

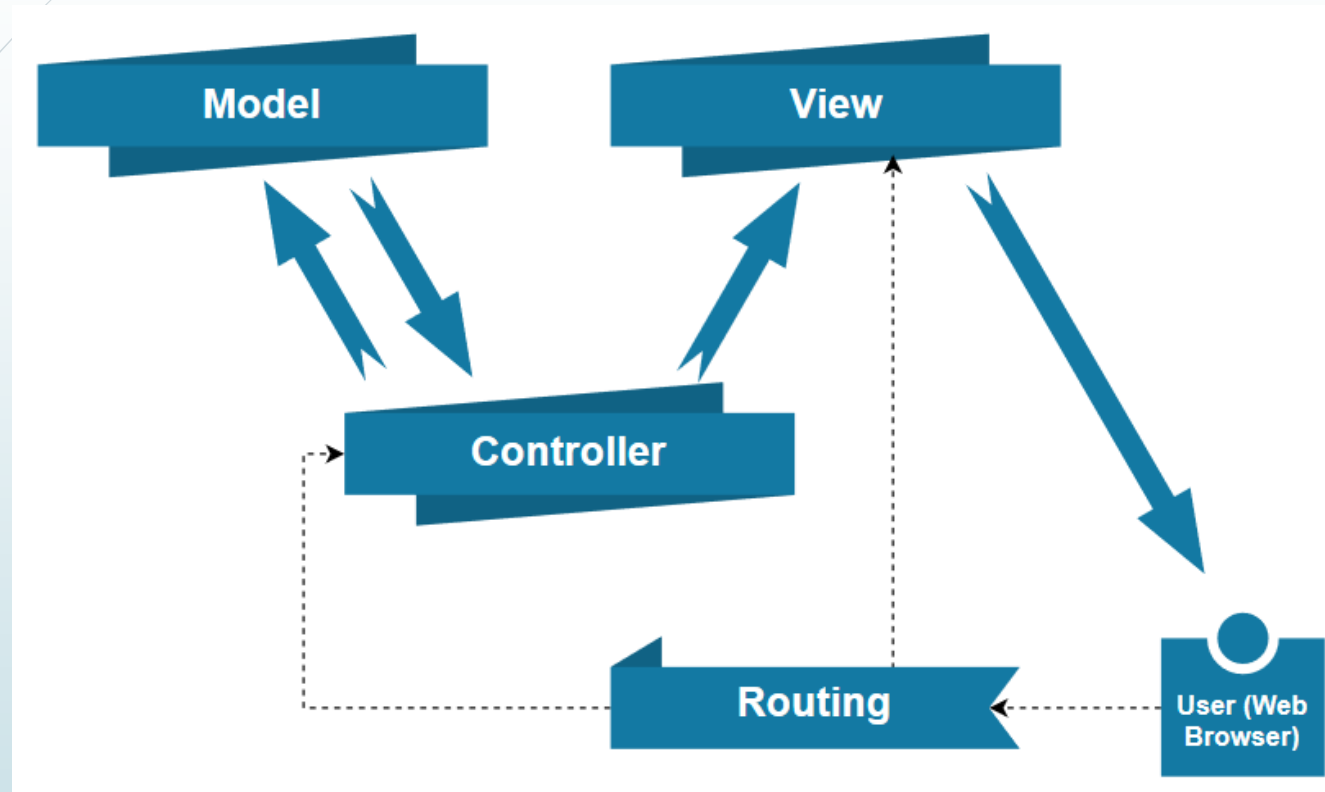


View

Lindung Siswanto

TI - POLNEP

Pengertian



- View adalah komponen MVC yang menangani tampilan → HTML, CSS, Javascript
- Dapat diakses langsung dari route → jika halaman statis



View Bawaan

- Welcome
- Disimpan pada folder resources\views
- Nama file → welcome.blade.php
- Penamaan view → <nama_file>.blade.php
- Blade → sebuah *templating engine bawaan Laravel*

resources > views >  mahasiswa.blade.php

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7  Route <title>Document</title>
8  </head>
9  <body>
10     <h1>Daftar Mahasiswa</h1>
11     <ol>
12         <li>Rudi Permana</li>
13         <li>Sari Citra Lestari</li>
14         <li>Rina Kumala Sari</li>
15         <li>James Situmorang</li>
16     </ol>
17 </body>
18 </html>
```

View

Sub Folder

- Buat folder baru bernama jurusan
- Pindahkan file mahasiswa.blade.php ke folder jurusan
- Ubah route sebagai berikut:

```
75     Route::get('/mahasiswa', function () {  
76         return view('jurusan.mahasiswa');  
77     });
```

- Tanda titik di dalam penulisan view() berfungsi sebagai penanda nama folder, bagaimana dengan '/'?
- Apa kesimpulan yang dapat diambil?

Mengirim data ke view

```
75 Route::get('/mahasiswa', function () {  
76     return view('jurusan.mahasiswa', ["mhs1"=>"Lukito Edi"]);  
77 });
```

view

```
10 <h1>Daftar Mahasiswa</h1>  
11 <ol>  
12 <li><?php echo $mhs1;?></li>  
13 <li>Sari Citra Lestari</li>  
14 <li>Rina Kumala Sari</li>  
15 <li>James Situmorang</li>  
16 </ol>  
17 </body>  
18 </html>
```

Mengirim array

```
75 Route::get('/mahasiswa', function () {  
76     $arrMahasiswa = [  
77         "mhs01" => "Lukito Edi",  
78         "mhs02" => "Rudiyanto",  
79         "mhs03" => "Kusumawardhani",  
80         "mhs04" => "Agung Laksono"  
81     ];  
82     return view('jurusan.mahasiswa', $arrMahasiswa);  
83 });
```

Mengirim array

```
10      <h1>Daftar Mahasiswa</h1>
11      <ol>
12          <li><?php echo $mhs01;?></li>
13          <li><?php echo $mhs02;?></li>
14          <li><?php echo $mhs03;?></li>
15          <li><?php echo $mhs04;?></li>
16      </ol>
17  </body>
18  </html>
```

View

Menggunakan foreach

```
75 Route::get('/mahasiswa', function () {  
76     $arrMahasiswa = ["Lukito Edi", "Rudiyanto",  
77     "Kusumawardhani", "Agung Laksono"];  
78     return view('jurusan.mahasiswa', ["mahasiswa" => $arrMahasiswa]);  
79 });
```

```
10 <h1>Daftar Mahasiswa</h1>  
11 <ol>  
12     <?php  
13         foreach ($mahasiswa as $nama)  
14             {  
15                 echo "<li>$nama</li>";  
16             }  
17     ?>  
18 </ol>
```

View

Cara lain

- Ganti kode baris 78 dengan kode berikut:

```
return view('jurusan.mahasiswa')->with('mahasiswa', $arrMahasiswa);
```

Latihan membuat route dan view

➡ `http://localhost/mahasiswa/Joko/20/Jakarta`

➡ Maka akan muncul :

Informasi mahasiswa

Nama : Joko

Umur : 20

Asal : Jakarta

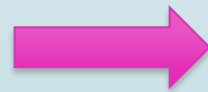
➡ Umur wajib berisi angka

➡ Jika asal tidak diisi, maka akan memunculkan pesan “Tidak diisi”

Blade Template

- Template engine berisi perintah tambahan untuk mempermudah pembuatan template. Template sendiri bisa disebut sebagai kerangka dasar tampilan.
- Fungsi Utama blade:
 - Mempersingkat penulisan perintah PHP.
 - Pemecahan file template (proses pembuatan layout).

```
<?php echo $nama; ?>
```



```
{{ $nama }}
```

Perbandingan

PHP Syntax	Blade Syntax
<code><?php echo \$var; ?></code>	<code>{{ \$var }}</code>
<code><?php echo htmlentities(\$var); ?></code>	<code>{{{ \$var }}}}</code>
<code><?php if(\$cond) : ?> .. <?php endif; ?></code>	<code>@if(\$cond) .. @endif</code>

Echo

```
/*script php*/  
<?php echo "Halo ini cara lama" ?>
```

```
{{"Halo ini pakai blade"}}
```

Echo variable dengan html escape

```
/*script php*/  
Halo ini cara lama <?php echo htmlspecialchars($name, ENT_QUOTES, 'UTF-8')  
?>
```

```
Nama mahasiswa adalah {{{$mahasiswa}}}
```

Echo dengan nilai default

```
/*script php*/  
Halo ini cara lama <?php echo isset ($name)? ($name): 'guest' ?>
```

```
/*cara baru dengan menggunakan blade template*/  
Halo ini cara blade template {{ $name or 'guest' }}
```


Kondisi

```
/*script php*/  
<?php  
if (status==0) {  
    echo "Proses Gagal";  
} elseif (status==1) {  
    echo "Proses Berhasil";  
} else {  
    echo "Tidak diketahui";  
}  
?>
```

```
@if($status==0)  
{{"Proses gagal"}}  
@elseif($status==1)  
{{"Proses berhasil"}}  
@else  
{{"Tidak diketahui"}}  
@endif
```

Pengulangan for

```
/* for statement*/  
<?php  
    for ($i = 1; $i <= 10; $i++) {  
        echo $i;  
    }  
?>
```

```
/* for statement */  
@for ($i = 1; $i <= 10; $i++)  
    {{ $i }}  
@endfor
```

Pengulangan While

```
<?php while ($i &lt;= 10) :  
    echo $i++;  
endwhile;  
?>
```

```
@while ($i &lt;= 10)  
    {{ $i ++ }}  
@endwhile
```

Pengulangan foreach

```
<?php
foreach($arr as $value)
{
    $value=$value + 1;
    echo $value;
}
?>
```

```
@foreach($arr as $value)
{{$value}}
@endforeach
```