

Pemrograman Web (PHP Framework - Laravel)

WILLI PRATAMA, S.Kom
AMIK Bumi Nusantara Cirebon (2020)

Pertemuan 9



Laravel Authentication & Authorization

Otentikasi Sistem Pada Laravel Framework

Proses pembuatan otentikasi (login / register) pengguna di dalam sebuah sistem Laravel bisa diimplementasikan dengan mudah.

Konfigurasi proses otentikasi baik itu dari sisi web (yang pengguna langsung akses melalui peramban internet) atau dari sisi API (Application Programming Interface) berada di berkas auth.php yang ada di folder config.

Laravel menggunakan 2 buah fasilitas dalam proses otentikasi, yaitu **guards** dan **providers**.

Guards di dalam laravel diartikan sebagai penjaga otentikasi, dalam hal ini berkaitan dengan informasi yang disimpan untuk digunakan sebagai data pengguna yang terotentikasi.

Providers di dalam laravel diartikan sebagai penyedia data dalam fungsi otentikasi di dalam laravel.

Providers adalah bagian dari **guards**, di mana setiap 'penjaga' otentikasi di dalam laravel akan mengambil informasi data pengguna dari sebuah 'penyedia' melalui sebuah kelas model.

Pertimbangan Basis Data Untuk Otentikasi Laravel

Laravel menyediakan sebuah model bernama **User.php** di dalam folder **App** yang bisa digunakan sebagai penyedia data otentikasi pengguna.

Jika ingin menggunakan provider lain selain User.php, pastikan model tersebut memiliki:

- Sebuah kolom bernama **email** yang bertipe data string (varchar) dan unik.
- Sebuah kolom bernama **email_verified_at** bertipe data timestamp.
- Sebuah kolom bernama **password** yang berisi panjang minimal 60 karakter string (varchar).
- Untuk memverifikasi sebuah pengguna telah ter-otentikasi ketika masuk ke dalam aplikasi laravel di sesi yang berbeda, pastikan ada sebuah kolom bernama **remember_token** yang berisi panjang 100 karakter string dengan karakteristik boleh dikosongkan (nullable).

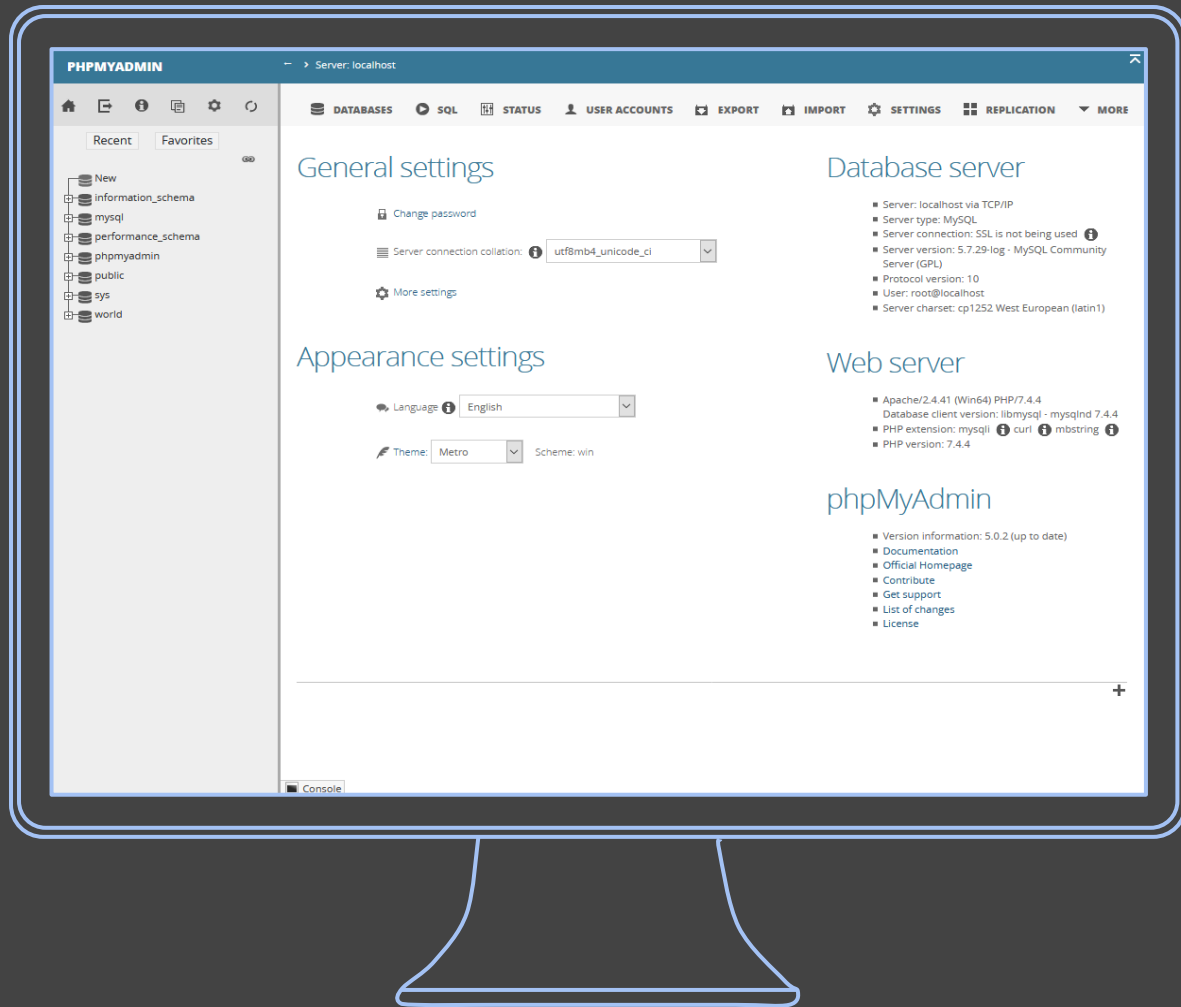
Mari Praktek!



Unduh berkas aset berikut ini:

<https://www.dropbox.com/s/frdlojywlxl8owo/img.zip?dl=0>

Ekstrak .zip ke dalam folder **public/img**
(jika belum ada folder **img**, buat baru folder tersebut).



Buka PHPMYADMIN

Buat sebuah database baru dan beri nama "laravel_pertemuan7"

Nama database:

laravel_pertemuan7

Format database:

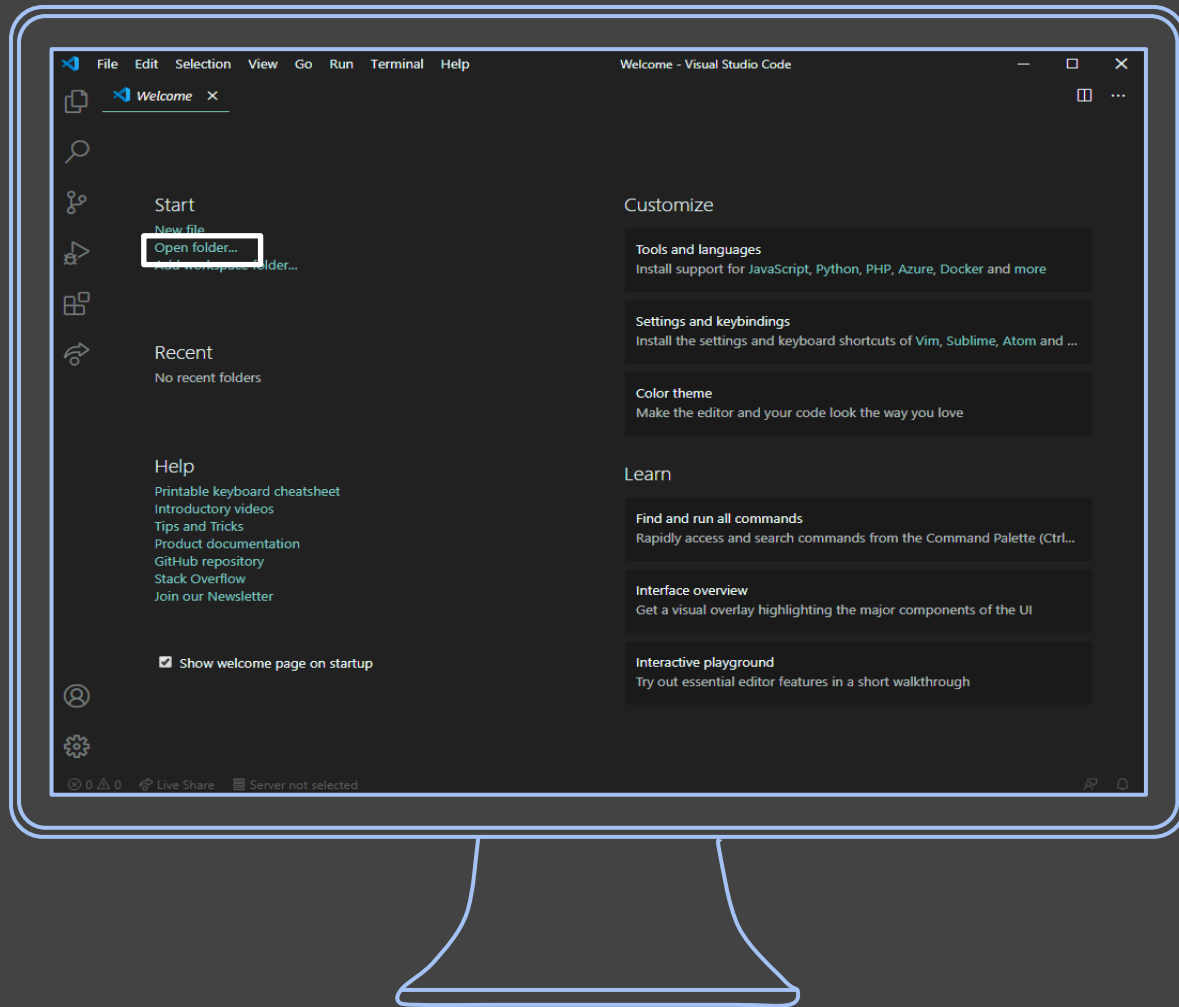
utf8mb4_general_ci


```
C:\>laravel> composer create-project --prefer-dist laravel/laravel pertemuan7
"7.*"
Creating a "laravel/laravel" project at "./pertemuan7"
Installing laravel/laravel (v7.30.1)
- Installing laravel/laravel (v7.30.1): Loading from cache
Created project in C:\\laravel\\pertemuan7
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 103 installs, 0 updates, 0 removals
- Installing voku/portable-ascii (1.5.6): Loading from cache
- Installing symfony/polyfill-ctype (v1.20.0): Loading from cache
- Installing phoption/phoption (1.7.5): Loading from cache
[...]
Generating optimized autoload files
[...]
72 packages you are using are looking for funding.
[...]
Package manifest generated successfully.
Application key set successfully.
C:\>laravel>
```

Buat Proyek Laravel Baru

Beri nama foldernya pertemuan7

```
composer create-project --prefer-dist  
laravel/laravel pertemuan7 "7.*"
```



Buka Visual Studio Code

Klik "Open folder..." lalu pilih folder laravel "pertemuan7" yang sebelumnya telah dibuat.



```
APP_NAME=Laravel
```

```
[...]
```

```
DB_CONNECTION=mysql
```

```
DB_HOST=127.0.0.1
```

```
DB_PORT=3306
```

```
DB_DATABASE=laravel_pertemuan7
```

```
DB_USERNAME=root
```

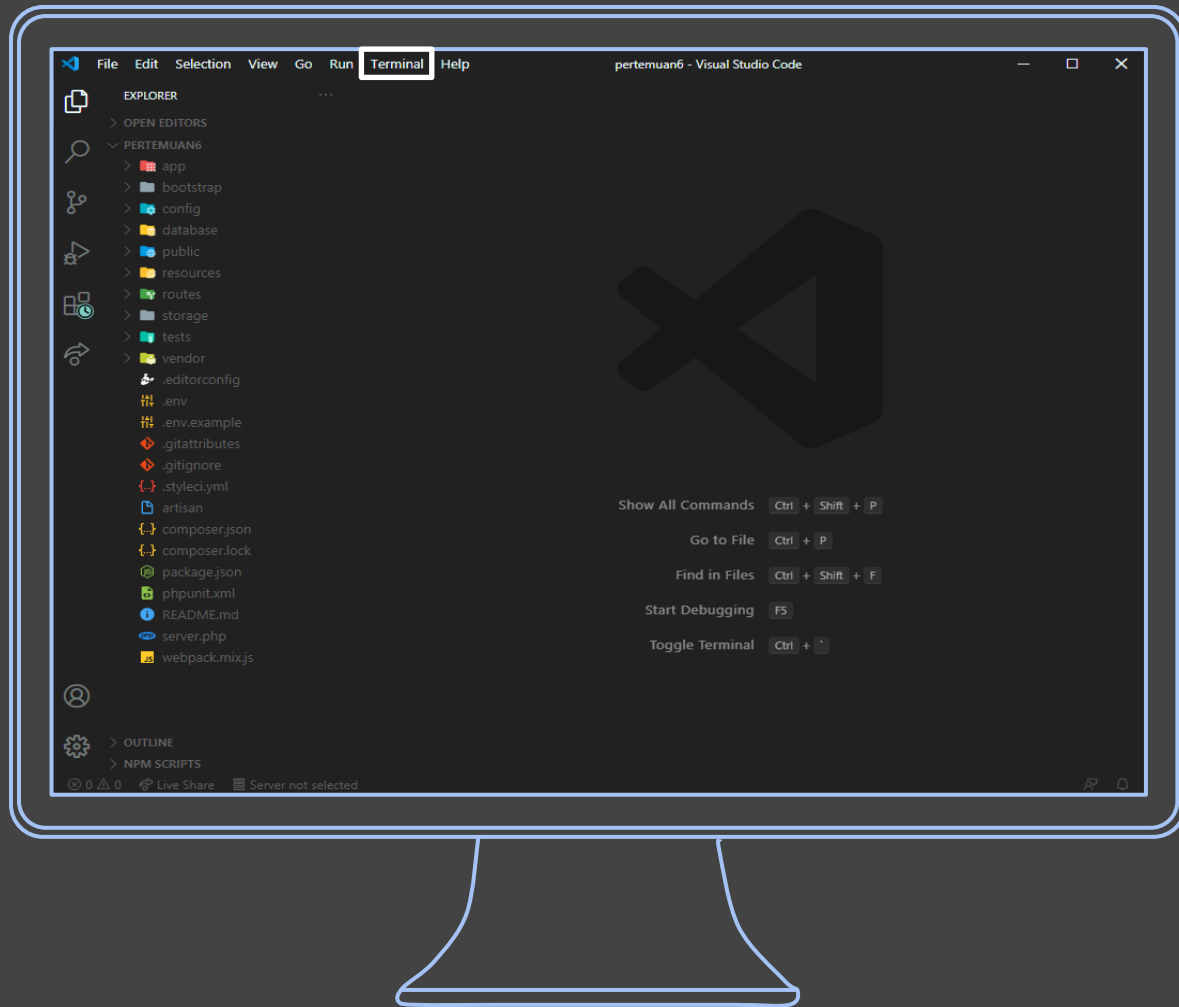
```
DB_PASSWORD=
```

```
[...]
```

dotENV | Konfigurasi Basis Data

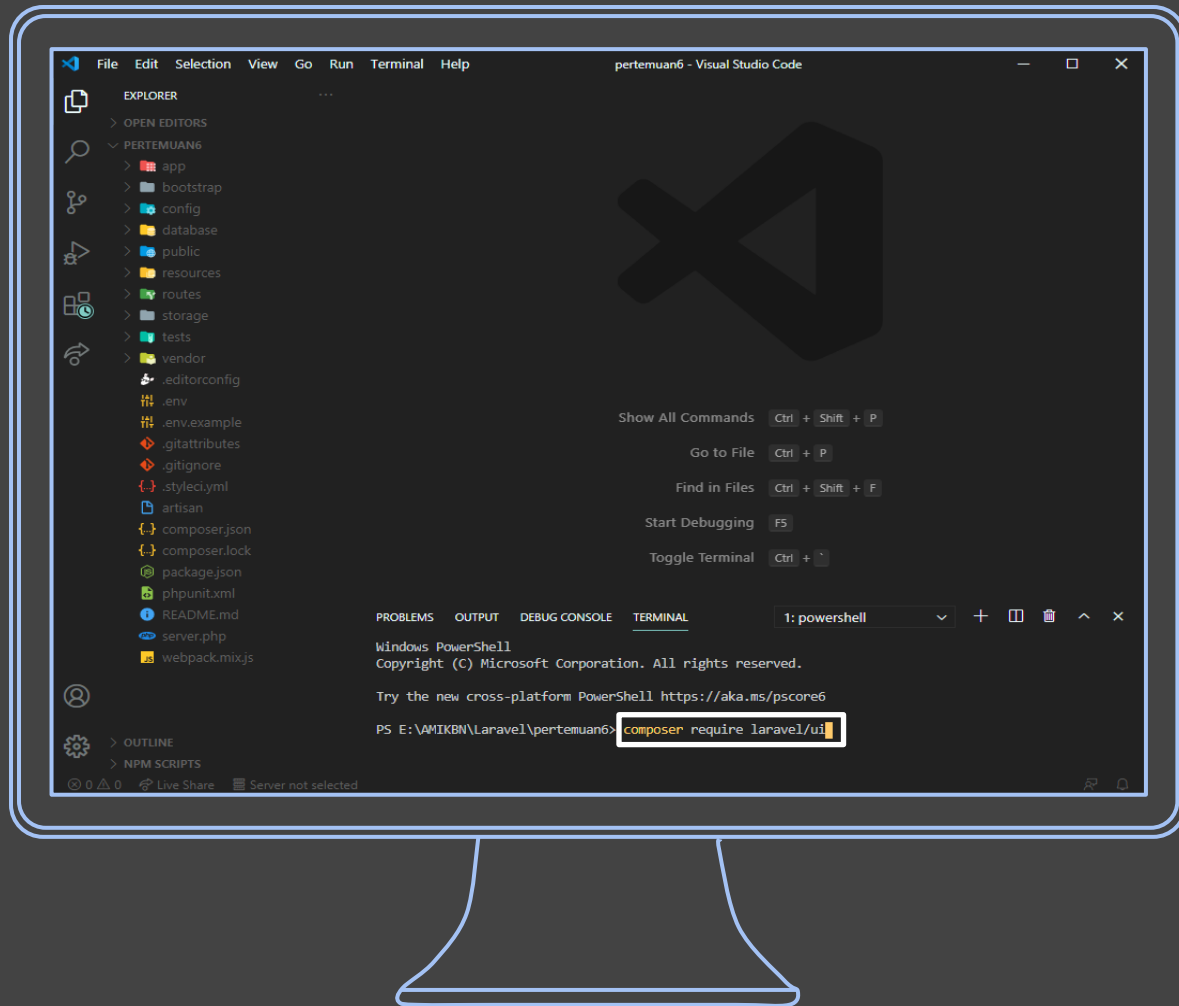
Buka berkas **.env** dan ubah bagian yang ada di samping sesuai konfigurasi di sistem operasi kalian.

Ganti isi **DB_DATABASE** menjadi **laravel_pertemuan7**.



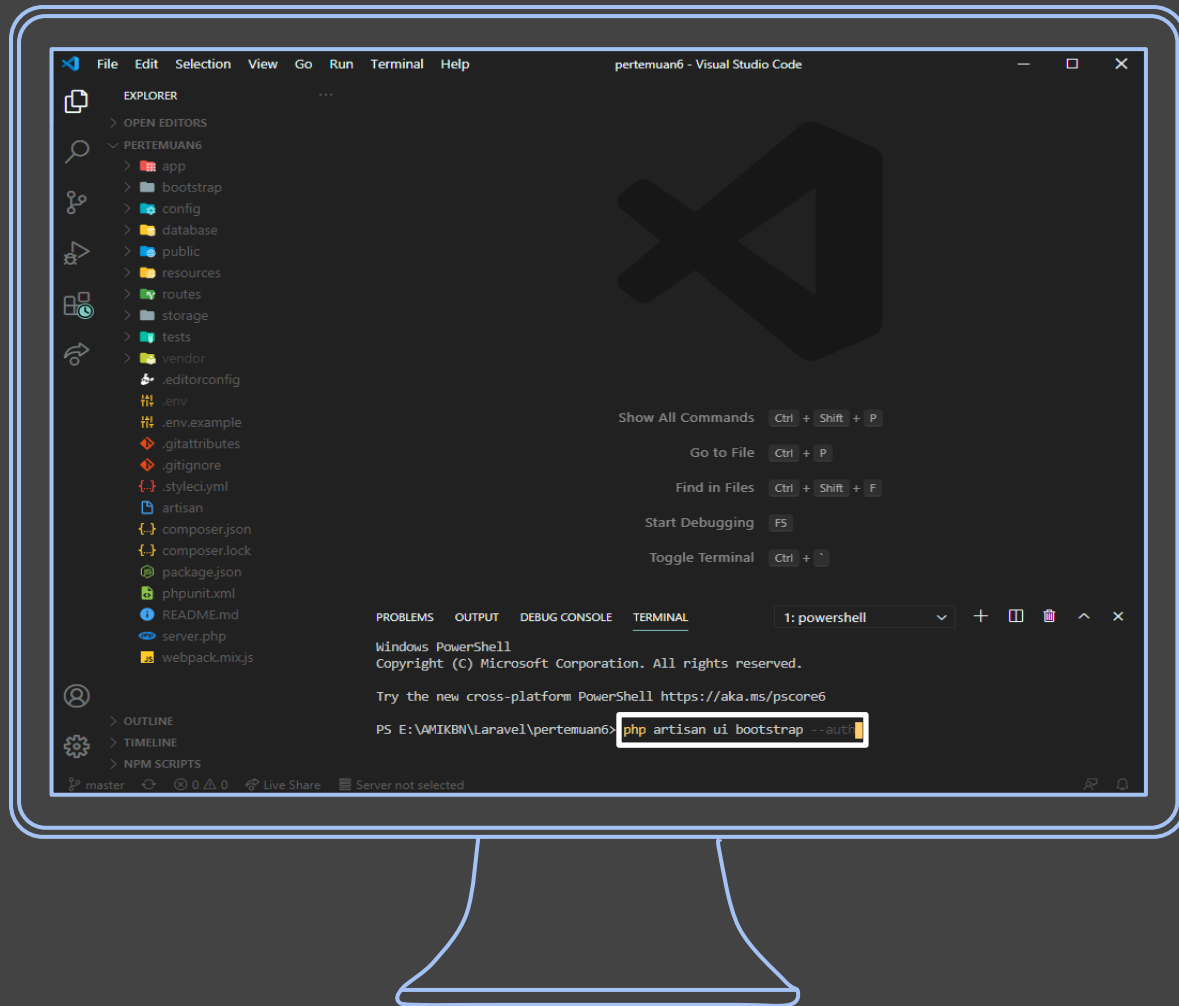
Buka terminal / cmd baru di
Visual Studio Code

Klik menu **Terminal** > **New
Terminal**



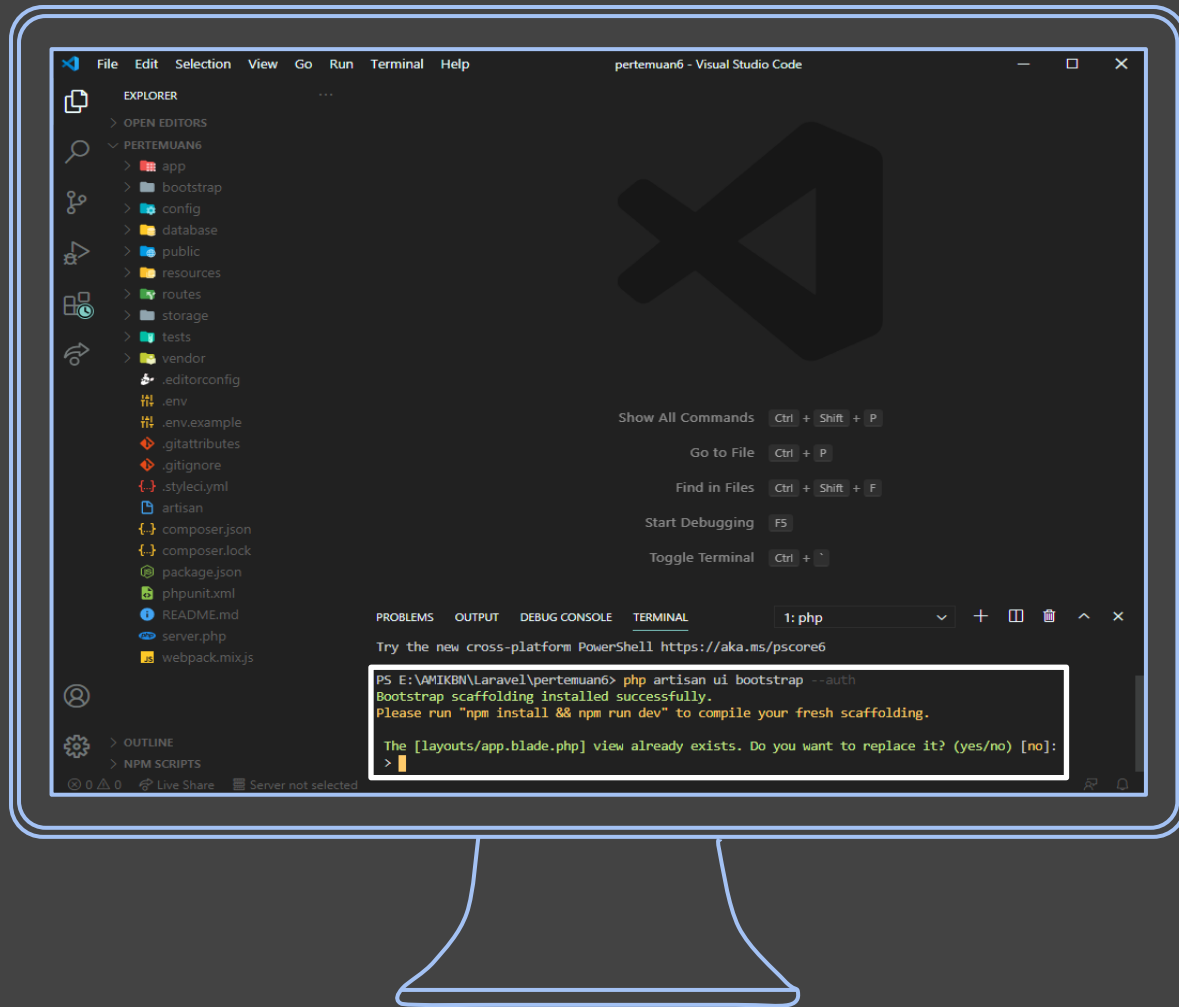
Pemasangan Paket Composer LARAVEL/UI

Di terminal yang sudah terbuka, ketik perintah **composer require laravel/ui:^2.4**



Frontend Scaffolding Bootstrap + Authentications

Di terminal yang sudah terbuka,
ketik perintah
**php artisan ui
bootstrap --auth**



php artisan ui --auth |
app.blade.php sudah ada di
dalam folder

Jika sebelumnya sudah pernah
membuat frontend scaffolding,
jangan lupa isi dari
app.blade.php (ketik **no** ketika
muncul masukan)

Sistem Yang Dipakai Untuk Keperluan Pribadi

Karena sistem yang kita akan bangun adalah untuk keperluan pribadi, maka tidak perlu otentikasi untuk banyak pengguna, cukup informasi pribadi kita saja yang disimpan di basis data.

Selanjutnya kita akan membuat migrasi basis data untuk menyimpan informasi pribadi sehingga kita tidak perlu melakukan permintaan pembuatan otentikasi pengguna lebih dari satu di lain waktu.


```
2014_10_12_000000_create_users_table.php - pertemuan6 - Visual Studio Code

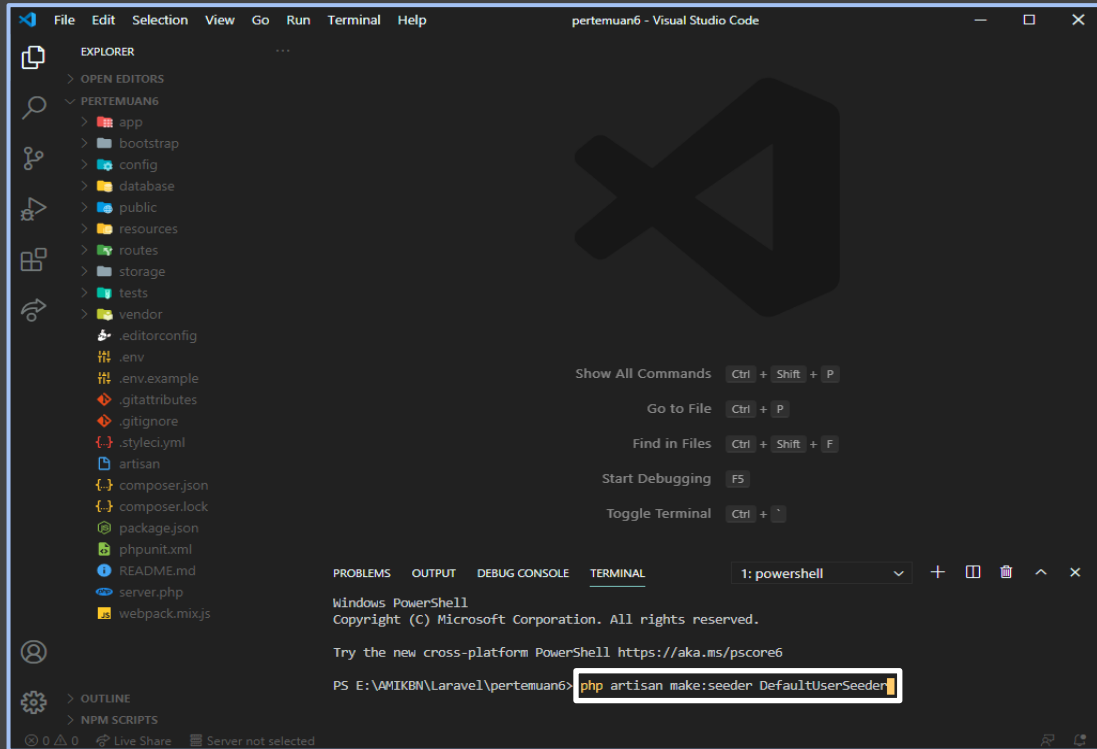
database > migrations > 2014_10_12_000000_create_users_table.php > CreateUsersTable > up

1 <?php
2
3 use Illuminate\Database\Migrations\Migration;
4 use Illuminate\Database\Schema\Blueprint;
5 use Illuminate\Support\Facades\Schema;
6
7 class CreateUsersTable extends Migration
8 {
9     /**
10      * Run the migrations.
11      *
12      * @return void
13      */
14     public function up()
15     {
16         Schema::create('users', function (Blueprint $table) {
17             $table->id();
18             $table->string('name');
19             $table->string('email')->unique();
20             $table->timestamp('email_verified_at')->nullable();
21             $table->string('password');
22             $table->rememberToken();
23             $table->timestamps();
24         });
25     }
26
27     /**
28      * Reverse the migrations.
29      *
30      * @return void
31      */
32     public function down()
33     {
34         Schema::dropIfExists('users');
35     }
36 }
```

Berkas migrasi create_users_table.php

Buka berkas
2014_10_12_000000_create_users_table.php
di dalam folder
database/migrations.

Pahami kolom-kolom yang ada
di dalam tabel users.



Buat penyemai data untuk tabel users

Di terminal yang sudah terbuka, ketik perintah **php artisan make:seeder DefaultUserSeeder**

```
<?php

use Carbon\Carbon;
use Illuminate\Support\Facades\DB;
use Illuminate\Database\Seeder;
use Illuminate\Support\Facades\Hash;

class DefaultUserSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        DB::table('users')->insert([
            'id' => 1,
            'name' => 'Nama Lengkap',
            'email' => 'E-mail aktif',
            'email_verified_at' => Carbon::now(),
            'password' => Hash::make('ketik password yang diinginkan'),
            'created_at' => Carbon::now(),
        ]);
    }
}
```

Buat penyemai data untuk tabel users

Buka berkas

DefaultUserSeeder.php di dalam
folder **database/seeds**.

Isikan kodenya seperti di
samping berikut.

```
<?php

use Illuminate\Database\Seeder;

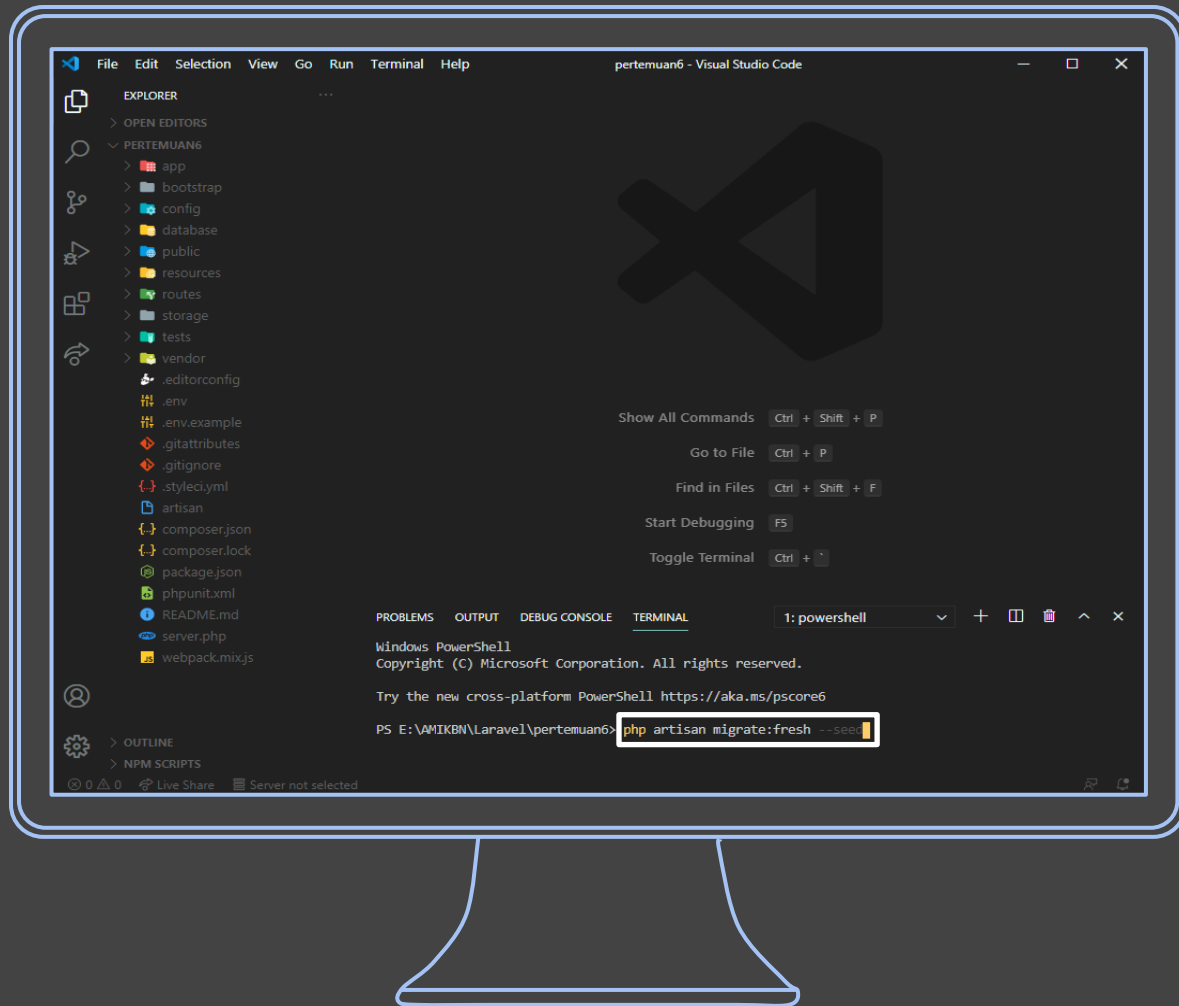
class DatabaseSeeder extends Seeder
{
    /**
     * Seed the application's database.
     *
     * @return void
     */
    public function run()
    {
        $this->call(DefaultUserSeeder::class);
    }
}
```

Buat penyemai data untuk tabel users

Buka berkas

DatabaseSeeder.php di dalam
folder **database/seeds**.

Isikan kodenya seperti di
samping berikut.



Migrasi basis data beserta penyemai data

Di terminal yang sudah terbuka,
ketik perintah
**php artisan migrate:fresh
--seed**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Website Pribadi</title>

  <!-- Bootstrap + Font Awesome Free CSS -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.14.0/css/all.min.css" />

</head>
<body>
  @include('layouts.navbar')

  <main role="main" class="p-4" style="min-height: calc(100vh - 112px)">
    @yield('content')
  </main>

  @include('layouts.footer')

  <!-- Bootstrap + Font Awesome + jQuery + Popper.js JS -->
  <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
  <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"></script>
  <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
  <script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.14.0/js/all.min.js"></script>

  @yield('script')
</body>
</html>
```

Frontend Scaffolding Bootstrap + Authentications

Buka berkas **app.blade.php** di dalam folder **layouts**.

Isikan kodenya seperti di samping berikut.

```

<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
  <a class="navbar-brand">
    
    Website Pribadiku
  </a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarContent" aria-controls="navbarContent" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>

  <div class="collapse navbar-collapse" id="navbarContent">
    <ul class="navbar-nav ml-auto">
      &nbsp;
      <li class="nav-item">
        <a class="btn {{ Route::is('beranda') ? 'btn-primary' : 'btn-outline-primary' }} btn-block" href="{{ route('beranda') }}"><i class="fas fa-home"></i> Beranda</a>
      </li>
      &nbsp;
      <li class="nav-item">
        <a class="btn {{ Route::is('hobi') ? 'btn-primary' : 'btn-outline-primary' }} btn-block" href="{{ route('hobi') }}"><i class="fas fa-photo-video"></i> Hobi</a>
      </li>
      &nbsp;
    </ul>
  </div>

```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **layouts** dan beri nama **navbar.blade.php**

Isikan kodenya seperti di samping berikut.

```
<li class="nav-item">
  <a class="btn {{ Route::is('tentang') ? 'btn-primary' : 'btn-outline-
  primary' }} btn-block" href="{{ route('tentang') }}"><i class="fas fa-
  address-book"></i> Tentang Saya</a>
</li>
&nbsp;
<li class="nav-item">
  <a class="btn btn-info btn-
  block" href="{{ route('login') }}"><i class="fas fa-sign-in-
  alt"></i> Masuk</a>
</li>
</ul>
</div>
</nav>
```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **layouts** dan beri nama **navbar.blade.php**

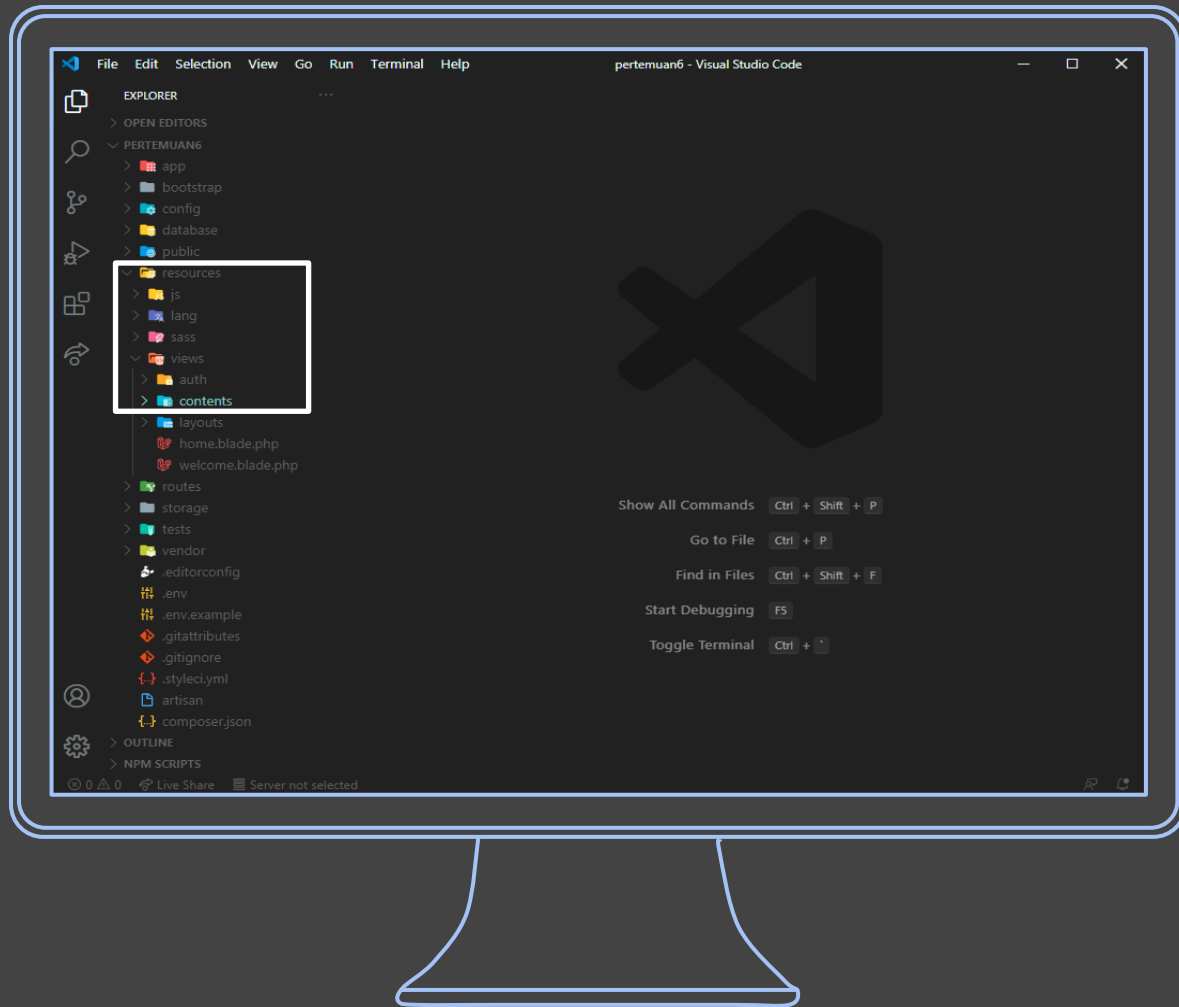
Isikan kodenya seperti di samping berikut.


```
<footer class="footer bg-dark text-light text-center">
  <div class="container py-3">
    © 2020, dibuat dengan <i class="fa fa-heart text-
      danger"></i> menggunakan <i class="fab fa-laravel text-
      danger"></i> | <a href="http://www.amikbn.ac.id/">AMIK Bumi Nusa
      ntara Cirebon</a>
  </div>
</footer>
```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **layouts** dan beri nama **footer.blade.php**

Isikan kodenya seperti di samping berikut.



Frontend Scaffolding Bootstrap + Authentications

Buka folder **resources/views**
lalu buat folder baru di
dalamnya dan beri nama
contents

```
@extends('layouts.app')

@section('content')
<div class="container" style="height: 10vh"></div>
<div class="row">
    <div class="col text-center">
        
        <h2 class="text-
        uppercase">{{ env('USER_NAME') }}</h2>
        <hr/>
        <p class="font-
        italic">{{ env('USER_MOTTO') }}</p>
    </div>
</div>
@endsection
```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **contents** dan beri nama **beranda.blade.php**

Isikan kodenya seperti di samping berikut.

```
@extends('layouts.app')

@section('content')
<div class="container">
    <div class="row justify-content-center align-self-center">
        <div class="col-lg-4 col-md-6 col-sm-12" style="margin-bottom:20px">
            <div class="card border-dark">
                
                <div class="card-body">
                    <h5 class="card-title">Touring</h5>
                    <p class="card-text">Nyeruntul Bareng Sedulur.</p>
                </div>
            </div>
        </div>
        <div class="col-lg-4 col-md-6 col-sm-12" style="margin-bottom:20px">
            <div class="card border-dark">
                
                <div class="card-body">
                    <h5 class="card-title">Full Stack Developer</h5>
                    <p class="card-text">Keep Calm and Enjoy Code!</p>
                </div>
            </div>
        </div>
    </div>
</div>
```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **contents** dan beri nama **hobi.blade.php**

Isikan kodenya seperti di samping berikut.

```
        </div>
    </div>
</div>

<div class="col-lg-4 col-md-6 col-sm-12" style="margin-bottom:20px">
    <div class="card border-dark">
        
        <div class="card-body">
            <h5 class="card-title">Part Time Gamer</h5>
            <p class="card-text">Glorious PC Master Race!</p>
        </div>
    </div>
</div>
</div>
</div>
@endsection
```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **contents** dan beri nama **hobi.blade.php**

Isikan kodenya seperti di samping berikut.

```

@extends('layouts.app')

@section('content')
<div class="container">
    <div class="card border-dark">
        <div class="card-header bg-dark text-light text-uppercase">
            Tentang Saya
        </div>
        <div class="card-body">
            <div class="row">
                <div class="col-lg-4 col-md-12 col-sm-12 text-center">
                    
                    <h2 class="text-uppercase">{{ env('USER_NAME') }}</h2>
                    <hr/>
                    <p class="font-italic">{{ env('USER_MOTTO') }}</p>
                </div>
                <div class="col-lg-8 col-md-12 col-sm-12">
                    <div class="card">
                        <h5 class="card-header">Kemampuan Saya</h5>
                        <div class="card-body">
                            <ul class="list-group">
                                <li class="list-group-item">

```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **contents** dan beri nama **tentang.blade.php**

Isikan kodenya seperti di samping berikut.

```
<div class="row" data-
toggle="tooltip" data-
placement="left" title="PHP">
  <div class="col-2 text-center">
    <i class="fab fa-2x fa-
php" style="color: #4D598F"></i>
  </div>
  <div class="col-10">
    <div class="progress border border-
light" style="height: 34px;">
      <div class="progress-
bar" role="progressbar" style="
width: 95%; background-
color: #4D598F;" aria-
valuenow="95" aria-
valuemin="0" aria-
valuemax="100">95</div>
    </div>
  </div>
</li>
<li class="list-group-item">
```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **contents** dan beri nama **tentang.blade.php**

Isikan kodenya seperti di samping berikut.

```
<div class="row" data-
toggle="tooltip" data-
placement="left" title="Laravel Framework"
>
  <div class="col-2 text-center">
    <i class="fab fa-2x fa-
laravel" style="color: #F44E39"></i>
  </div>
  <div class="col-10">
    <div class="progress" style="height
: 34px;">
      <div class="progress-
bar" role="progressbar" style="
width: 88%; background-
color: #F44E39;" aria-
valuenow="88" aria-
valuemin="0" aria-
valuemax="100">88%</div>
    </div>
  </div>
</li>
<li class="list-group-item">
```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **contents** dan beri nama **tentang.blade.php**

Isikan kodenya seperti di samping berikut.


```
<div class="row" data-
toggle="tooltip" data-
placement="left" title="Twitter Bootstrap"
>
  <div class="col-2 text-center">
    <i class="fab fa-2x fa-
bootstrap" style="color: #563D7C"><
/i>
  </div>
  <div class="col-10">
    <div class="progress" style="height
: 34px;">
      <div class="progress-
bar" role="progressbar" style="
width: 90%; background-
color: #563D7C;" aria-
valuenow="90" aria-
valuemin="0" aria-
valuemax="100">90%</div>
    </div>
  </div>
</div>
</li>
</ul>
</div>
```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **contents** dan beri nama **tentang.blade.php**

Isikan kodenya seperti di samping berikut.



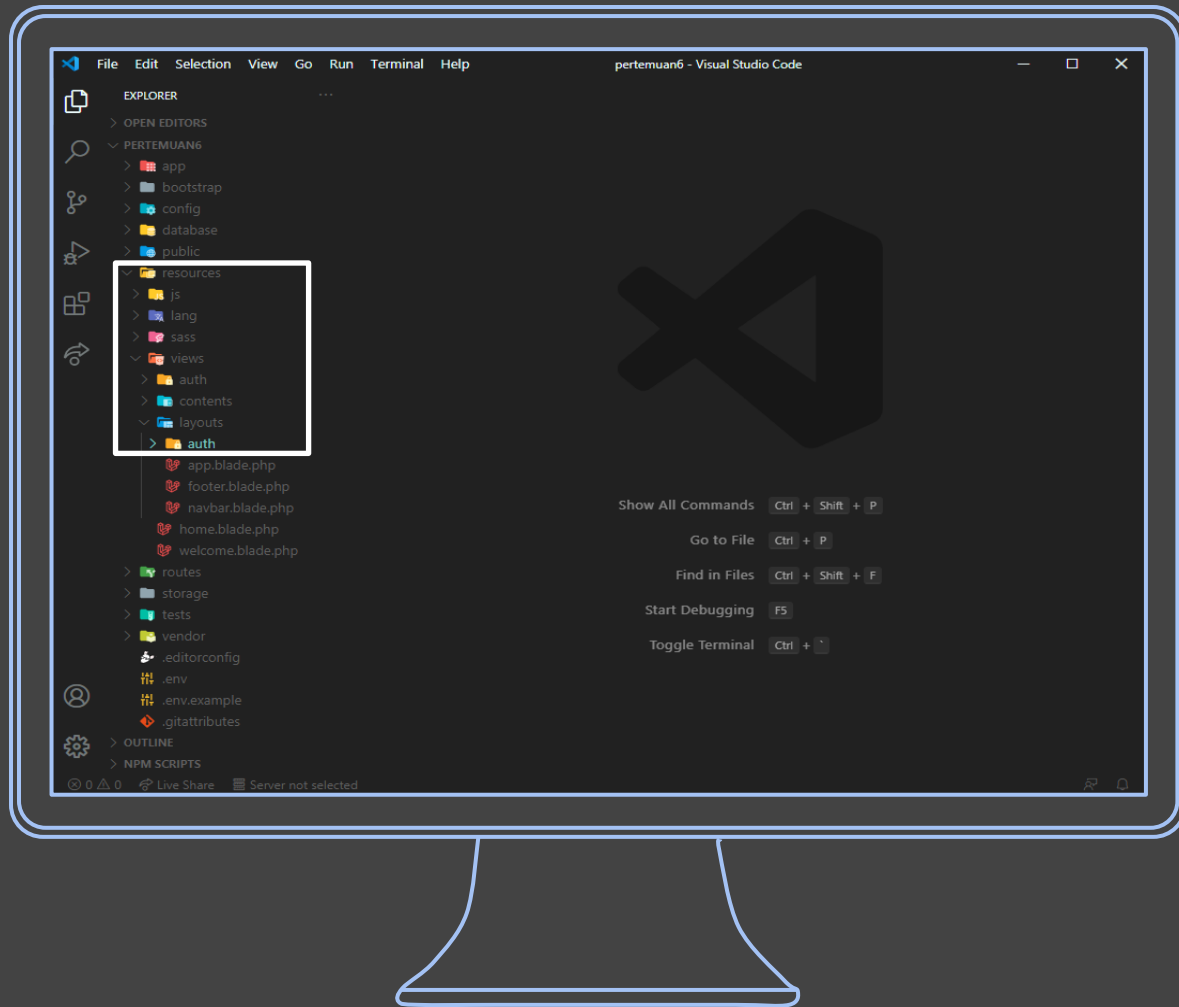
```
        </div>
    </div>
</div>
</div>
</div>
</div>
@endsection

@section('script')
<script>
$(document).ready(function(){
    $('[data-toggle="tooltip"]').tooltip();
});
</script>
@endsection
```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **contents** dan beri nama **tentang.blade.php**

Isikan kodenya seperti di samping berikut.



Frontend Scaffolding Bootstrap + Authentications

Buka folder
resources/views/layouts lalu
buat folder baru di dalamnya
dan beri nama **auth**

```

<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
  <a class="navbar-brand">
    
    Website Pribadiku
  </a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarContent" aria-controls="navbarContent" aria-expanded="false" aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
  </button>

  <div class="collapse navbar-collapse" id="navbarContent">
    <ul class="navbar-nav">
      &nbsp;
      <li class="nav-item">
        <a class="btn {{ Route::is('beranda') ? 'btn-primary' : 'btn-outline-primary' }} btn-block" href="{{ route('beranda') }}"><i class="fas fa-home"></i> Beranda</a>
      </li>
      &nbsp;
      <li class="nav-item">
        <a class="btn {{ Route::is('hobi') ? 'btn-primary' : 'btn-outline-primary' }} btn-block" href="{{ route('hobi') }}"><i class="fas fa-photo-video"></i> Hobi</a>
      </li>
      &nbsp;
    </ul>
  </div>

```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder **resources/views/layouts/auth** dan beri nama **navbar.blade.php**

Isikan kodenya seperti di samping berikut.

```

        <li class="nav-item">
            <a class="btn {{ Route::is('tentang') ? 'btn-primary' : 'btn-outline-
            primary' }} btn-block" href="{{ route('tentang') }}"><i class="fas fa-
            address-book"></i> Tentang Saya</a>
        </li>
    </ul>
    <ul class="navbar-nav ml-auto">
        &nbsp;
        <li class="nav-item">
            <a class="btn {{ Route::is('home') ? 'btn-success' : 'btn-outline-
            success' }} btn-block" href="{{ route('home') }}"><i class="fas fa-
            tachometer-alt"></i> Dasbor</a>
        </li>
        &nbsp;
        <li class="nav-item">
            <a class="btn btn-danger btn-
            block" href="{{ route('logout') }}" onclick="event.preventDefault(); docum
            ent.getElementById('logout-form').submit();"><i class="fas fa-sign-out-
            alt"></i> Keluar</a>
        </li>
        <form id="logout-form" action="{{ route('logout') }}" method="POST"
            style="display: none;"
            @csrf
        </form>
    </ul>
</div>
</nav>

```

Frontend Scaffolding Bootstrap + Authentications

Buat berkas blade di dalam folder
resources/views/layouts/auth
dan beri nama **navbar.blade.php**

Isikan kodenya seperti di samping
berikut.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Website Pribadi</title>

  <!-- Bootstrap + Font Awesome Free CSS -->
  <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" />
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.14.0/css/all.min.css" />
</head>
<body>
  @guest
  @include('layouts.navbar')
  @else
  @include('layouts.auth.navbar')
  @endguest

  <main role="main" class="p-4" style="min-height: calc(100vh - 112px)">
    @yield('content')
  </main>

  @include('layouts.footer')
```

Frontend Scaffolding Bootstrap + Authentications

Buka berkas **app.blade.php** di dalam folder **layouts**.

Isikan kodenya seperti di samping berikut.

```
<!-- Bootstrap + Font Awesome + jQuery + Popper.js JS -->
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.14.0/js/all.min.js"></script>

@yield('script')
</body>
</html>
```

Frontend Scaffolding Bootstrap + Authentications

Buka berkas **app.blade.php** di dalam folder **layouts**.

Isikan kodenya seperti di samping berikut.

```
<?php

use Illuminate\Support\Facades\Route;

/*
|-----
| Web Routes
|-----
|
| Here is where you can register web routes for your application. These
| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something great!
|
*/

Auth::routes(
    [
        'register' => false,
    ]
);

Route::get('/', function () {
    return view('contents.beranda');
})->name('beranda');

Route::get('/hobi', function () {
    return view('contents.hobi');
})->name('hobi');

Route::get('/tentang', function () {
    return view('contents.tentang');
})->name('tentang');

Route::get('/home', 'HomeController@index')->name('home');
```

Ubah Web Routing

Buka **web.php** yang ada di folder **routes** dan ubah isinya seperti di samping.

Auth::routes

Jika kita ingin mengkonfigurasi rute otentikasi, seperti menonaktifkan rute pendaftaran pengguna baru, bisa mengubah hal-hal tersebut di dalam array

```
Auth::routes(  
    [  
        // konfigurasi  
    ]  
);
```

Konfigurasi rute yang bisa diubah antara lain:

- 'register' => true/false,
 // mengontrol rute pendaftaran pengguna
- 'reset' => true/false,
 // mengontrol rute reset password pengguna
- 'verify' => true/false,
 // mengontrol rute verifikasi email pengguna

```
File Edit Selection View Go Run Terminal Help
HomeController.php - pertemuan6 - Visual Studio Code

EXPLORER
  OPEN EDITORS
  PERTEMUAN6
  app
    Console
    Exceptions
    Http
      Controllers
        Auth
          Controller.php
          HomeController.php
        Middleware
        Kernel.php
      Providers
        User.php
    bootstrap
    config
    database
    public
    resources
    routes
    storage
    tests
    vendor
    .editorconfig
    .env
    .env.example
    .gitattributes
    .gitignore
    .styleci.yml
  OUTLINE
  NPM SCRIPTS

app > Http > Controllers > HomeController.php > HomeController > __construct
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  class HomeController extends Controller
8  {
9
10     /**
11      * Create a new controller instance.
12      *
13      * @return void
14      */
15     public function __construct()
16     {
17         $this->middleware('auth');
18     }
19
20     /**
21      * Show the application dashboard.
22      *
23      * @return \Illuminate\Contracts\Support\Renderable
24      */
25     public function index()
26     {
27         return view('home');
28     }
29 }
```

Detail HomeController

Buka **HomeController.php** yang ada di folder **app/Http/Controllers**.

Fungsi – Fungsi Yang Ada Di Dalam HomeController

`function __construct()`

- Fungsi **bawaan** PHP yang membolehkan developer untuk **mendeklarasikan kelas-kelas untuk metode pembangun baris kode**. Contoh di HomeController adalah adanya **middleware auth**, di mana akses ke controller ini hanya bisa diakses oleh pengguna yang sudah terotentikasi.

`function index()`

- Fungsi penamaan standar yang menjadi indikator bahwa fungsi ini adalah fungsi pertama di dalam hirarki sebuah percabangan alamat pada sebuah controller.

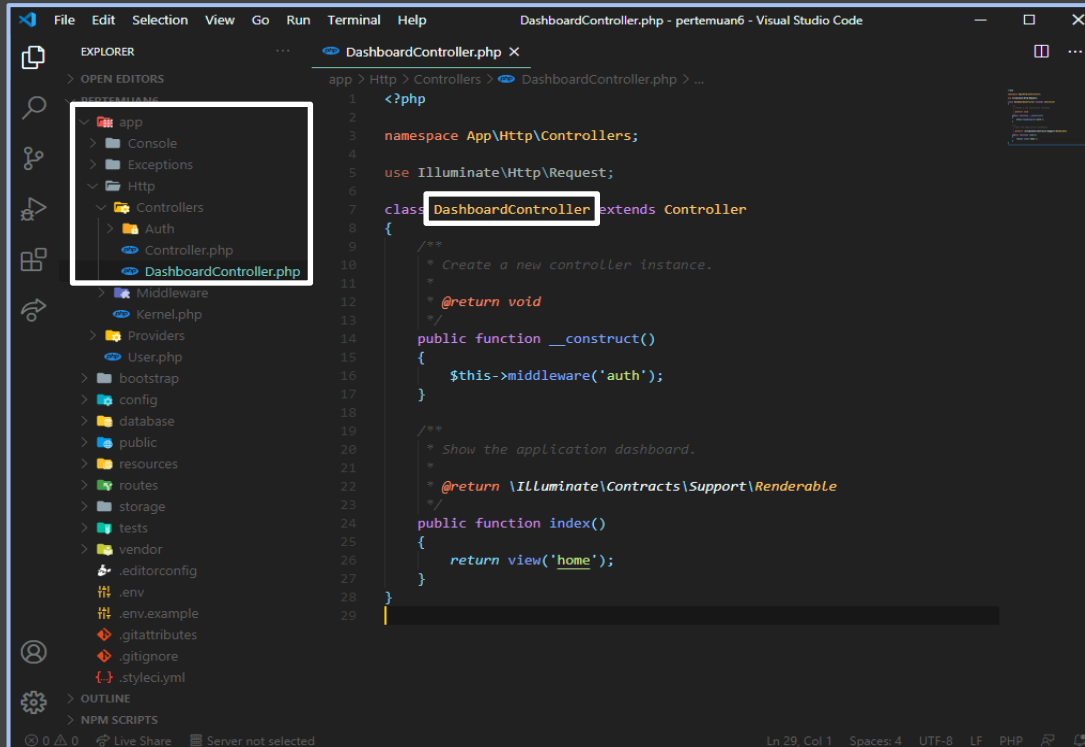


Rute **/beranda** dan **/home** merupakan sebuah duplikat walaupun beda bahasa, mari kita ubah!

```
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
  [ ... ]
  &nbsp;
  <li class="nav-item">
    <a class="btn {{ Route::is('dashboard.index') ? 'btn-success' : 'btn-
    outline-success' }}" href="{{ route('dashboard.index') }}"><i class="fas fa-tachometer-
    alt"></i> Dasbor</a>
  </li>
  [ ... ]
</nav>
```

Frontend Scaffolding Bootstrap + Authentications

Ubah berkas blade
navbar.blade.php di dalam
folder
resources/views/layouts/auth
pada bagian seperti yang di
samping.



File Edit Selection View Go Run Terminal Help DashboardController.php - pertemuan6 - Visual Studio Code

EXPLORER

- app
- Console
- Exceptions
- Http
 - Controllers
 - Auth
 - Controller.php
 - DashboardController.php
- Middleware
- Kernel.php
- Providers
 - User.php
- bootstrap
- config
- database
- public
- resources
- routes
- storage
- tests
- vendor
- .editorconfig
- .env
- .env.example
- .gitattributes
- .gitignore
- .styleci.yml

OUTLINE

NPM SCRIPTS

Ln 29, Col 1 Spaces: 4 UTF-8 LF PHP

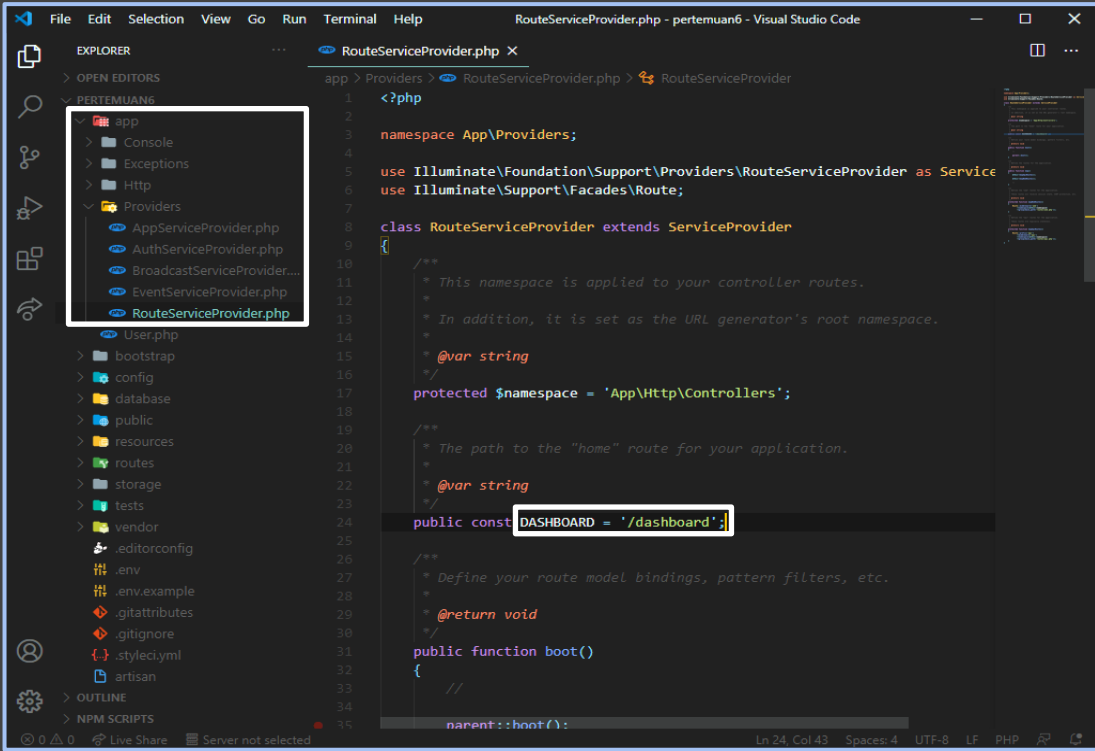
```
<?php
namespace App\Http\Controllers;

use Illuminate\Http\Request;

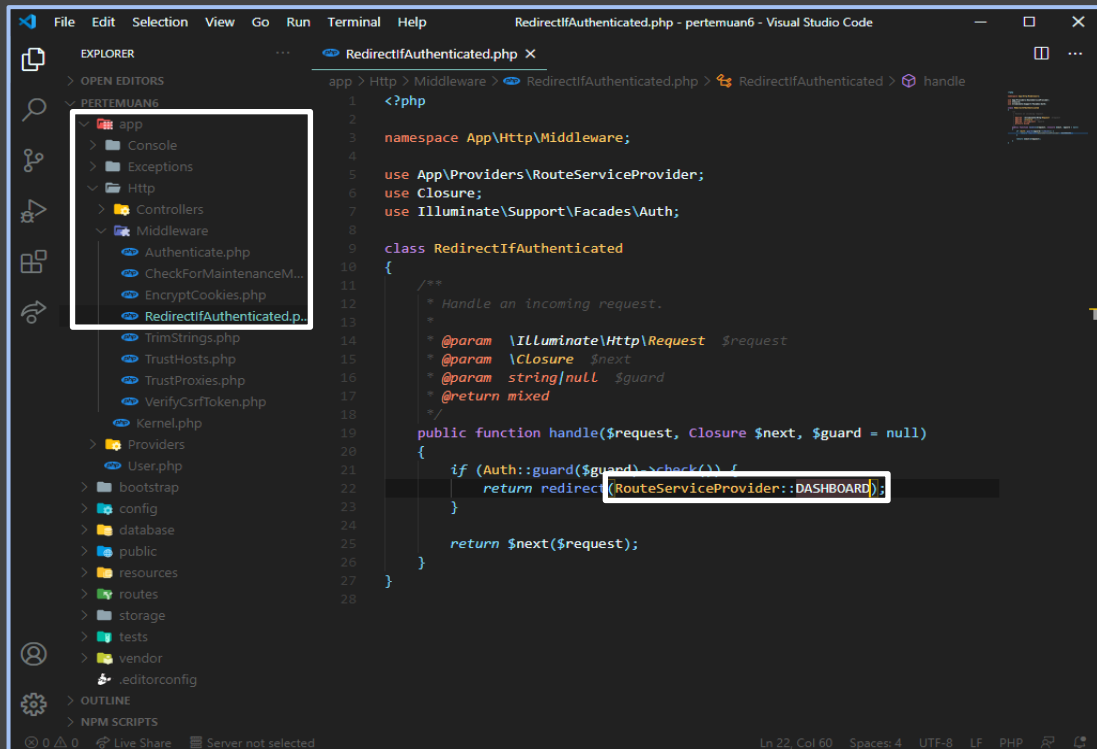
class DashboardController extends Controller
{
    /**
     * Create a new controller instance.
     *
     * @return void
     */
    public function __construct()
    {
        $this->middleware('auth');
    }

    /**
     * Show the application dashboard.
     *
     * @return \Illuminate\Contracts\Support\Renderable
     */
    public function index()
    {
        return view('home');
    }
}
```

Ubah Nama
HomeController menjadi
DashboardController



Buka berkas **RouteServiceProvider.php** yang ada di folder **app/Providers**.

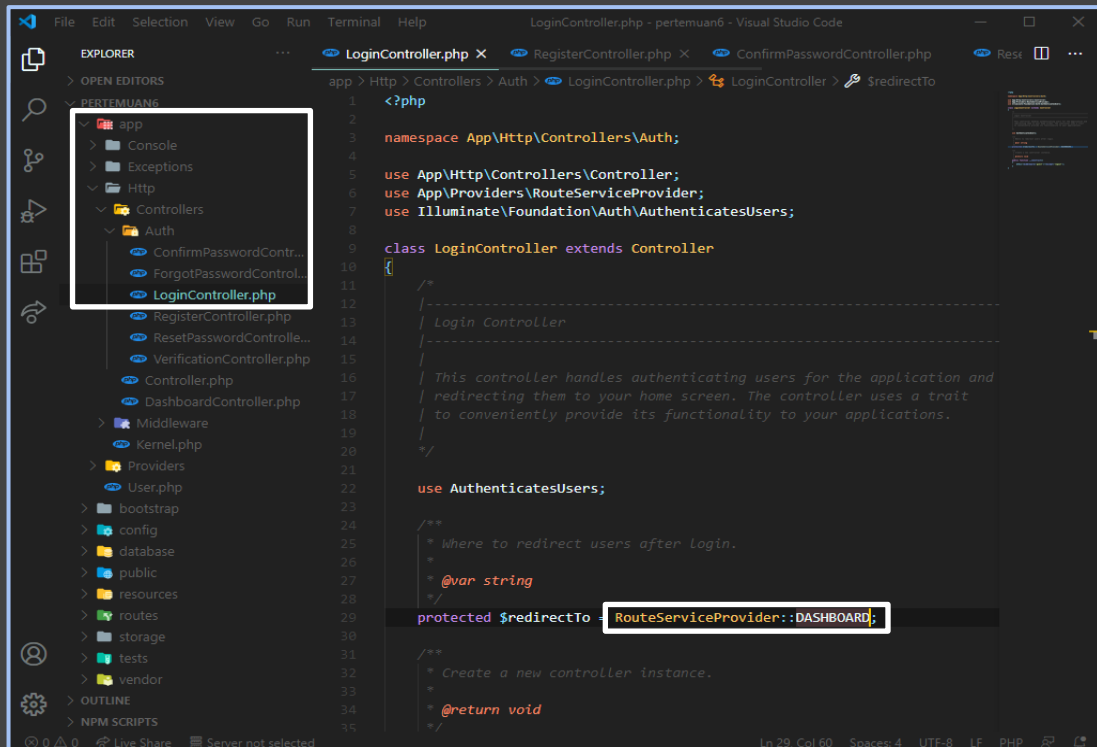


The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar is open, showing a project structure with folders like Console, Exceptions, Http, Controllers, and Middleware. The file `RedirectIfAuthenticated.php` is highlighted in the Middleware folder. The main editor displays the code for `RedirectIfAuthenticated.php`. The code is a PHP class that implements the `Illuminate\Http\Middleware\RedirectIfAuthenticated` interface. It has a `handle` method that checks if the user is authenticated and redirects to the dashboard if so. The `handle` method is defined as follows:

```
1 <?php
2
3 namespace App\Http\Middleware;
4
5 use App\Providers\RouteServiceProvider;
6 use Closure;
7 use Illuminate\Support\Facades\Auth;
8
9 class RedirectIfAuthenticated
10 {
11     /**
12      * Handle an incoming request.
13      *
14      * @param \Illuminate\Http\Request $request
15      * @param \Closure $next
16      * @param string|null $guard
17      * @return mixed
18      */
19     public function handle($request, Closure $next, $guard = null)
20     {
21         if (Auth::guard($guard)->check()) {
22             return redirect(RouteServiceProvider::DASHBOARD);
23         }
24
25         return $next($request);
26     }
27 }
28
```

Ubah Constanta **HOME** di **RedirectIfAuthenticated**

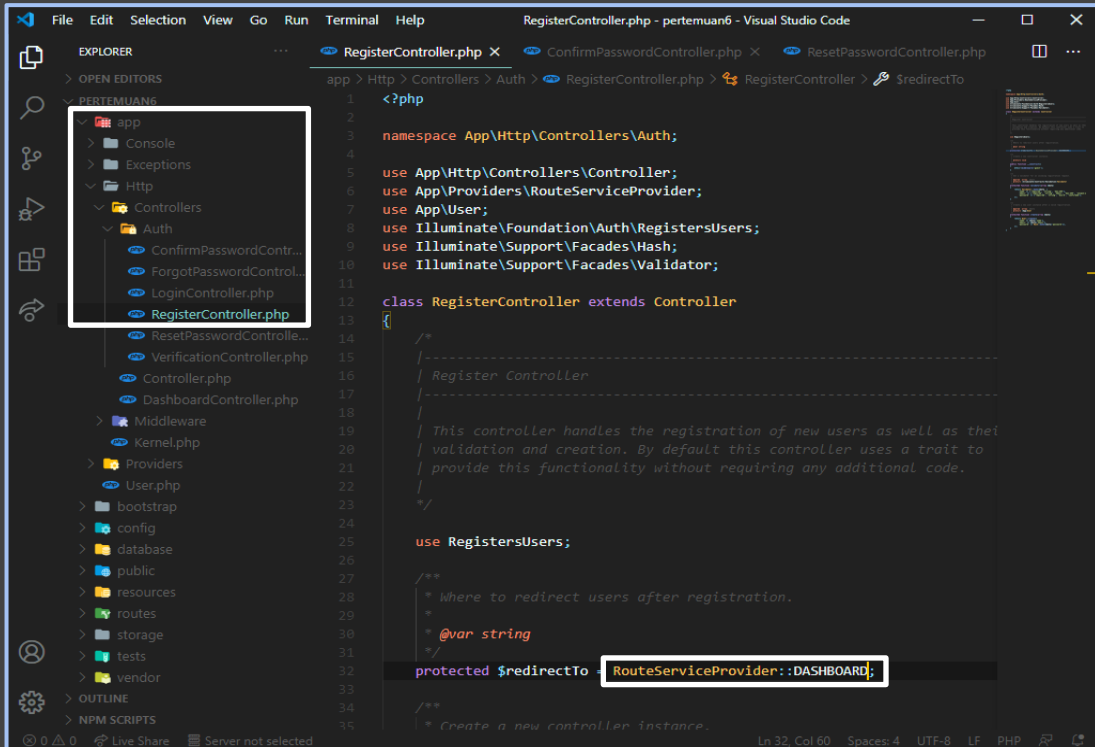
Buka berkas **RedirectIfAuthenticated.php** yang ada di folder **app/Http/Middleware**.



```
File Edit Selection View Go Run Terminal Help LoginController.php - pertemuan6 - Visual Studio Code
LoginController.php x RegisterController.php x ConfirmPasswordController.php Rest
app > Http > Controllers > Auth > LoginController.php > LoginController > $redirectTo
1 <?php
2
3 namespace App\Http\Controllers\Auth;
4
5 use App\Http\Controllers\Controller;
6 use App\Providers\RouteServiceProvider;
7 use Illuminate\Foundation\Auth\AuthenticatesUsers;
8
9 class LoginController extends Controller
10 {
11     /**
12      * Login Controller
13      *
14      * This controller handles authenticating users for the application and
15      * redirecting them to your home screen. The controller uses a trait
16      * to conveniently provide its functionality to your applications.
17      */
18
19     use AuthenticatesUsers;
20
21     /**
22      * Where to redirect users after login.
23      *
24      * @var string
25      */
26     protected $redirectTo = RouteServiceProvider::DASHBOARD;
27
28     /**
29      * Create a new controller instance.
30      *
31      * @return void
32      */
33 }
```

Ubah Constanta **HOME** di **LoginController**

Buka berkas **LoginController.php** yang ada di folder **app/Http/Auth**.



```
File Edit Selection View Go Run Terminal Help RegisterController.php - pertemuan6 - Visual Studio Code

EXPLORER
... RegisterController.php X ConfirmPasswordController.php X ResetPasswordController.php
> OPEN EDITORS
PERTEMUANG
  app
  > Console
  > Exceptions
  > Http
  > Controllers
    Auth
      ConfirmPasswordContro...
      ForgotPasswordControl...
      LoginController.php
      RegisterController.php
      ResetPasswordControle...
      VerificationController.php
      Controller.php
      DashboardController.php
  > Middleware
  > Kernel.php
  > Providers
  > User.php
  > bootstrap
  > config
  > database
  > public
  > resources
  > routes
  > storage
  > tests
  > vendor
  > OUTLINE
  > NPM SCRIPTS

app > Http > Controllers > Auth > RegisterController.php > $redirectTo
1  <?php
2
3  namespace App\Http\Controllers\Auth;
4
5  use App\Http\Controllers\Controller;
6  use App\Providers\RouteServiceProvider;
7  use App\User;
8  use Illuminate\Foundation\Auth\RegistersUsers;
9  use Illuminate\Support\Facades\Hash;
10 use Illuminate\Support\Facades\Validator;
11
12 class RegisterController extends Controller
13 {
14     /**
15      *
16      * Register Controller
17      *
18      * This controller handles the registration of new users as well as thei
19      * validation and creation. By default this controller uses a trait to
20      * provide this functionality without requiring any additional code.
21      */
22     /**
23      *
24      * use RegistersUsers;
25
26     /**
27      * Where to redirect users after registration.
28      *
29      * @var string
30      */
31     protected $redirectTo = RouteServiceProvider::DASHBOARD;
32
33     /**
34      * Create a new controller instance.
35
```

Ubah Constanta HOME di RegisterController

Buka berkas **RegisterController.php** yang ada di folder **app/Http/Auth**.

```
File Edit Selection View Go Run Terminal Help ConfirmPasswordController.php - pertemuan6 - Visual Studio Code
ConfirmPasswordController.php X ResetPasswordController.php VerificationController.php
app > Http > Controllers > Auth > ConfirmPasswordController.php > ConfirmPasswordController > $redirectTo
1 <?php
2
3 namespace App\Http\Controllers\Auth;
4
5 use App\Http\Controllers\Controller;
6 use App\Providers\RouteServiceProvider;
7 use Illuminate\Foundation\Auth\ConfirmsPasswords;
8
9 class ConfirmPasswordController extends Controller
10 {
11     /**
12      * Confirm Password Controller
13      *
14      * This controller is responsible for handling password confirmations and
15      * uses a simple trait to include the behavior. You're free to explore
16      * this trait and override any functions that require customization.
17      */
18
19     /**
20      * Where to redirect users when the intended url fails.
21      *
22      * @var string
23      */
24     protected $redirectTo = RouteServiceProvider::DASHBOARD;
25
26     /**
27      * Create a new controller instance.
28      *
29      * @return void
30      */
31 }
```

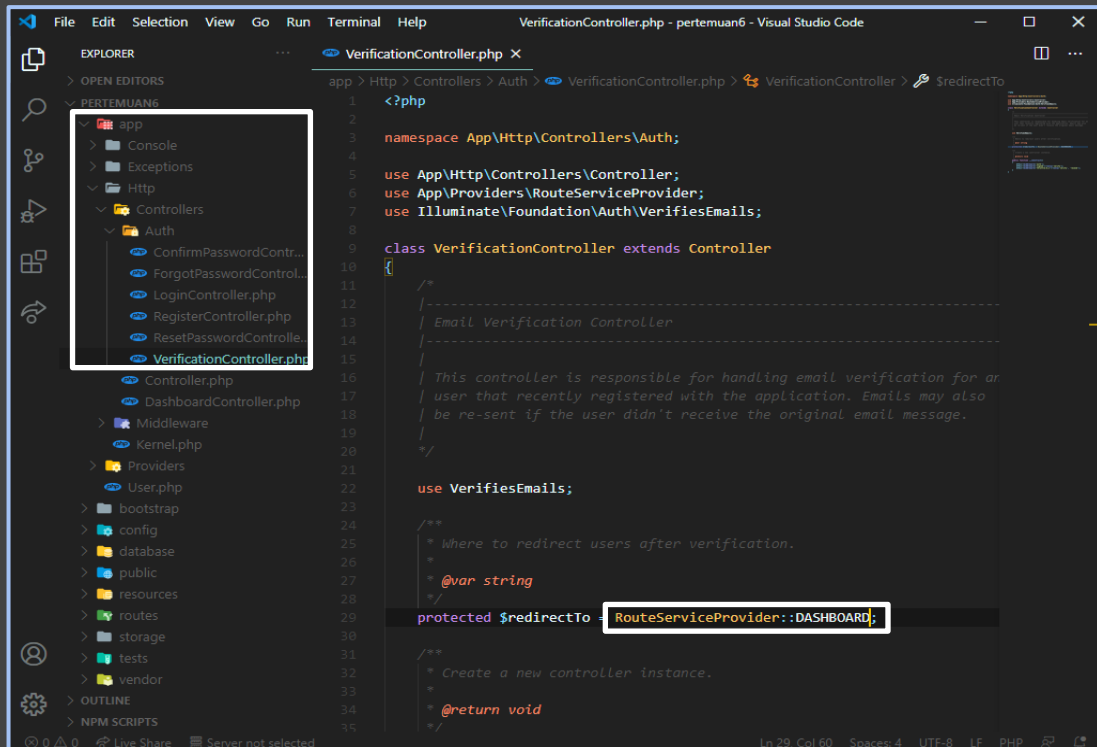
Ubah Constanta **HOME** di **ConfirmPasswordController**

Buka berkas **ConfirmPasswordController.php** yang ada di folder **app/Http/Auth**.

```
File Edit Selection View Go Run Terminal Help ResetPasswordController.php - pertemuan6 - Visual Studio Code
ResetPasswordController.php X VerificationController.php
app > Http > Controllers > Auth > ResetPasswordController.php > ResetPasswordController > $redirectTo
1 <?php
2
3 namespace App\Http\Controllers\Auth;
4
5 use App\Http\Controllers\Controller;
6 use App\Providers\RouteServiceProvider;
7 use Illuminate\Foundation\Auth\ResetsPasswords;
8
9 class ResetPasswordController extends Controller
10 {
11     /**
12      * Password Reset Controller
13      *
14      * This controller is responsible for handling password reset requests
15      * and uses a simple trait to include this behavior. You're free to
16      * explore this trait and override any methods you wish to tweak.
17      */
18
19     use ResetsPasswords;
20
21     /**
22      * Where to redirect users after resetting their password.
23      *
24      * @var string
25      */
26     protected $redirectTo = RouteServiceProvider::DASHBOARD;
27
28 }
```

Ubah Constanta **HOME** di **ResetPasswordController**

Buka berkas **ResetPasswordController.php** yang ada di folder **app/Http/Auth**.



```
File Edit Selection View Go Run Terminal Help VerificationController.php - pertemuan6 - Visual Studio Code

EXPLORER
  app
  Console
  Exceptions
  Http
  Controllers
    Auth
      ConfirmationController.php
      ForgotPasswordController.php
      LoginController.php
      RegisterController.php
      ResetPasswordController.php
      VerificationController.php
    Controller.php
  DashboardController.php
  Middleware
  Kernel.php
  Providers
    User.php
  bootstrap
  config
  database
  public
  resources
  routes
  storage
  tests
  vendor
  OUTLINE
  NPM SCRIPTS

VerificationController.php
  app > Http > Controllers > Auth > VerificationController.php > VerificationController > $redirectTo
  1 <?php
  2
  3 namespace App\Http\Controllers\Auth;
  4
  5 use App\Http\Controllers\Controller;
  6 use App\Providers\RouteServiceProvider;
  7 use Illuminate\Foundation\Auth\VerifiesEmails;
  8
  9 class VerificationController extends Controller
 10 {
 11     /**
 12      * Email Verification Controller
 13      *
 14      * This controller is responsible for handling email verification for an
 15      * user that recently registered with the application. Emails may also
 16      * be re-sent if the user didn't receive the original email message.
 17      */
 18
 19     use VerifiesEmails;
 20
 21     /**
 22      * Where to redirect users after verification.
 23      *
 24      * @var string
 25      */
 26     protected $redirectTo = RouteServiceProvider::DASHBOARD;
 27
 28     /**
 29      * Create a new controller instance.
 30      *
 31      * @return void
 32      */
 33 }
```

Ubah Constanta **HOME** di **VerificationController**

Buka berkas **VerificationController.php** yang ada di folder **app/Http/Auth**.

```
<?php

use Illuminate\Support\Facades\Route;

/*
|-----
| Web Routes
|-----
|
| Here is where you can register web routes for your application. These
| routes are loaded by the RouteServiceProvider within a group which
| contains the "web" middleware group. Now create something great!
|
*/

Auth::routes(
    [
        'register' => false,
    ]
);

Route::get('/', function () {
    return view('contents.beranda');
})->name('beranda');

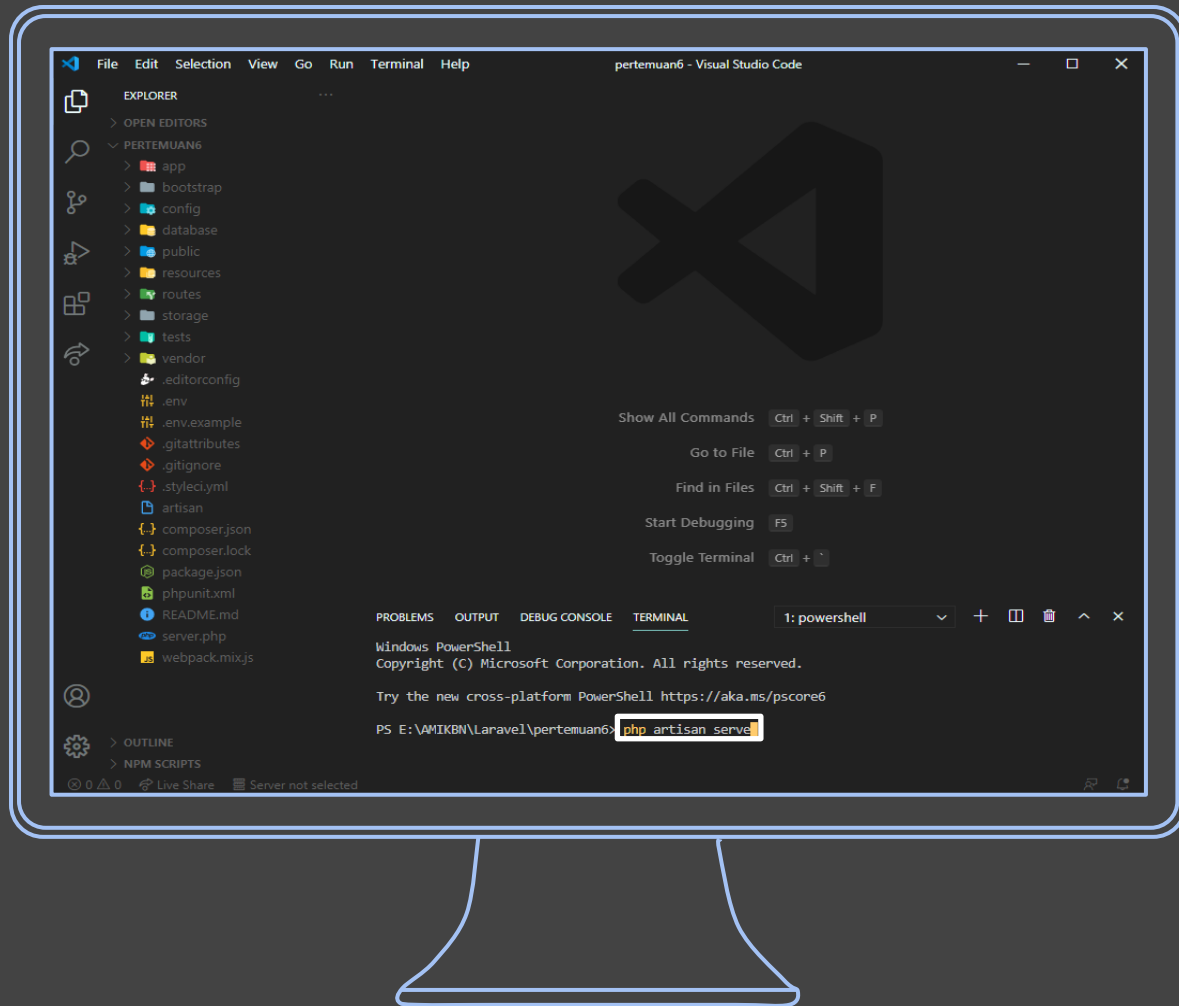
Route::get('/hobi', function () {
    return view('contents.hobi');
})->name('hobi');

Route::get('/tentang', function () {
    return view('contents.tentang');
})->name('tentang');

Route::get('/dashboard', 'DashboardController@index')->name('dashboard.index');
```

Ubah Web Routing

Buka **web.php** yang ada di folder **routes** dan ubah isinya seperti di samping.



Mari Tes Hasil Prakteknya

Jalankan perintah **php artisan serve**



Sampai Ketemu
Kembali di
Pertemuan
Berikutnya!