

QUIZ
PEMROGRAMAN BERBASIS WEB LANJUTAN



ADAM MALIK SIREGAR
NIM 0702203202

DOSEN PENGAMPU
SUENDRI, M.Kom

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS SAINS DAN TEKNOLOGI
UIN SUMATERA UTARA MEDAN
2023**

QUIZ

(DESAIN DATABASE RELATIONAL, APLIKASI CMV, HTML + BOOTSTRAP)

I. PEMBUATAN DATABASE

Langkah langkah untuk membuat database seperti di soal yaitu :

1. Query untuk membuat table tb_golongan

```
1 CREATE TABLE `tb_golongan` (  
2   `gol_id` tinyint(3) UNSIGNED NOT NULL,  
3   `gol_kode` varchar(10) COLLATE utf8mb4_unicode_ci NOT NULL,  
4   `gol_nama` varchar(50) COLLATE utf8mb4_unicode_ci NOT NULL,  
5   `created_at` timestamp NULL DEFAULT NULL,  
6   `updated_at` timestamp NULL DEFAULT NULL,  
7   PRIMARY KEY (`gol_id`)  
8 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

2. Query untuk membuat table tb_users

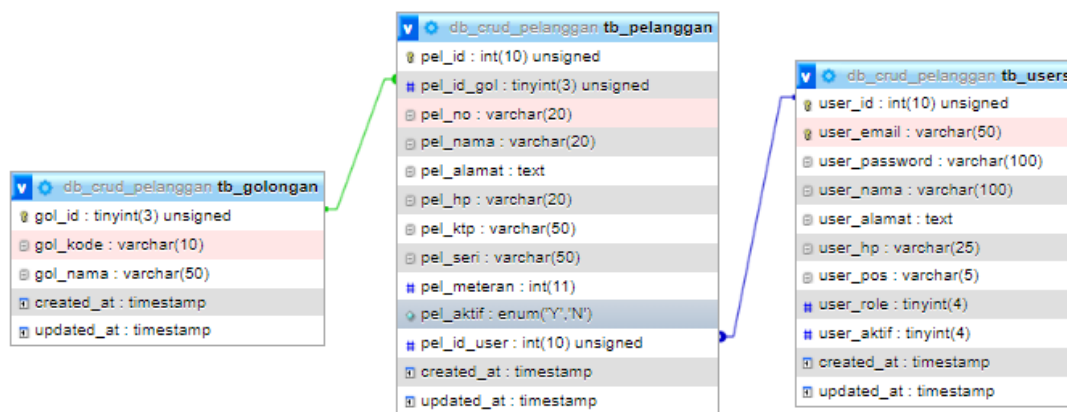
```
1 CREATE TABLE `tb_users` (  
2   `user_id` int(10) UNSIGNED NOT NULL,  
3   `user_email` varchar(50) COLLATE utf8mb4_unicode_ci NOT NULL,  
4   `user_password` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL,  
5   `user_nama` varchar(100) COLLATE utf8mb4_unicode_ci NOT NULL,  
6   `user_alamat` text COLLATE utf8mb4_unicode_ci NOT NULL,  
7   `user_hp` varchar(25) COLLATE utf8mb4_unicode_ci NOT NULL,  
8   `user_pos` varchar(5) COLLATE utf8mb4_unicode_ci NOT NULL,  
9   `user_role` tinyint(4) NOT NULL,  
10  `user_aktif` tinyint(4) NOT NULL,  
11  `created_at` timestamp NULL DEFAULT NULL,  
12  `updated_at` timestamp NULL DEFAULT NULL,  
13  PRIMARY KEY (`user_id`)|  
14 ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unicode_ci;
```

3. Query untuk membuat table tb_pelanggan

```
1 CREATE TABLE `tb_pelanggan` (  
2   `pel_id` int(10) UNSIGNED NOT NULL,  
3   `pel_id_gol` tinyint(3) UNSIGNED NOT NULL,  
4   `pel_no` varchar(20) COLLATE utf8mb4_unicode_ci NOT NULL,  
5   `pel_nama` varchar(20) COLLATE utf8mb4_unicode_ci NOT NULL,  
6   `pel_alamat` text COLLATE utf8mb4_unicode_ci NOT NULL,  
7   `pel_hp` varchar(20) COLLATE utf8mb4_unicode_ci NOT NULL,  
8   `pel_ktp` varchar(50) COLLATE utf8mb4_unicode_ci NOT NULL,  
9   `pel_seri` varchar(50) COLLATE utf8mb4_unicode_ci NOT NULL,  
10  `pel_meteran` int(11) NOT NULL,  
11  `pel_aktif` enum('Y','N') COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT 'Y',  
12  `pel_id_user` int(10) UNSIGNED NOT NULL,  
13  `created_at` timestamp NULL DEFAULT NULL,  
14  `updated_at` timestamp NULL DEFAULT NULL,  
15  PRIMARY KEY (`pel_id`),  
16  FOREIGN KEY (`pel_id_gol`) REFERENCES `tb_golongan` (`gol_id`),
```

HASIL AKHIR DATABASE

Setelah semua query diatas dilakukan. Maka hasil database yang dihasilkan yaitu :



II. PEMBUATAN APLIKASI DENGAN KONSEP MVC SERTA TAMPILAN HTML + BOOTSTRAP

MVC (Model-View-Controller) adalah sebuah pola desain arsitektur perangkat lunak yang digunakan untuk memisahkan logika aplikasi, presentasi tampilan, dan data. Berikut adalah penjelasan singkat tentang masing-masing komponen dalam pola desain MVC:

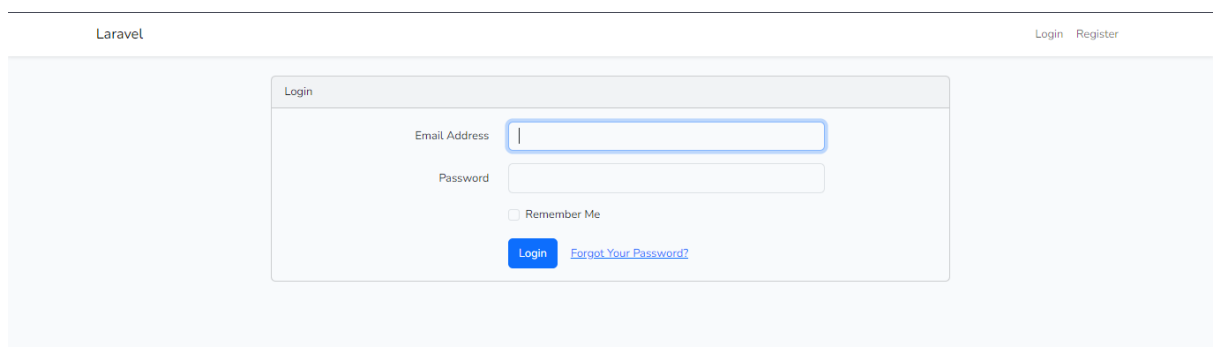
1. Model: Model mewakili logika bisnis dan data aplikasi. Model berisi metode dan fungsi yang digunakan untuk memanipulasi dan mengakses data. Model dapat berinteraksi dengan database, melakukan validasi data, dan mengambil/menyimpan data ke penyimpanan persisten.

2. View: View adalah tampilan atau antarmuka pengguna. View bertanggung jawab untuk menampilkan informasi kepada pengguna dan menerima input dari pengguna. View dapat berupa halaman HTML, template, atau elemen antarmuka pengguna lainnya. View tidak memiliki logika bisnis, tetapi hanya bertugas untuk menampilkan data yang diberikan oleh Model.
3. Controller: Controller bertindak sebagai penghubung antara Model dan View. Controller menerima input dari pengguna melalui View, memproses input tersebut, dan berinteraksi dengan Model untuk memperoleh data yang diperlukan. Setelah itu, Controller mengirimkan data tersebut ke View yang tepat untuk ditampilkan kepada pengguna. Controller juga dapat mengubah Model atau mengarahkan pengguna ke halaman lain berdasarkan interaksi pengguna.

Pembuatan aplikasi ini menggunakan Laravel 10 yang telah mendukung proses MVC didalamnya. Pada aplikasi ini terdapat beberapa modul yang terbentuk, yaitu Modul Login, Modul Pelanggan, Modul Users, Modul Golongan. Berikut penjelasan lebih lengkapnya :

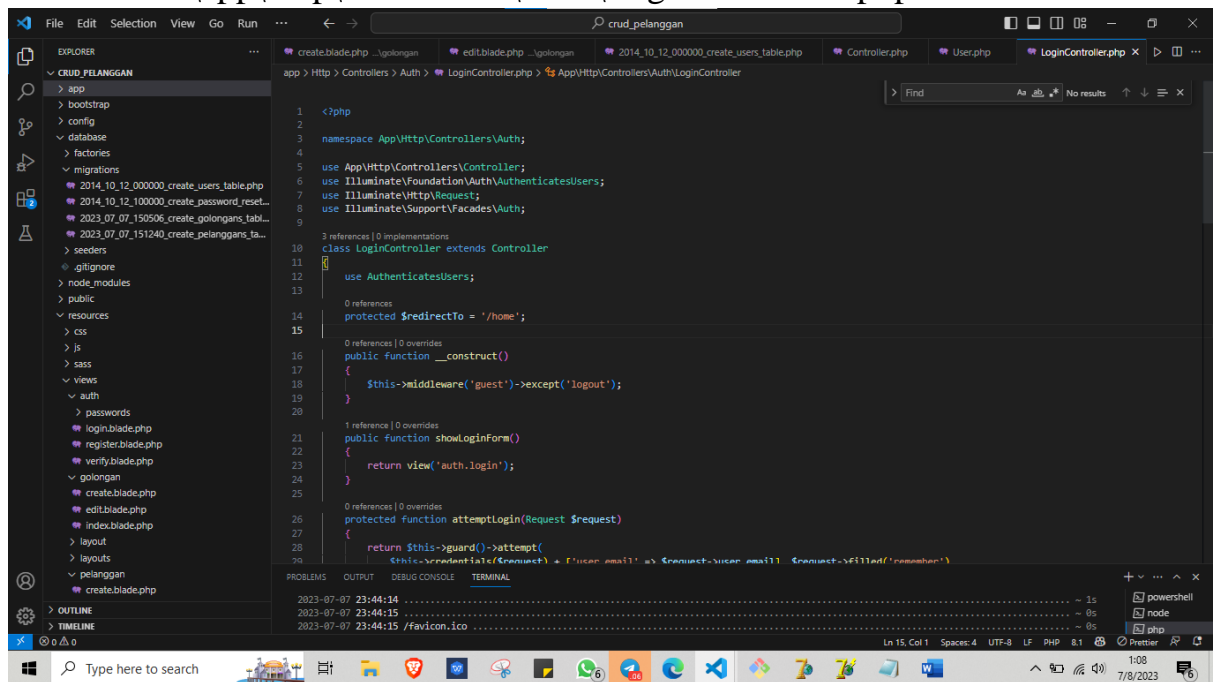
1. Modul Login

Modul login digunakan untuk melakukan validasi terhadap user akan menggunakan aplikasi ini. Tujuannya yaitu untuk menjaga aplikasi dari user yang tidak bertanggung jawab.



The image shows a screenshot of a web application's login page. At the top left, the word "Laravel" is displayed. At the top right, there are links for "Login" and "Register". The main content area features a "Login" form with a title bar. Inside the form, there are two input fields: "Email Address" and "Password". Below the "Password" field is a checkbox labeled "Remember Me". At the bottom of the form, there is a blue "Login" button and a blue link labeled "Forgot Your Password?".

Controller : \app\Http\Controllers\Auth\LoginController.php



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

use App\Http\Controllers\Controller;
use Illuminate\Foundation\Auth\AuthenticatesUsers;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;

class LoginController extends Controller
{
    use AuthenticatesUsers;

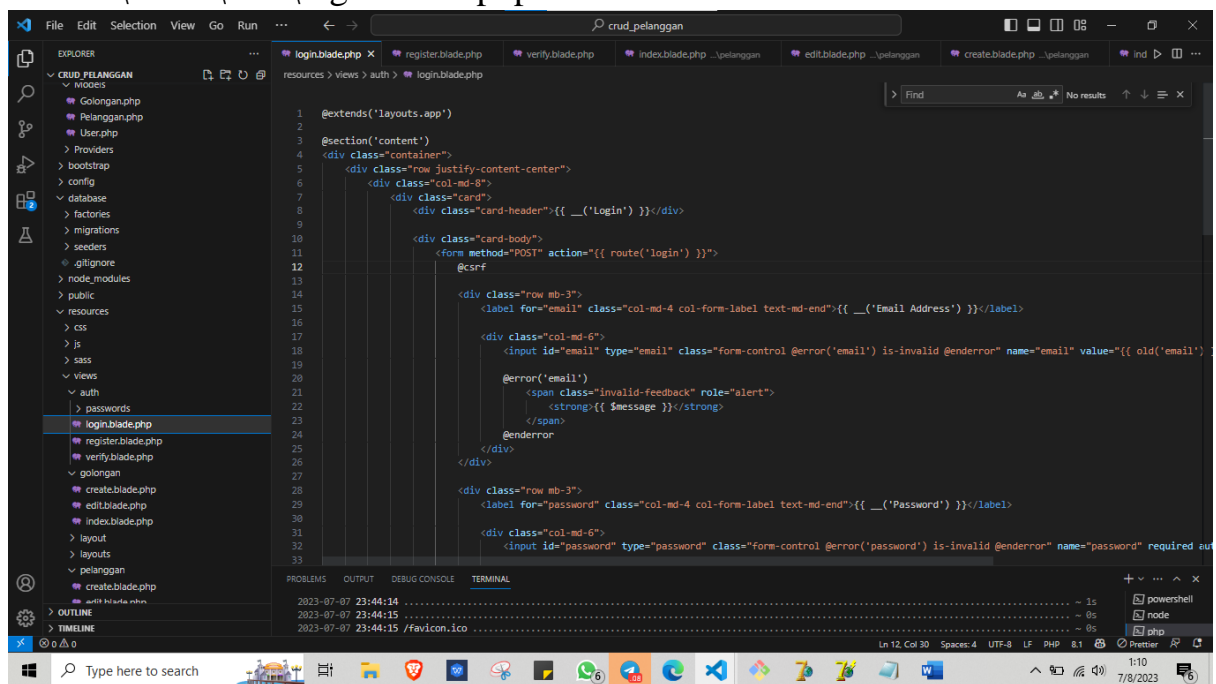
    protected $redirectTo = '/home';

    public function __construct()
    {
        $this->middleware('guest')->except('logout');
    }

    public function showLoginForm()
    {
        return view('auth.login');
    }

    protected function attemptLogin(Request $request)
    {
        return $this->guard()->attempt(
            $this->credentials($request) + ['remember' => $request->filled('remember')]
        );
    }
}
```

View : \views\auth\login.blade.php



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

@section('content')


{{ __('Login') }}



<form method="POST" action="{{ route('login') }}">
@csrf



{{ __('Email Address') }}



{{ $message }}

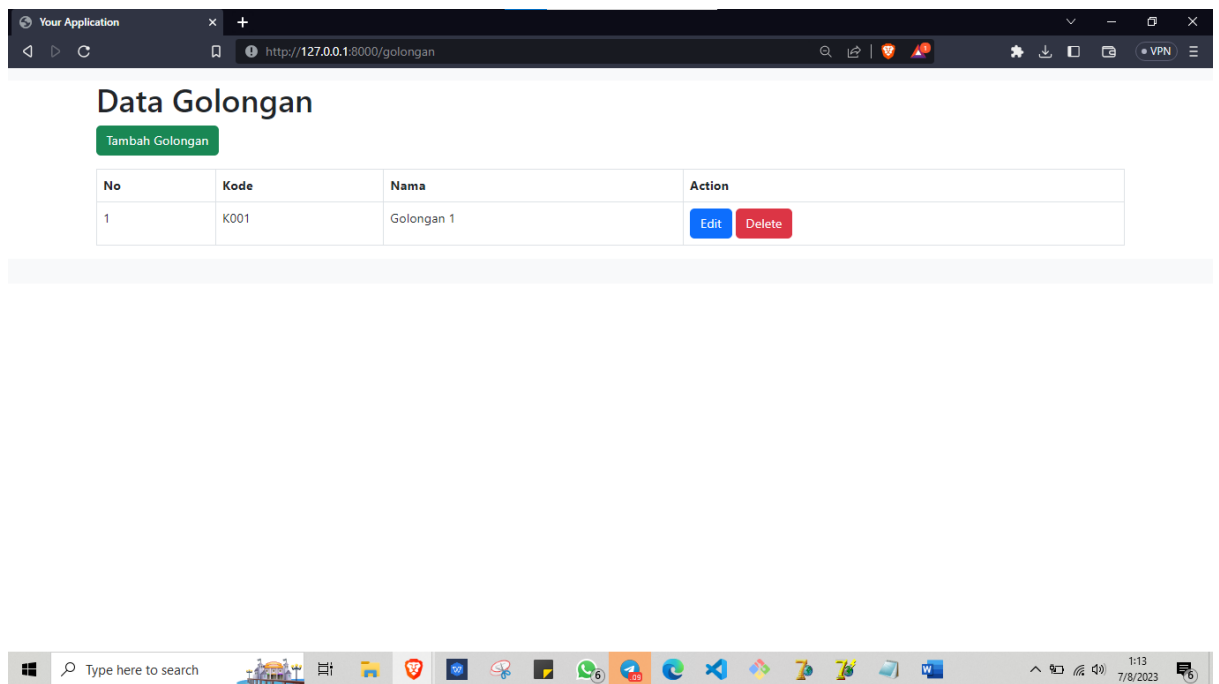


{{ __('Password') }}

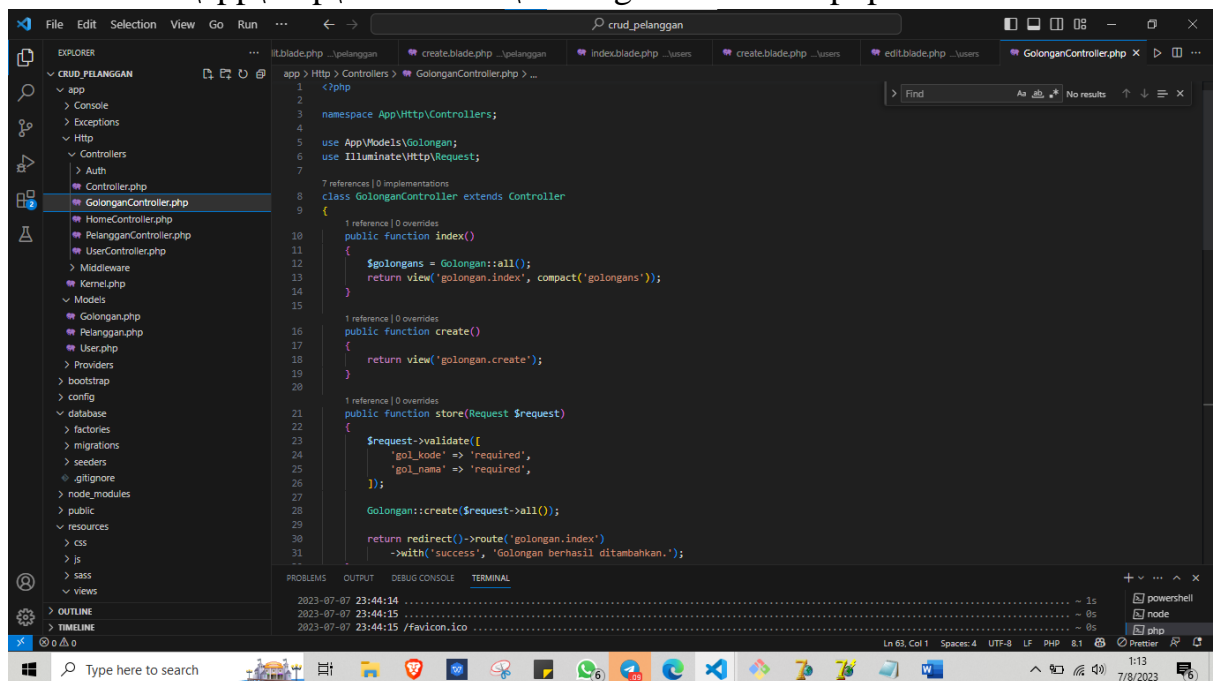

```

2. Modul Golongan

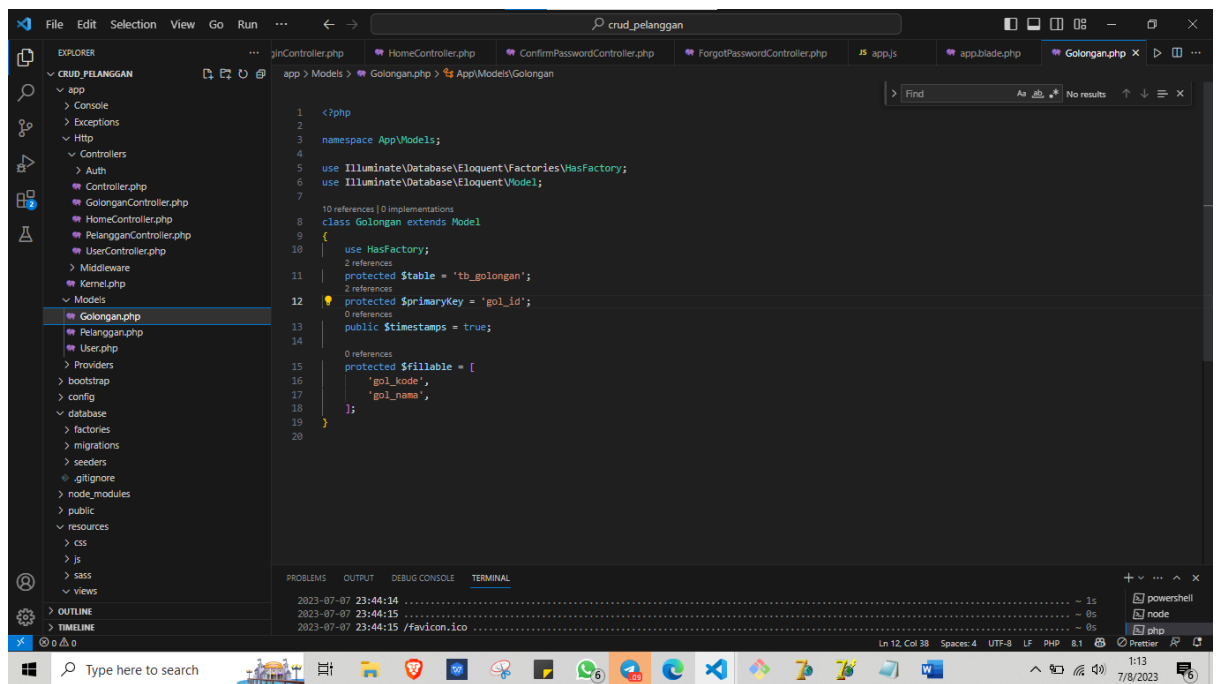
Modul golongan digunakan untuk mengatur data golongan. Pada modul ini dapat melakukan Create, Edit, dan Delete.



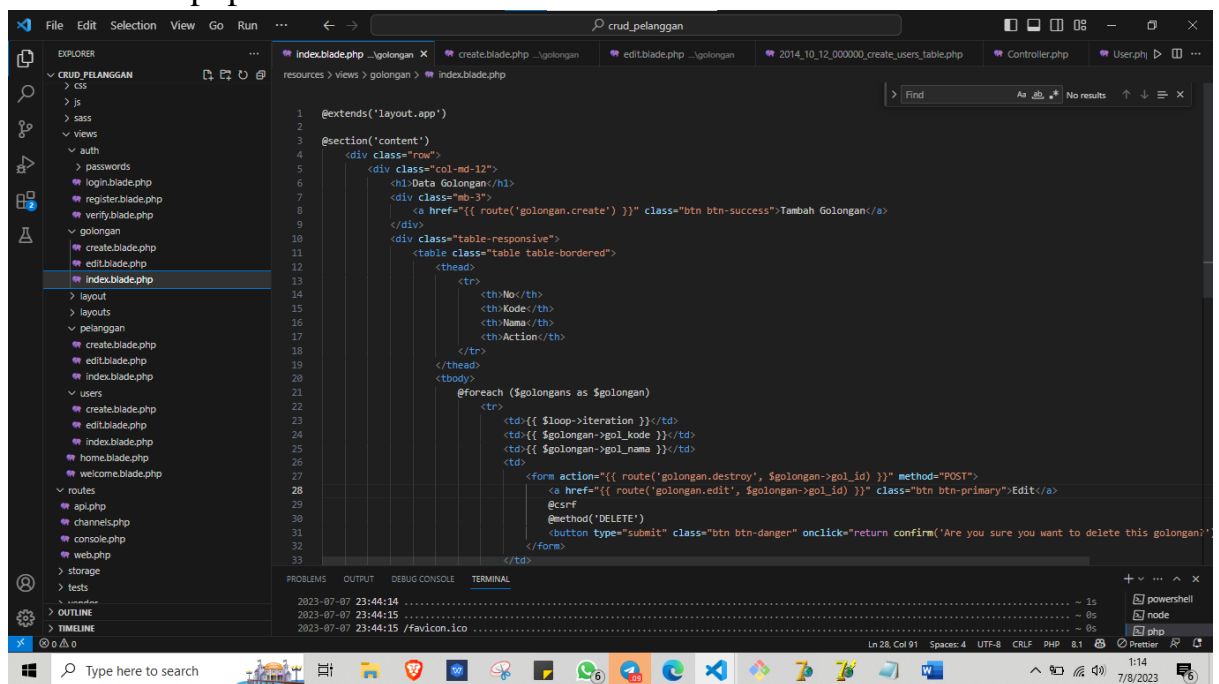
Controller : \app\Http\Controllers\GolonganController.php

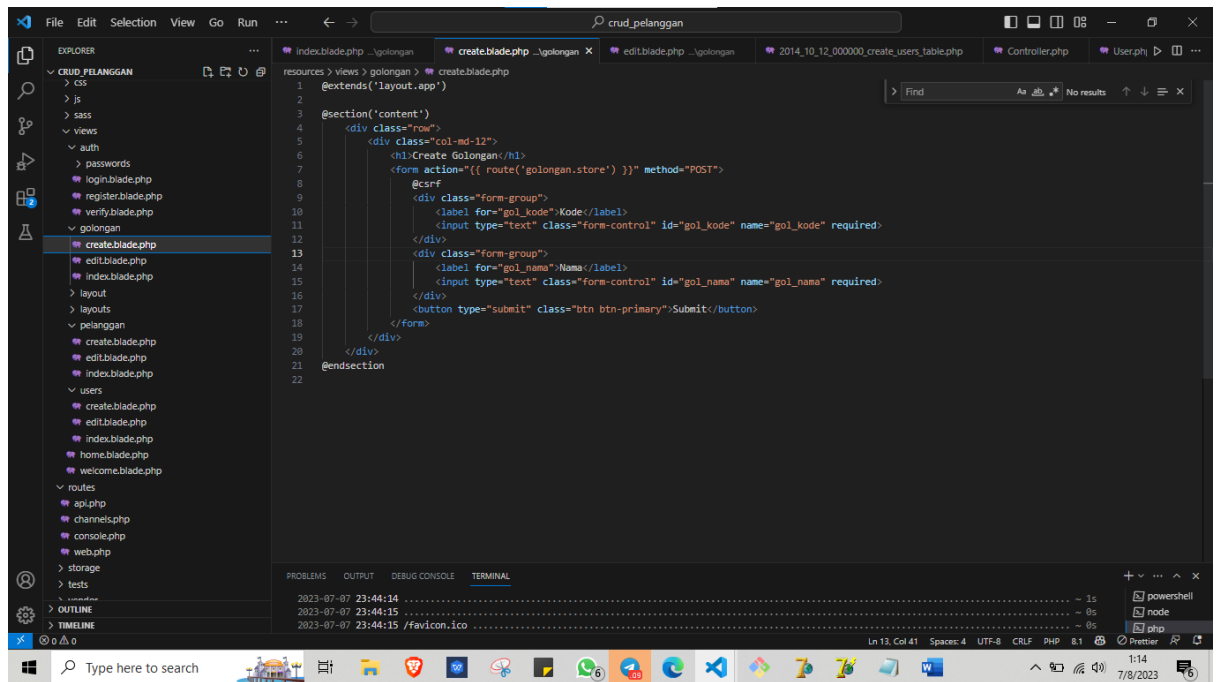


Model : \app\Models\Golongan.php



View : \views\golongan
index.blade.php

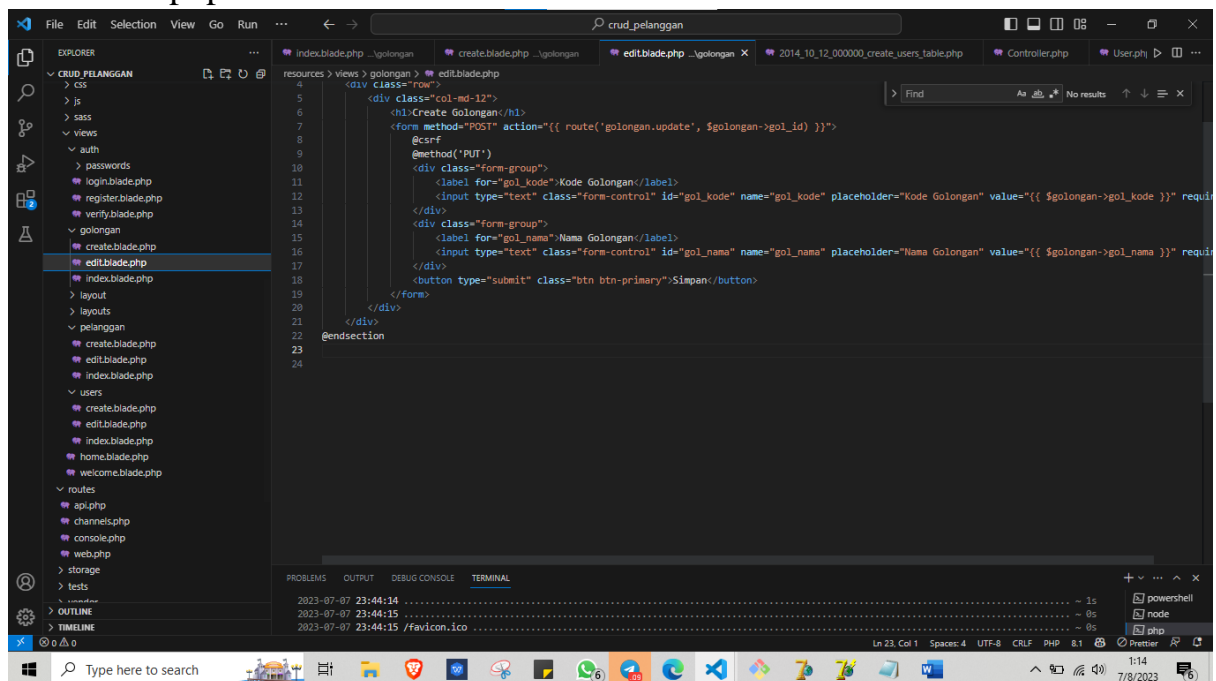




The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying the project structure. The main editor window shows the `create.blade.php` file in the `resources > views > golongan` directory. The code is a Blade template that extends the `layout.app` and contains a form for creating a new 'golongan' (category). The form has two input fields: `gol_kode` (required) and `gol_nama` (required), followed by a submit button labeled 'Submit'.

```
1 @extends('layout.app')
2
3 @section('content')
4     <div class="row">
5         <div class="col-md-12">
6             <h1>Create Golongan</h1>
7             <form action="{{ route('golongan.store') }}" method="POST">
8                 @csrf
9                 <div class="form-group">
10                     <label for="gol_kode">Kode</label>
11                     <input type="text" class="form-control" id="gol_kode" name="gol_kode" required>
12                 </div>
13                 <div class="form-group">
14                     <label for="gol_nama">Nama</label>
15                     <input type="text" class="form-control" id="gol_nama" name="gol_nama" required>
16                 </div>
17                 <button type="submit" class="btn btn-primary">Submit</button>
18             </form>
19         </div>
20     </div>
21 @endsection
```

Edit.blade.php

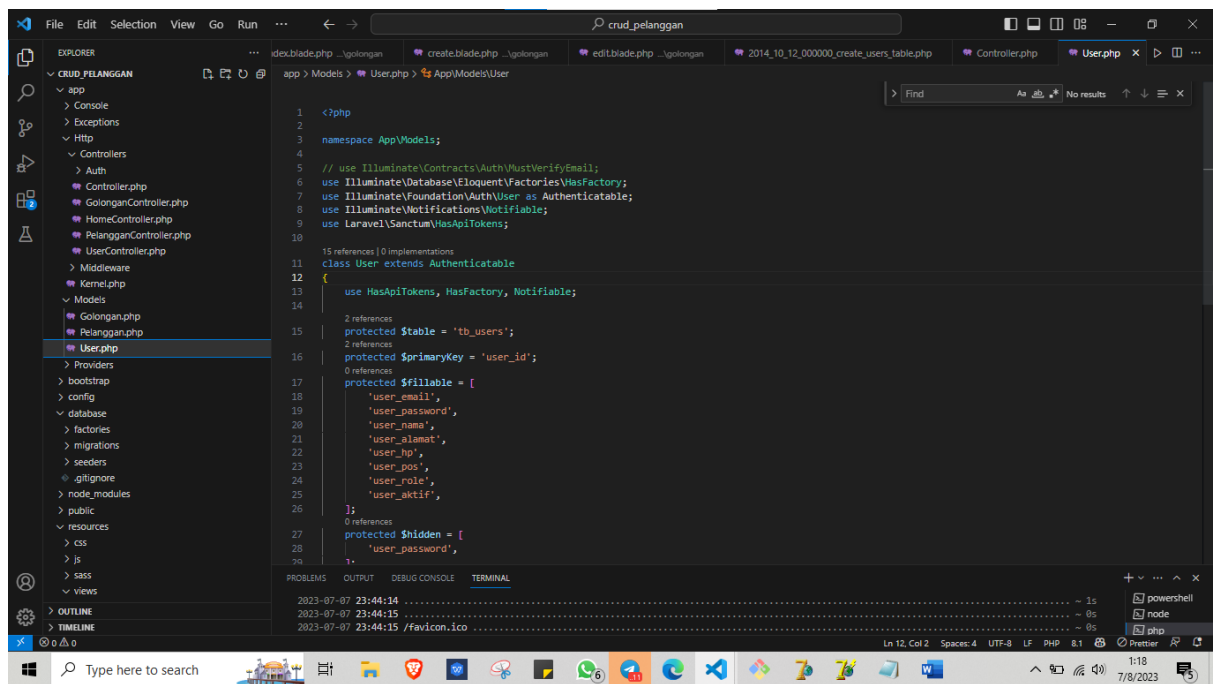


The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying the project structure. The main editor window shows the `edit.blade.php` file in the `resources > views > golongan` directory. The code is a Blade template that extends the `layout.app` and contains a form for editing an existing 'golongan' (category). The form has two input fields: `gol_kode` (required, with a placeholder and value from the model) and `gol_nama` (required, with a placeholder and value from the model), followed by a submit button labeled 'Simpan'.

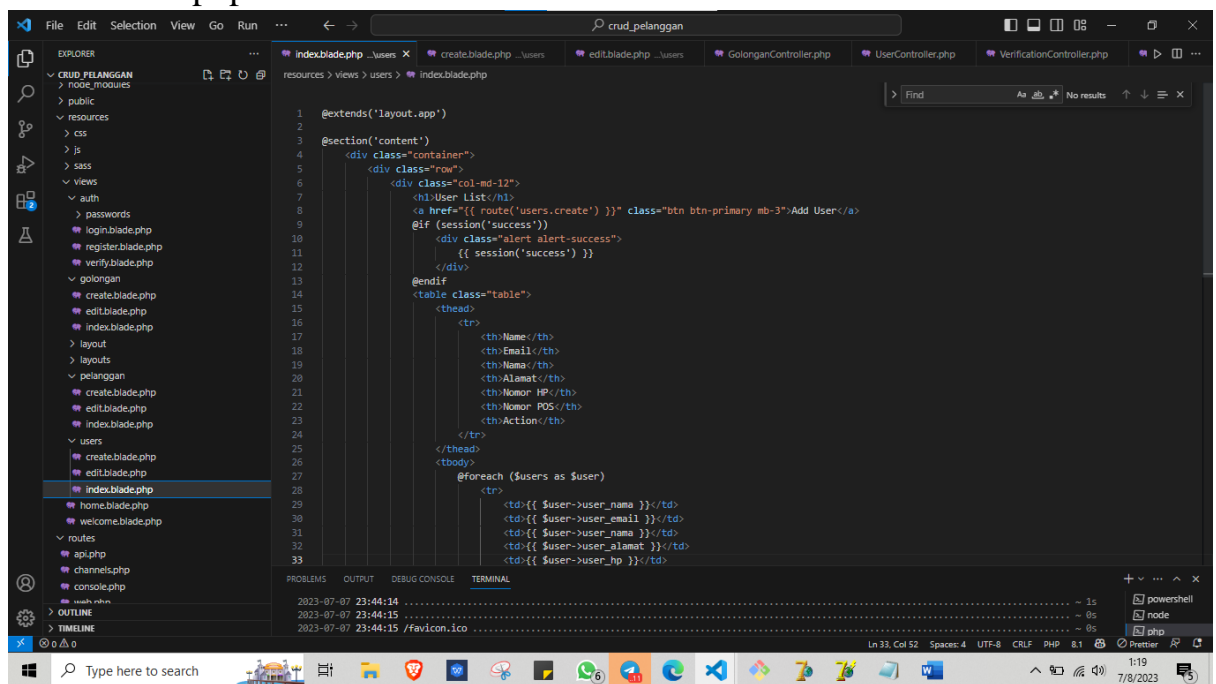
```
1 @extends('layout.app')
2
3 @section('content')
4     <div class="row">
5         <div class="col-md-12">
6             <h1>Create Golongan</h1>
7             <form method="POST" action="{{ route('golongan.update', $golongan->gol_id) }}">
8                 @csrf
9                 @method('PUT')
10                 <div class="form-group">
11                     <label for="gol_kode">Kode Golongan</label>
12                     <input type="text" class="form-control" id="gol_kode" name="gol_kode" placeholder="Kode Golongan" value="{{ $golongan->gol_kode }}" required>
13                 </div>
14                 <div class="form-group">
15                     <label for="gol_nama">Nama Golongan</label>
16                     <input type="text" class="form-control" id="gol_nama" name="gol_nama" placeholder="Nama Golongan" value="{{ $golongan->gol_nama }}" required>
17                 </div>
18                 <button type="submit" class="btn btn-primary">Simpan</button>
19             </form>
20         </div>
21     </div>
22 @endsection
```

3. Modul Users

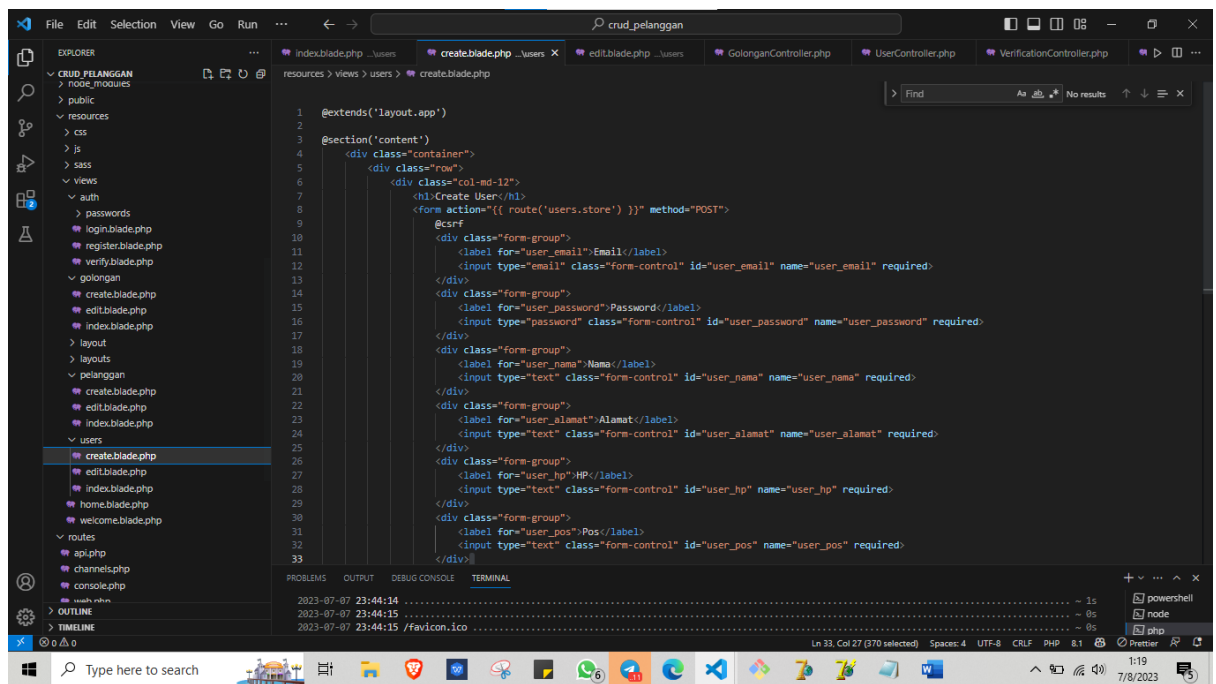
Modul users digunakan untuk mengatur data users pada aplikasi. Modul users dapat melakukan create, edit, dan delete data users.



Views : \views\users
Index.blade.php

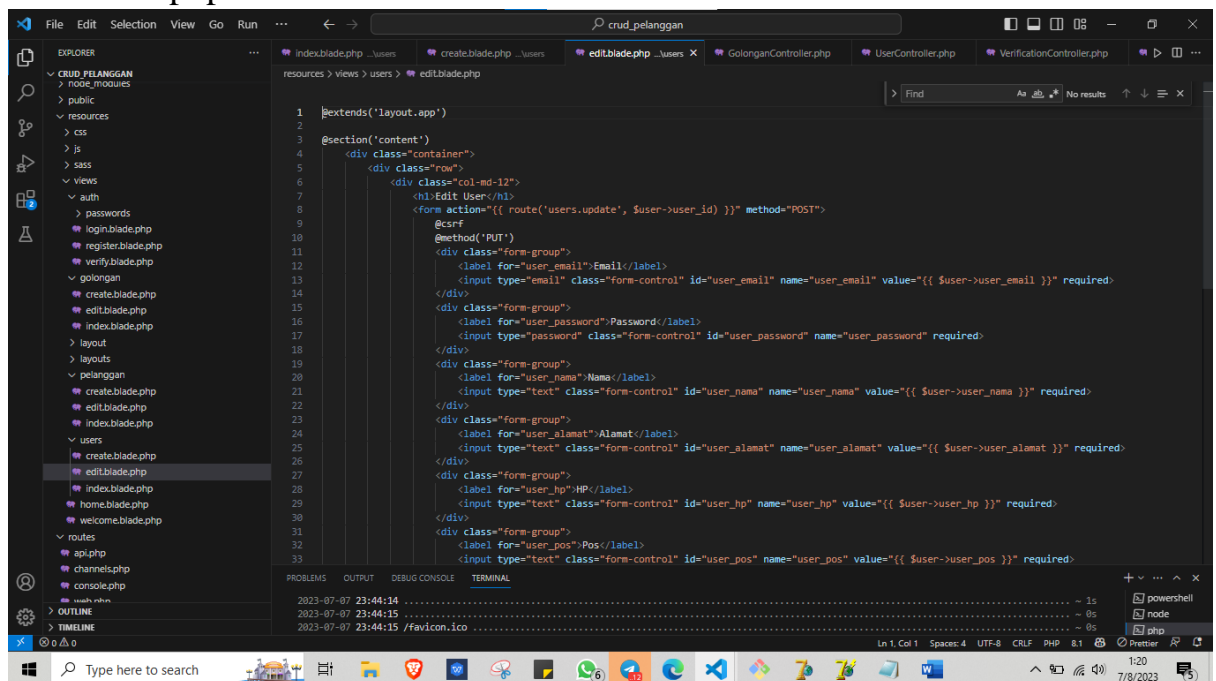


create.blade.php



```
1 @extends('layout.app')
2
3 @section('content')
4     <div class="container">
5         <div class="row">
6             <div class="col-md-12">
7                 <h1>Create User</h1>
8                 <form action="{{ route('users.store') }}" method="POST">
9                     @csrf
10                     <div class="form-group">
11                         <label for="user_email">Email</label>
12                         <input type="email" class="form-control" id="user_email" name="user_email" required>
13                     </div>
14                     <div class="form-group">
15                         <label for="user_password">Password</label>
16                         <input type="password" class="form-control" id="user_password" name="user_password" required>
17                     </div>
18                     <div class="form-group">
19                         <label for="user_nama">Name</label>
20                         <input type="text" class="form-control" id="user_nama" name="user_nama" required>
21                     </div>
22                     <div class="form-group">
23                         <label for="user_alamat">Alamat</label>
24                         <input type="text" class="form-control" id="user_alamat" name="user_alamat" required>
25                     </div>
26                     <div class="form-group">
27                         <label for="user_hp">HP</label>
28                         <input type="text" class="form-control" id="user_hp" name="user_hp" required>
29                     </div>
30                     <div class="form-group">
31                         <label for="user_pos">Pos</label>
32                         <input type="text" class="form-control" id="user_pos" name="user_pos" required>
33                     </div>
34                 </form>
35             </div>
36         </div>
37     </div>
38 @endsection
```

Edit.blade.php

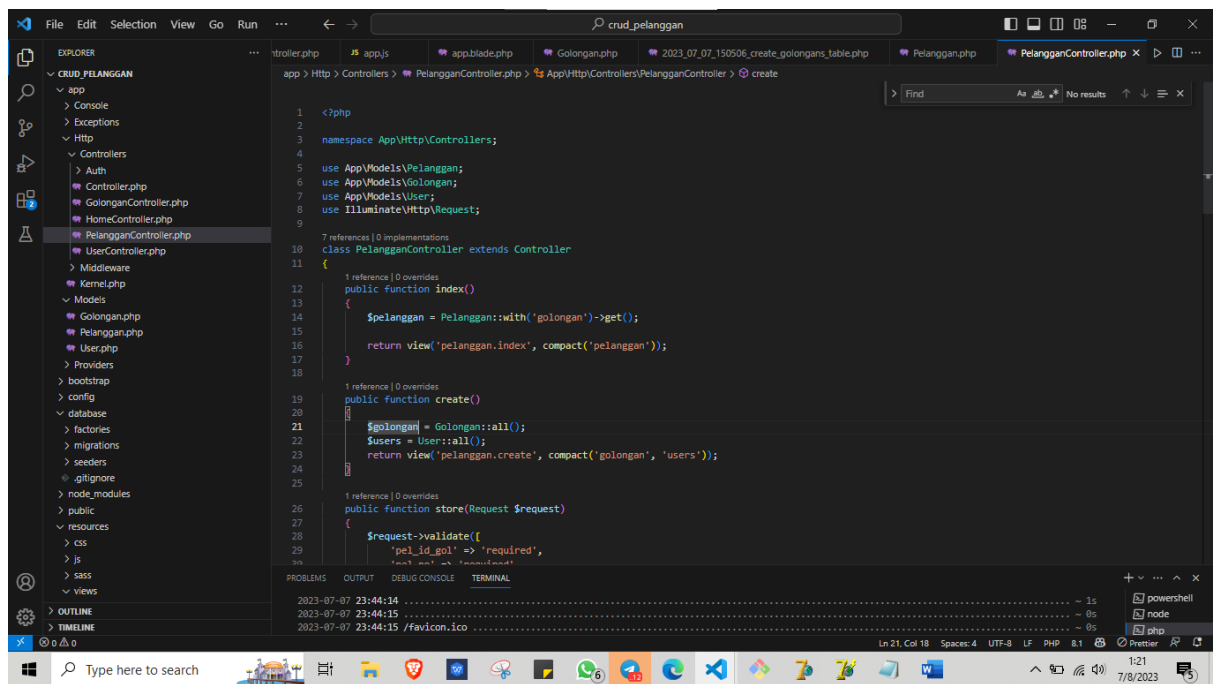


```
1 @extends('layout.app')
2
3 @section('content')
4     <div class="container">
5         <div class="row">
6             <div class="col-md-12">
7                 <h1>Edit User</h1>
8                 <form action="{{ route('users.update', $user->user_id) }}" method="POST">
9                     @csrf
10                     @method('PUT')
11                     <div class="form-group">
12                         <label for="user_email">Email</label>
13                         <input type="email" class="form-control" id="user_email" name="user_email" value="{{ $user->user_email }}" required>
14                     </div>
15                     <div class="form-group">
16                         <label for="user_password">Password</label>
17                         <input type="password" class="form-control" id="user_password" name="user_password" required>
18                     </div>
19                     <div class="form-group">
20                         <label for="user_nama">Name</label>
21                         <input type="text" class="form-control" id="user_nama" name="user_nama" value="{{ $user->user_nama }}" required>
22                     </div>
23                     <div class="form-group">
24                         <label for="user_alamat">Alamat</label>
25                         <input type="text" class="form-control" id="user_alamat" name="user_alamat" value="{{ $user->user_alamat }}" required>
26                     </div>
27                     <div class="form-group">
28                         <label for="user_hp">HP</label>
29                         <input type="text" class="form-control" id="user_hp" name="user_hp" value="{{ $user->user_hp }}" required>
30                     </div>
31                     <div class="form-group">
32                         <label for="user_pos">Pos</label>
33                         <input type="text" class="form-control" id="user_pos" name="user_pos" value="{{ $user->user_pos }}" required>
34                     </div>
35                 </form>
36             </div>
37         </div>
38     </div>
39 @endsection
```

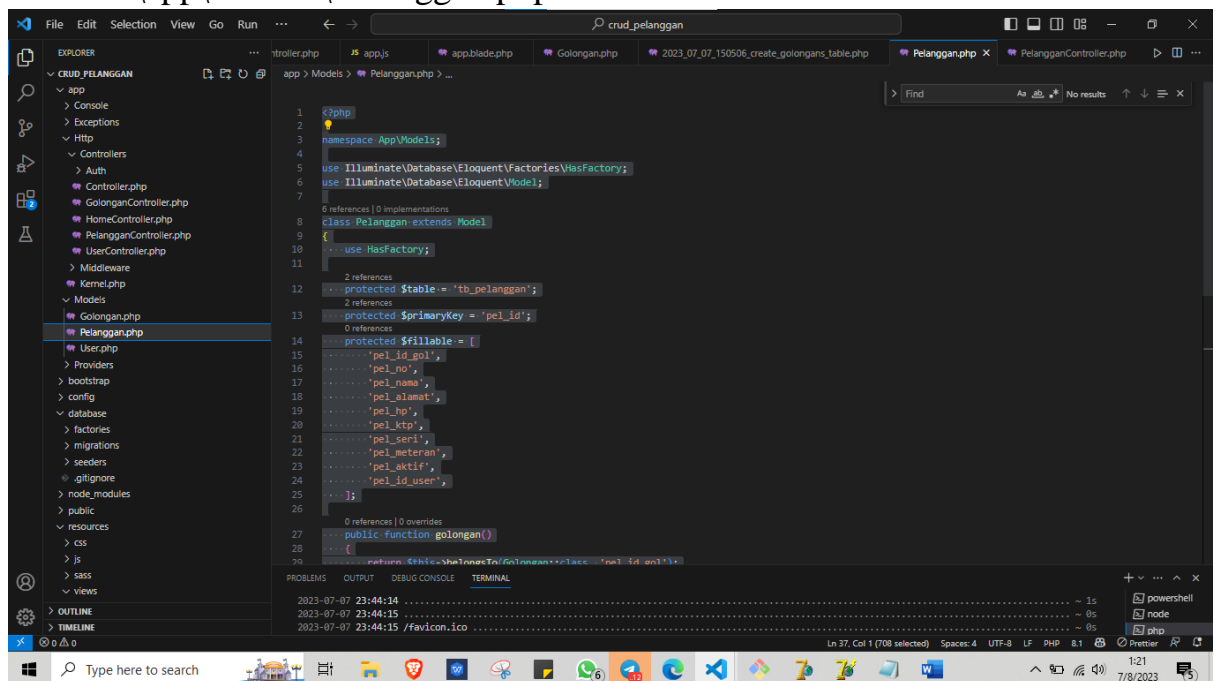
4. Modul Pelanggan

Modul pelanggan digunakan untuk mendaftarkan pelanggan di aplikasi ini. Data pelanggan berelasi dengan data golongan dan data users. Jadi pada data pelanggan akan terinput golongan beserta users yang menangannya.

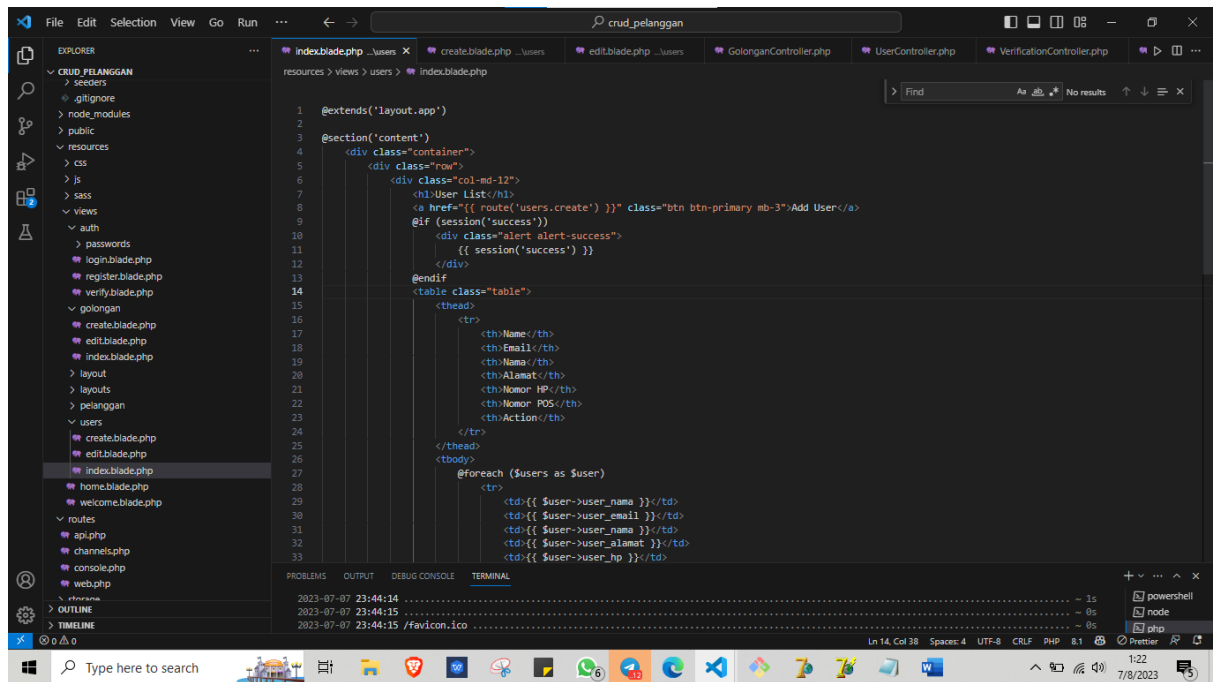
Controller : \app\Http\Controllers\PelangganController.php



Model : \app\Models\Pelanggan.php

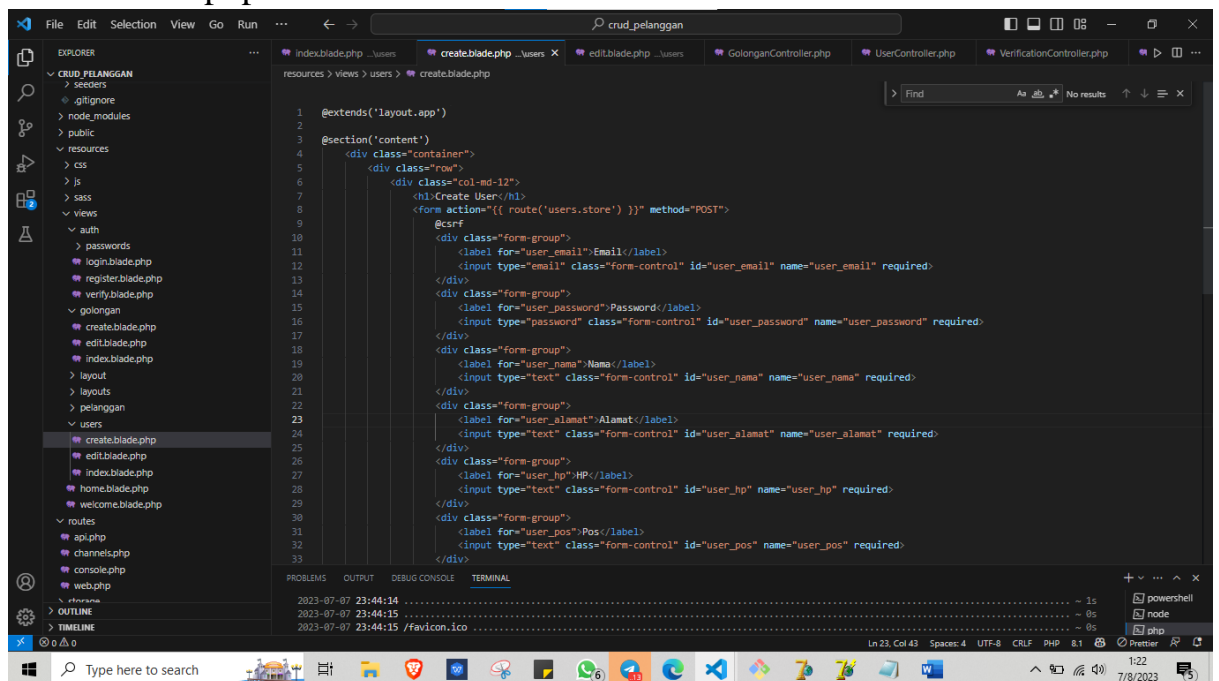


Views : \views\pelanggan
Index.blade.php



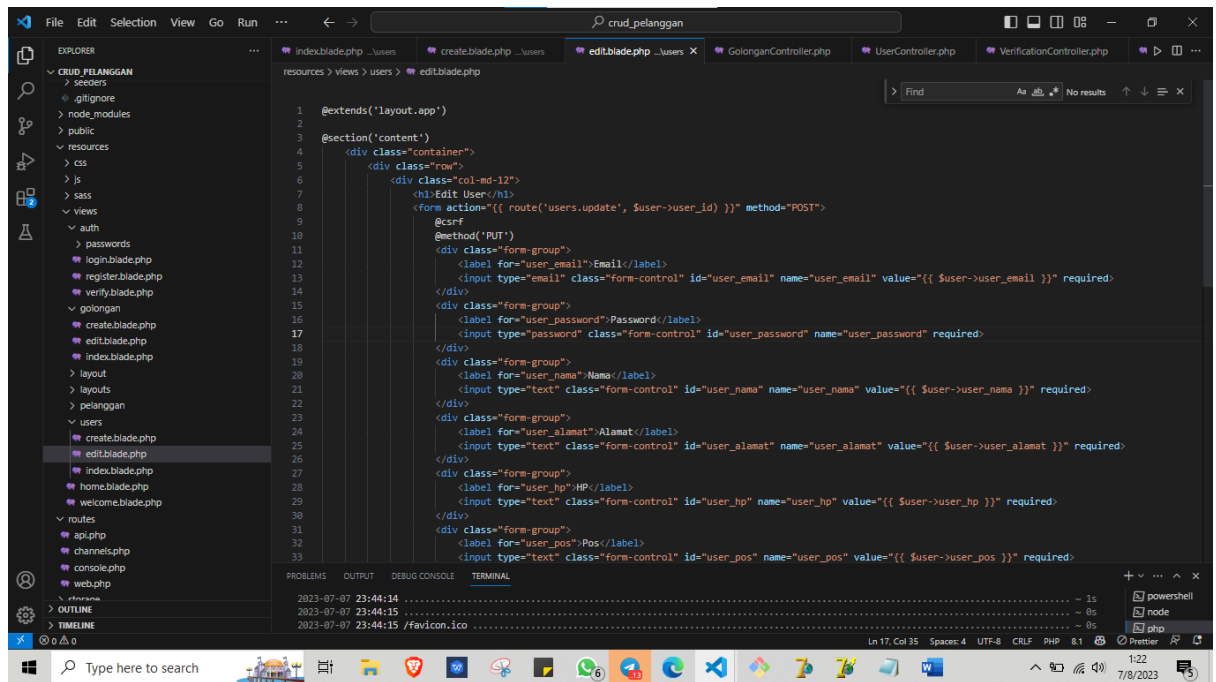
```
1 @extends('layout.app')
2
3 @section('content')
4     <div class="container">
5         <div class="row">
6             <div class="col-md-12">
7                 <h1>User List</h1>
8                 <a href="{{ route('users.create') }}" class="btn btn-primary mb-3">Add User</a>
9                 @if (session('success'))
10                     <div class="alert alert-success">
11                         {{ session('success') }}
12                     </div>
13                 @endif
14                 <table class="table">
15                     <thead>
16                         <tr>
17                             <th>Name</th>
18                             <th>Email</th>
19                             <th>Nama</th>
20                             <th>Alamat</th>
21                             <th>Nomor HP</th>
22                             <th>Nomor POS</th>
23                             <th>Action</th>
24                         </tr>
25                     </thead>
26                     <tbody>
27                         @foreach ($users as $user)
28                             <tr>
29                                 <td>{{ $user->user_nama }}</td>
30                                 <td>{{ $user->user_email }}</td>
31                                 <td>{{ $user->user_nama }}</td>
32                                 <td>{{ $user->user_alamat }}</td>
33                                 <td>{{ $user->user_hp }}</td>
```

Create.blade.php



```
1 @extends('layout.app')
2
3 @section('content')
4     <div class="container">
5         <div class="row">
6             <div class="col-md-12">
7                 <h1>Create User</h1>
8                 <form action="{{ route('users.store') }}" method="POST">
9                     @csrf
10                     <div class="form-group">
11                         <label for="user_email">Email</label>
12                         <input type="email" class="form-control" id="user_email" name="user_email" required>
13                     </div>
14                     <div class="form-group">
15                         <label for="user_password">Password</label>
16                         <input type="password" class="form-control" id="user_password" name="user_password" required>
17                     </div>
18                     <div class="form-group">
19                         <label for="user_nama">Nama</label>
20                         <input type="text" class="form-control" id="user_nama" name="user_nama" required>
21                     </div>
22                     <div class="form-group">
23                         <label for="user_alamat">Alamat</label>
24                         <input type="text" class="form-control" id="user_alamat" name="user_alamat" required>
25                     </div>
26                     <div class="form-group">
27                         <label for="user_hp">HP</label>
28                         <input type="text" class="form-control" id="user_hp" name="user_hp" required>
29                     </div>
30                     <div class="form-group">
31                         <label for="user_pos">Pos</label>
32                         <input type="text" class="form-control" id="user_pos" name="user_pos" required>
33                     </div>
```

Edit.blade.php



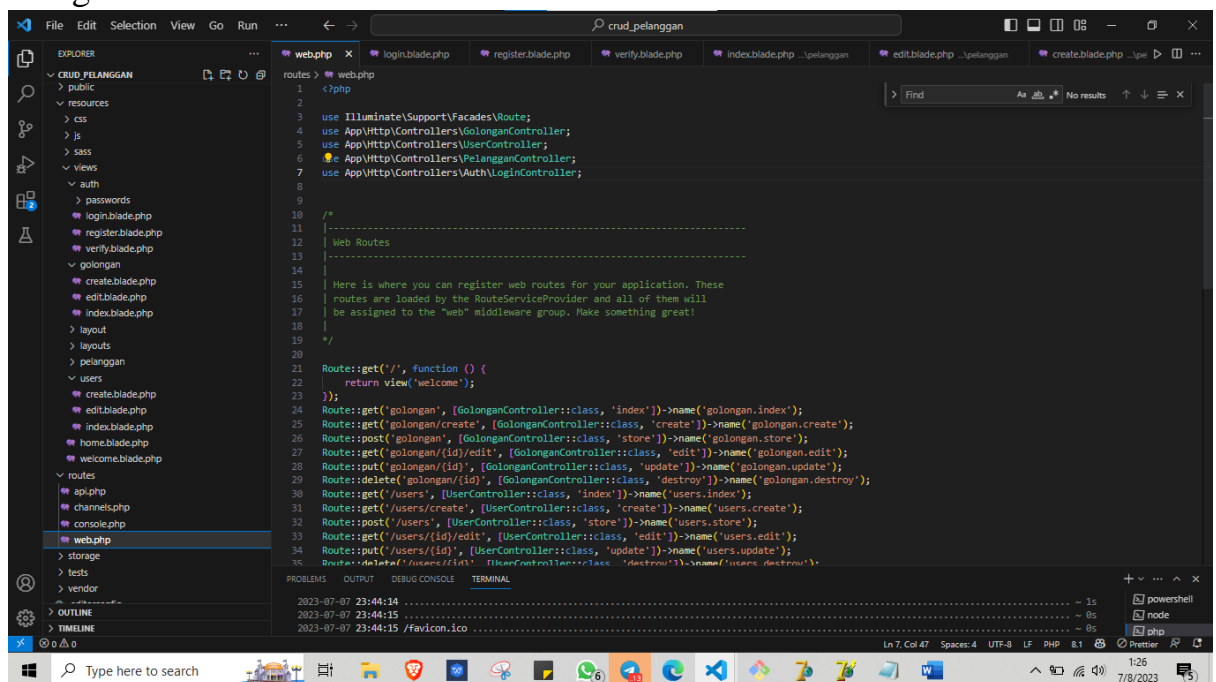
```
1 @extends('layout.app')
2
3 @section('content')
4     <div class="container">
5         <div class="row">
6             <div class="col-md-12">
7                 <h1>Edit User</h1>
8                 <form action="{{ route('users.update', $user->user_id) }}" method="POST">
9                     @csrf
10                     @method('PUT')
11                     <div class="form-group">
12                         <label for="user_email">Email</label>
13                         <input type="email" class="form-control" id="user_email" name="user_email" value="{{ $user->user_email }}" required>
14                     </div>
15                     <div class="form-group">
16                         <label for="user_password">Password</label>
17                         <input type="password" class="form-control" id="user_password" name="user_password" required>
18                     </div>
19                     <div class="form-group">
20                         <label for="user_nama">Nama</label>
21                         <input type="text" class="form-control" id="user_nama" name="user_nama" value="{{ $user->user_nama }}" required>
22                     </div>
23                     <div class="form-group">
24                         <label for="user_alamat">Alamat</label>
25                         <input type="text" class="form-control" id="user_alamat" name="user_alamat" value="{{ $user->user_alamat }}" required>
26                     </div>
27                     <div class="form-group">
28                         <label for="user_hp">HP</label>
29                         <input type="text" class="form-control" id="user_hp" name="user_hp" value="{{ $user->user_hp }}" required>
30                     </div>
31                     <div class="form-group">
32                         <label for="user_pos">Pos</label>
33                         <input type="text" class="form-control" id="user_pos" name="user_pos" value="{{ $user->user_pos }}" required>
```

5. Config

Pengaturan pada aplikasi ini dapat ditemui pada beberapa file, yaitu routes\web.php, dan crud_pelanggan\.env.

Web.php

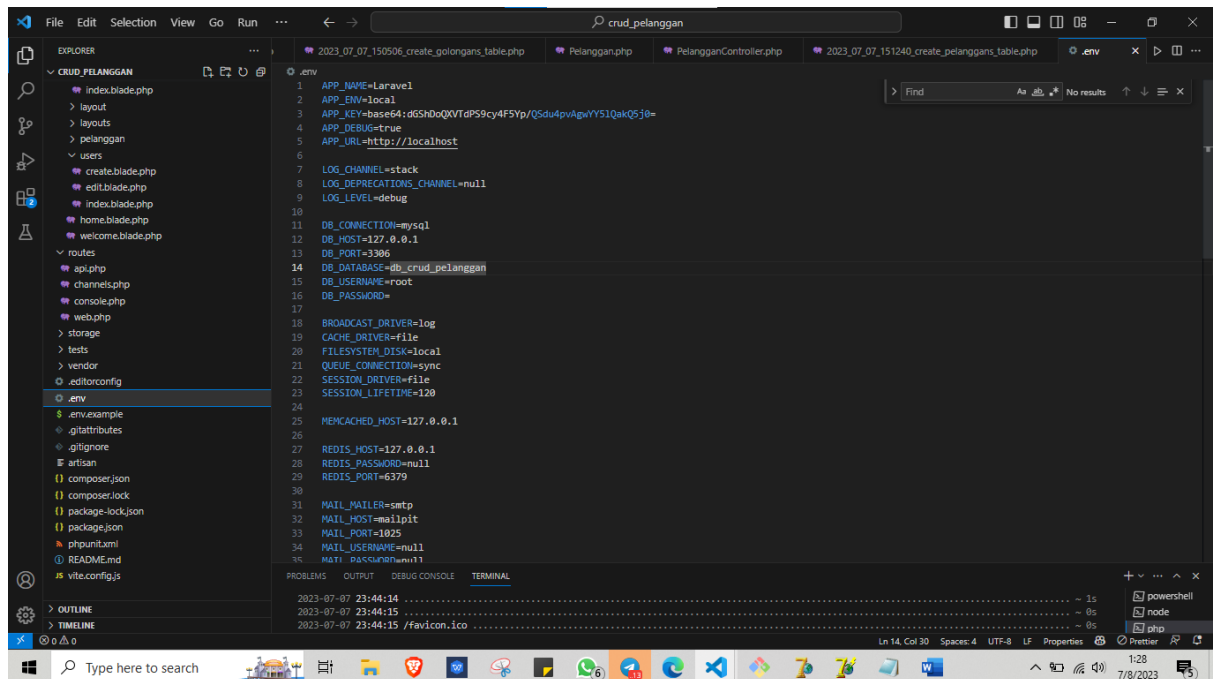
Web.php digunakan untuk melakukan pengaturan pada link url yang digunakan. File tersebut berisi url yang dapat diakses dan controller yang mengatur url tersebut.



```
1 <?php
2
3 use Illuminate\Support\Facades\Route;
4 use App\Http\Controllers\GolonganController;
5 use App\Http\Controllers\UserController;
6 use App\Http\Controllers\PelangganController;
7 use App\Http\Controllers\Auth\LoginController;
8
9
10 /*
11 |-----
12 | Web Routes
13 |-----
14 |
15 | Here is where you can register web routes for your application. These
16 | routes are loaded by the RouteServiceProvider and all of them will
17 | be assigned to the "web" middleware group. Make something great!
18 |
19 | */
20
21 Route::get('/', function () {
22     return view('welcome');
23 });
24
25 Route::get('golongan', [GolonganController::class, 'index'])->name('golongan.index');
26 Route::post('golongan/create', [GolonganController::class, 'create'])->name('golongan.create');
27 Route::put('golongan/{id}/edit', [GolonganController::class, 'edit'])->name('golongan.edit');
28 Route::delete('golongan/{id}', [GolonganController::class, 'destroy'])->name('golongan.destroy');
29
30 Route::get('/users', [UserController::class, 'index'])->name('users.index');
31 Route::post('/users/create', [UserController::class, 'create'])->name('users.create');
32 Route::post('/users', [UserController::class, 'store'])->name('users.store');
33 Route::get('/users/{id}/edit', [UserController::class, 'edit'])->name('users.edit');
34 Route::put('/users/{id}', [UserController::class, 'update'])->name('users.update');
35 Route::delete('/users/{id}', [UserController::class, 'destroy'])->name('users.destroy');
```

.env

.env merupakan file yang digunakan untuk mengatur segala bentuk setting yang berhubungan dengan properties aplikasi. Contohnya yaitu melakukan setting pada koneksi database. Dapat dilakukan dengan mengubah value dari db_host, db_port, db_database, db_username, dan db_password.



The screenshot shows a code editor with the following content in the .env file:

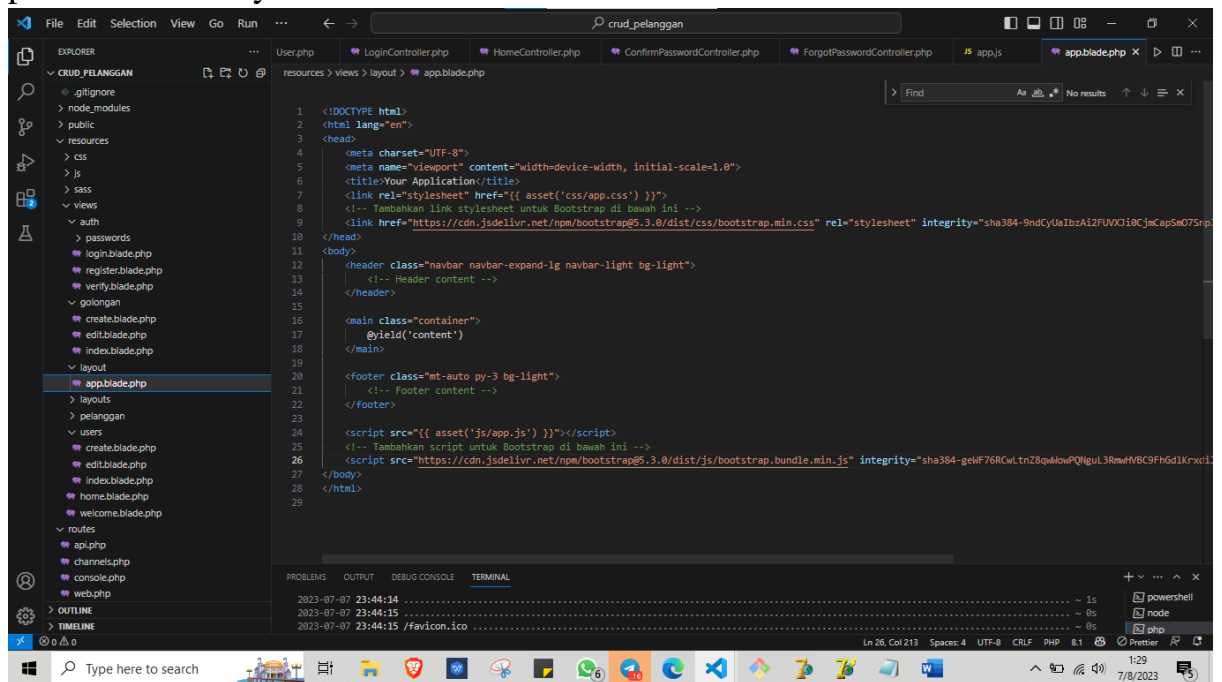
```
1 APP_NAME=Laravel
2 APP_ENV=local
3 APP_KEY=base64:dShDoQXVtdPS9cy4FSYp/QSdu4pvAgwYYS1QakQ5j0=
4 APP_DEBUG=true
5 APP_URL=http://localhost
6
7 LOG_CHANNEL=stack
8 LOG_DEPRECATIONS_CHANNEL=null
9 LOG_LEVEL=debug
10
11 DB_CONNECTION=mysql
12 DB_HOST=127.0.0.1
13 DB_PORT=3306
14 DB_DATABASE=db_crud_pelanggan
15 DB_USERNAME=root
16 DB_PASSWORD=
17
18 BROADCAST_DRIVER=log
19 CACHE_DRIVER=file
20 FILESYSTEM_DISK=local
21 QUEUE_CONNECTION=sync
22 SESSION_DRIVER=file
23 SESSION_LIFETIME=120
24
25 MEMCACHED_HOST=127.0.0.1
26
27 REDIS_HOST=127.0.0.1
28 REDIS_PASSWORD=null
29 REDIS_PORT=6379
30
31 MAIL_MAILER=smtp
32 MAIL_HOST=mailpit
33 MAIL_PORT=1025
34 MAIL_USERNAME=null
35 MAIL_PASSWORD=null
```

The editor's Explorer panel on the left shows a project structure with folders like crud_pelanggan, layout, layouts, pelanggan, users, and files like create.blade.php, edit.blade.php, index.blade.php, home.blade.php, welcome.blade.php, routes, api.php, channels.php, console.php, web.php, storage, tests, vendor, .editorconfig, .env, .env.example, .gitattributes, .gitignore, artisan, composer.json, composer.lock, package-lock.json, package.json, phpunit.xml, README.md, and vite.config.js. The bottom status bar indicates the file is at Line 14, Column 30, using UTF-8 encoding.

6. Bootstrap

Pada aplikasi ini digunakan bootstrap sebagai komponen untuk memperbagus tampilan pada aplikasi. Pemanggilan bootstrap dilakukan

pada direktori layout



```
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   <title>Your Application</title>
7   <link rel="stylesheet" href="{{ asset('css/app.css') }}">
8   <!-- Tambahkan link stylesheet untuk Bootstrap di bawah ini -->
9   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-9ndCylbAzA2F0V0i0CjMCapS075rp"
10 </head>
11 <body>
12   <header class="navbar navbar-expand-lg navbar-light bg-light">
13     <!-- Header content -->
14   </header>
15
16   <main class="container">
17     @yield('content')
18   </main>
19
20   <footer class="mt-auto py-3 bg-light">
21     <!-- Footer content -->
22   </footer>
23
24   <script src="{{ asset('js/app.js') }}"></script>
25   <!-- Tambahkan script untuk Bootstrap di bawah ini -->
26   <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js" integrity="sha384-geH76RwLtnZ8qMlowPQNgul3RmWf8C9fhGd1Krci"
27 </body>
28 </html>
29
```

III. GITHUB REPOSITORY

Untuk meninjau aplikasi lebih mendalam, saya telah membuat repository pada github untuk meletakkan kode program dari aplikasi yang telah saya buat ini. Berikut link githubnya : <https://github.com/adamsrg30/quiz-pbwl.git>