas-&gt;storeAs('uploads', \$namaFile);     echo "Proses Upload Berhasil, File Berada di: ".\$path;     //echo \$request-&gt;berkas-&gt;getClientOriginalName()."Lolos Validasi"; } </pre> <ul style="list-style-type: none"> <li>• Setelah mengupload gambar</li> </ul>  
3.	<p>Cara lain adalah dengan meng-generate nama acak sendiri. Sebagai contoh, membuat nama file upload berdasarkan nama user, ditambah dengan digit acak dari fungsi <code>time()</code>. Berikut kode programnya: (app/Http/Controllers/FileUploadController.php)</p> <pre> public function prosesFileUpload(Request \$request) {     \$request-&gt;validate([         'berkas' =&gt; 'required file image max:5000',]);     //\$path = \$request-&gt;berkas-&gt;store('uploads');     \$extfile = \$request-&gt;berkas-&gt;getClientOriginalName();     \$namaFile = 'web'.time().".\$extfile";     \$path = \$request-&gt;berkas-&gt;storeAs('uploads', \$namaFile);     echo "Proses Upload Berhasil, File Berada di: ".\$path;     //echo \$request-&gt;berkas-&gt;getClientOriginalName()."Lolos Validasi"; } </pre>

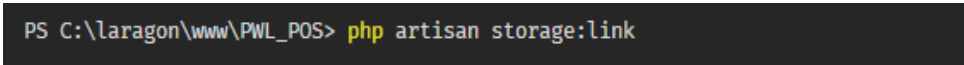
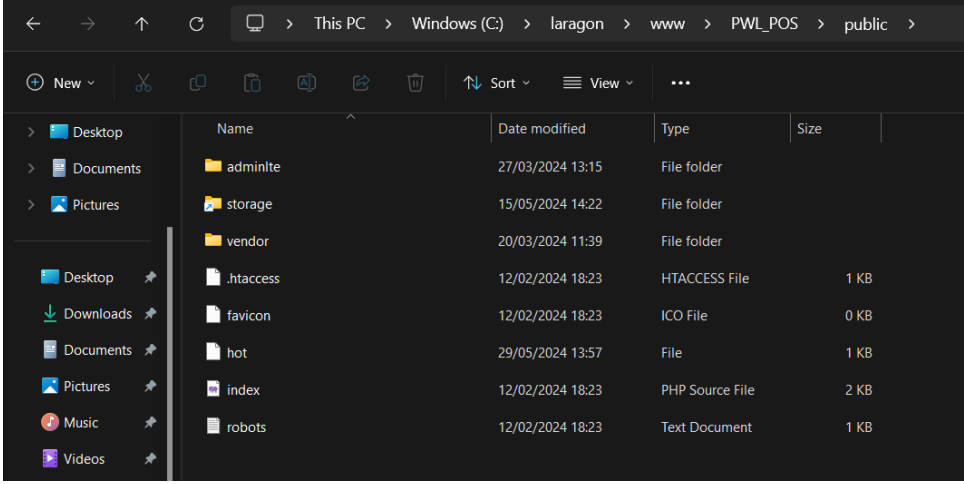
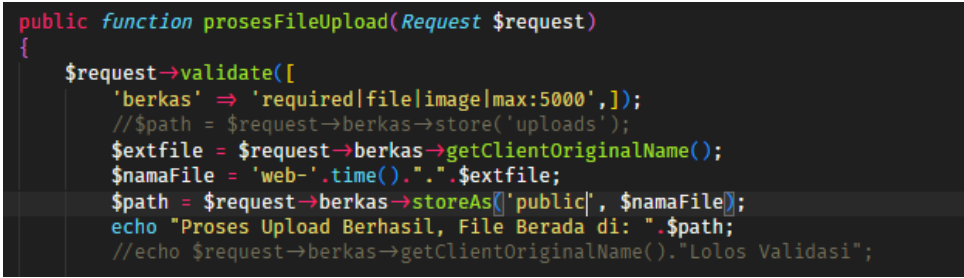
- Setelah mengupload gambar



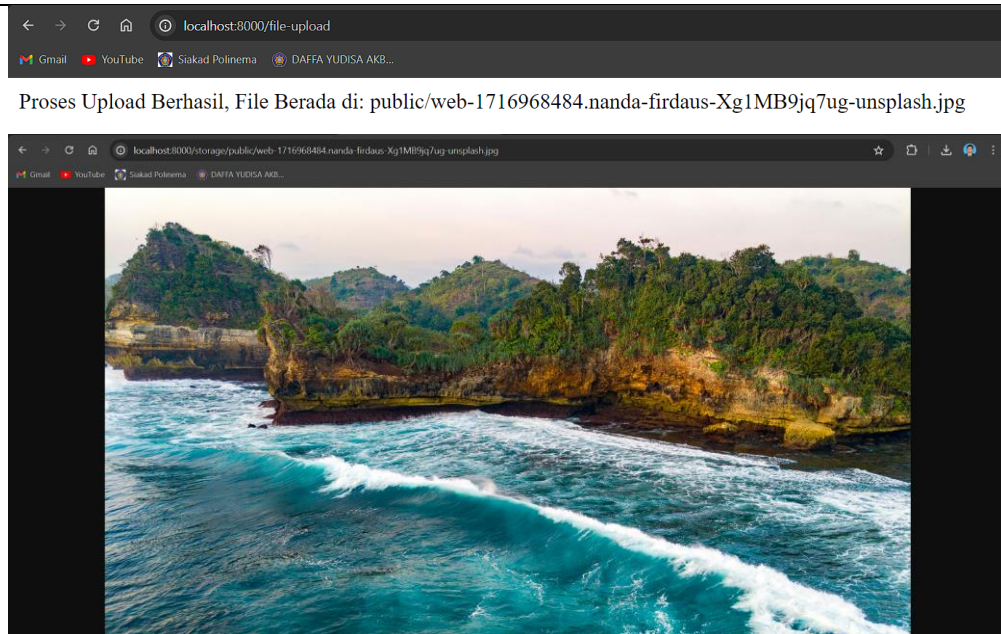
Proses Upload Berhasil, File Berada di: uploads/web1716968044.berkas.jpg



## E. Membuat Symlink

No.	Langkah Praktikum
1.	<p>Membuat symlink yang menghubungkan folder storage\app dengan folder public\ php artisan storage:link</p>  
2.	<p>Modifikasi dari method prosesFileUpload():</p>  <pre> public function prosesFileUpload(Request \$request) {     \$request-&gt;validate([         'berkas' =&gt; 'required file image max:5000',]);     // \$path = \$request-&gt;berkas-&gt;store('uploads');     \$extfile = \$request-&gt;berkas-&gt;getClientOriginalName();     \$namaFile = 'web-' . time() . '.' . \$extfile;     \$path = \$request-&gt;berkas-&gt;storeAs('public', \$namaFile);     echo "Proses Upload Berhasil, File Berada di: ".\$path;     // echo \$request-&gt;berkas-&gt;getClientOriginalName(). "LoLos Validasi"; } </pre> <ul style="list-style-type: none"> <li>• Setelah mengupload gambar</li> </ul>



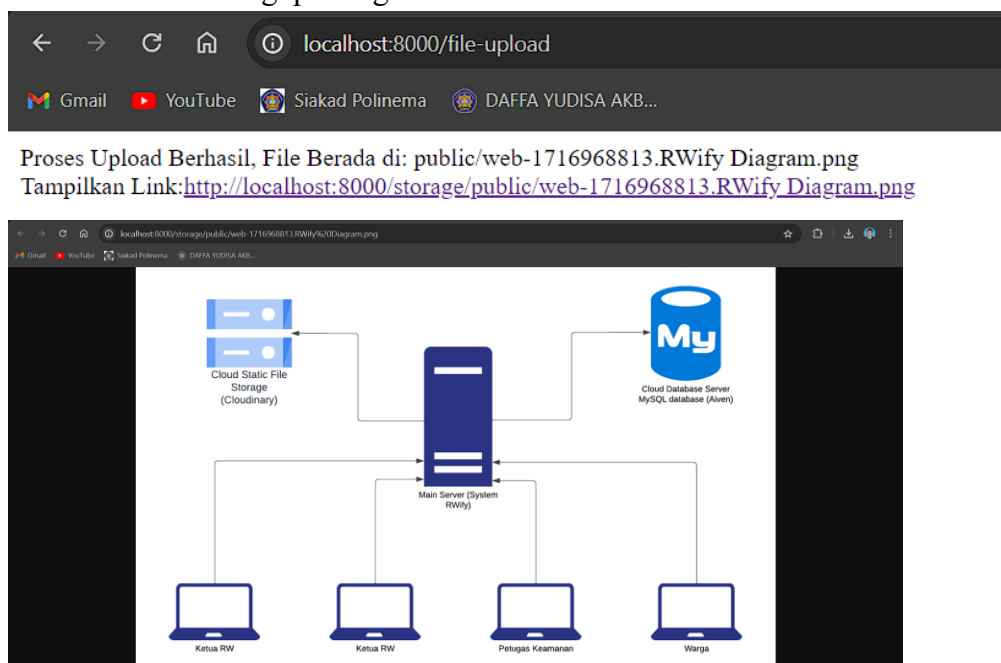


3. Menambahkan link agar gambar dapat diakses, tambahkan kode pada method prosesFileUpload:

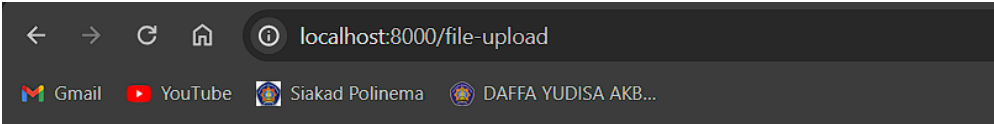

```
public function prosesFileUpload(Request $request)
{
    $request->validate([
        'berkas' => 'required|file|image|max:5000',]);
    // $path = $request->berkas->store('uploads');
    $extfile = $request->berkas->getClientOriginalName();
    $namaFile = 'web-' . time() . "." . $extfile;
    $path = $request->berkas->storeAs('public', $namaFile);

    $pathBaru = asset('storage/' . $namaFile);
    echo "Proses Upload Berhasil, File Berada di: " . $path;
    echo "<br>";
    echo "Tampilkan Link:<a href='$pathBaru'>$pathBaru</a>";
    // echo $request->berkas->getClientOriginalName(). "Lolos Validasi";
}
```

- Setelah mengupload gambar

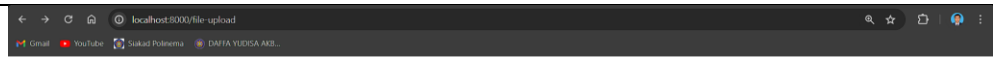


## F. Method Move

No.	Langkah Praktikum
1.	<p>Modifikasi controller FileUploadController.php</p> <pre> 14 public function prosesFileUpload(Request \$request) 15 { 16     \$request-&gt;validate([ 17         'berkas' =&gt; 'required file image max:5000',]); 18     //\$path = \$request-&gt;berkas-&gt;store('uploads'); 19     \$extfile = \$request-&gt;berkas-&gt;getClientOriginalName(); 20     \$namaFile = 'web-' . time() . "." . \$extfile; 21 22     \$path = \$request-&gt;berkas-&gt;move('gambar', \$namaFile); 23     \$path = str_replace("\\", "/", \$path); 24     echo "Variabel Path Berisi: \$path &lt;br&gt;"; 25 26     \$pathBaru = asset('gambar/' . \$namaFile); 27     echo "Proses Upload Berhasil, Data Disimpan Pada: ".\$path; 28     echo "&lt;br&gt;"; 29     echo "Tampilkan Link:&lt;a href='".\$pathBaru.'"&gt;\$pathBaru&lt;/a&gt;"; 30     //echo \$request-&gt;berkas-&gt;getClientOriginalName()."Lolos Validasi"; </pre> <ul style="list-style-type: none"> <li>Setelah mengupload gambar</li> </ul>  <p>Variabel Path Berisi: gambar/web-1716969173.6358a87ad7b05.jpg          Proses Upload Berhasil, Data Disimpan Pada: gambar/web-1716969173.6358a87ad7b05.jpg          Tampilkan Link: <a href="http://localhost:8000/gambar/web-1716969173.6358a87ad7b05.jpg">http://localhost:8000/gambar/web-1716969173.6358a87ad7b05.jpg</a></p> 

## Tugas

No.	Langkah Pengerjaan
1.	<p>Buat form file upload dimana kita bisa menentukan sendiri nama file akhir. Nama file ini diambil dari inputan text box yang juga ada di dalam form tersebut. Setelah di submit, langsung tampilkan file tersebut di web browser menggunakan tag <b>Error! Filename not specified.</b></p>



## File Upload

Nama File

Masukkan Nama Baru Untuk File

Gambar Profile

Choose File

No file chosen

Upload



## File Upload

Nama File

Pantai di Malang

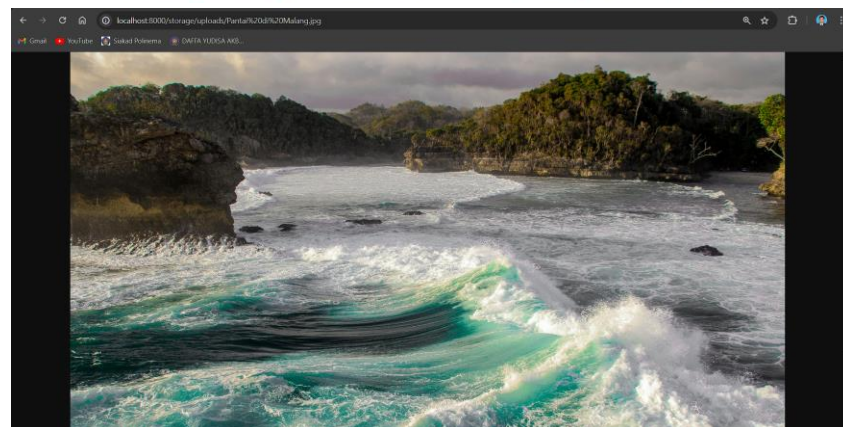
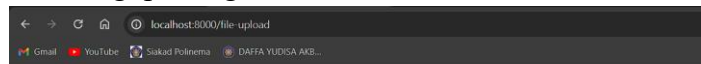
Gambar Profile

Choose File

anggit-rizkianto-lW6xqMz1Yjs-unsplash.jpg

Upload

- Setelah mengupload gambar



## 2. Tambahkan proses upload image dan application pada starter code

### • Controller

```
app > Http > Controllers > FileUploadController.php > PHP Intellisense > FileUploadController > prosesFileUpload
7 class FileUploadController extends Controller
8 {
9     public function fileUpload()
10     {
11         return view('file-upload');
12     }
13
14     public function prosesFileUpload(Request $request)
15     {
16         $request->validate([
17             'namaBaruFile' => 'required',
18             'berkas' => 'required|file|image|max:5000',
19             // $path = $request->berkas->store('uploads');
20             $extfile = $request->berkas->getClientOriginalExtension();
21             $namaFile = $request->namaBaruFile.".".$extfile;
22
23             $path = $request->berkas->storeAs('uploads', $namaFile);
24
25             echo "Gambar Berhasil Diupload <a href='storage/$path'>$namaFile</a>";
26             echo "<br><br>";
27             echo "<img src='storage/$path' width='750px'>";
28             // echo $request->berkas->getClientOriginalName()." Lolos Validasi";
29             // dump($request->berkas);
30             // return "Pemrosesan File Upload Disini";
31             // if($request->hasFile('berkas')){
32             //     echo "path(): ".$request->berkas->path();
33             //     echo "<br>";
34             //     echo "extension(): ".$request->berkas->extension();
35             //     echo "<br>";
36             //     echo "getClientOriginalExtension(): ".$request->berkas->getClientOriginalExtension();
37             //     echo "<br>";
38             //     echo "getMimeType(): ".$request->berkas->getMimeType();
39             //     echo "<br>";
40             //     echo "getClientOriginalName(): ".$request->berkas->getClientOriginalName();
41             //     echo "<br>";
42             //     echo "getSize(): ".$request->berkas->getSize();
43             // } else {
44             //     echo "Tidak Ada Berkas yang Diupload";
45             // }
46
47 }
48
```

### • Views

```
resources > Views > file-upload.blade.php > html > body > div.container mt-3 > form > div mb-3 > input namaBaruFile form-control > input berkas form-control > button submit class="btn btn-primary my-2" > Upload
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport" content="width=device-width, initial-scale=1.0">
6     <meta http-equiv="X-UA-Compatible" content="ie=edge">
7     <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet"
8     integrity="sha384-QWIKZqyPPPEjISv5WwR90Fekp06YctnYm0r5pNlyT2bRjXh0JmHjY6hWwALEwIN" crossorigin="anonymous">
9     <title>File Upload</title>
10 </head>
11 <body>
12     <div class="container mt-3">
13         <h2>File Upload</h2>
14         <hr>
15
16         <form action="{{ url('/file-upload') }}" method="POST" enctype="multipart/form-data">
17             @csrf
18             <div class="mb-3">
19                 <label for="namaBaruFile" class="form-label">Nama File</label>
20                 <input type="text" class="form-control" placeholder="Masukkan Nama Baru Untuk File" id="namaBaruFile" name="namaBaruFile">
21                 <label for="berkas" class="form-label">Gambar Profile</label>
22                 <input type="file" class="form-control" id="berkas" name="berkas">
23                 @error('berkas')
24                     <div class="text-danger">{{ $message }}</div>
25                 @enderror
26             </div>
27
28             <button type="submit" class="btn btn-primary my-2">Upload</button>
29         </form>
30     </div>
31 </body>
32 </html>
```