

LAPORAN WORKSHOP WEB APPLICATION FRAMEWORK



Nama : Siti Hannaniyah Sucipto

NIM : E412111104

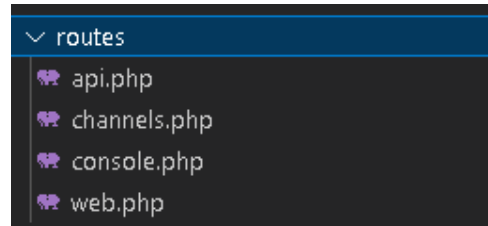
Golongan C

**Teknik Informatika
Jurusan Teknologi Informasi
Politeknik Negeri Jember
2023**

Acara 5

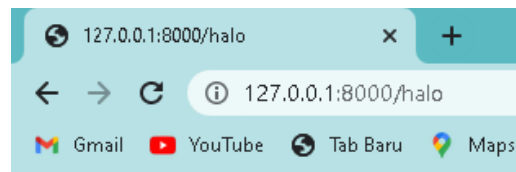
```
D:\>cd project
D:\project>cd project1
D:\project\project1>php artisan serve
Laravel development server started: http://127.0.0.1:8000
[Tue Mar  7 07:48:27 2023] PHP 8.0.11 Development Server (http://127.0.0.1:8000)
arted
```

Routes



1. Route Default

```
Route::get('/halo', function () {
    return ('halo');
});
```

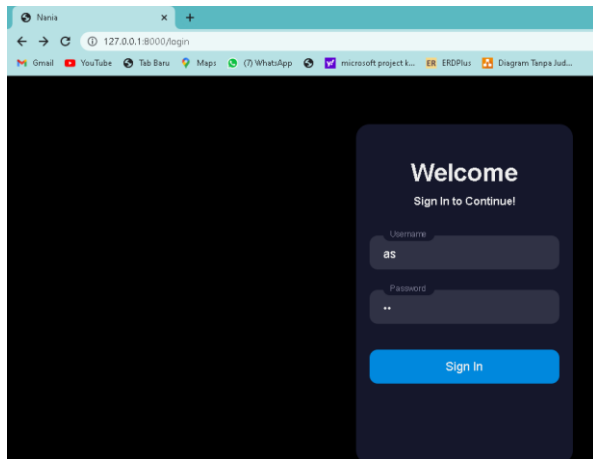


halo

2. Router get

```
Route::get('/login', function () {
    return view('login');
});
```

Maka akan masuk ke halaman welcome



3. **CSRF** merupakan sebuah fitur keamanan yang digunakan untuk mencegah serangan CSRF

```

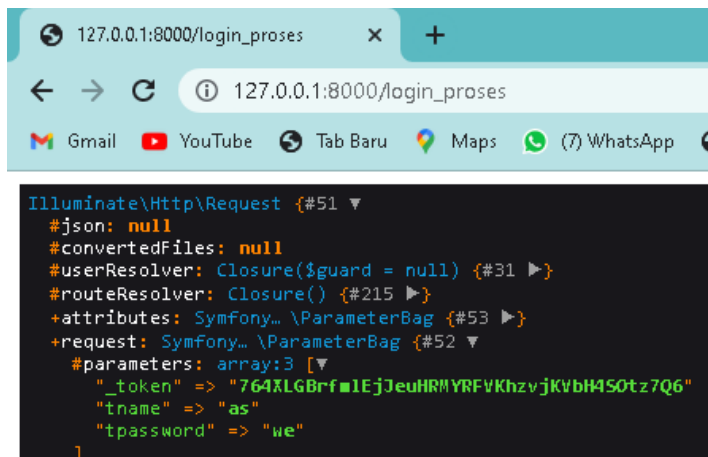
<body>
  <form method="POST" action="/login_proses">
    @csrf
    <div class="form">
      <div class="title">Welcome</div>
      <div class="subtitle">Sign In to Continue!</div>
      <div class="input-container ic1">
        <input id="username" class="input" type="text" placeholder=" " name="tname" />
        <div class="cut"></div>
        <label for="username" class="placeholder">Username</label>
      </div>
      <div class="input-container ic2">
        <input id="password" class="input" type="password" placeholder=" " name="tpassword">
        <div class="cut"></div>
        <label for="password" class="placeholder">Password</label>
      </div>
      <button type="text" class="submit">Sign In</button>
    </div>
  </form>
</body>
</html>

```

```

Route::POST('/login_proses', function (Request $request) {
    dd($request);
});

```

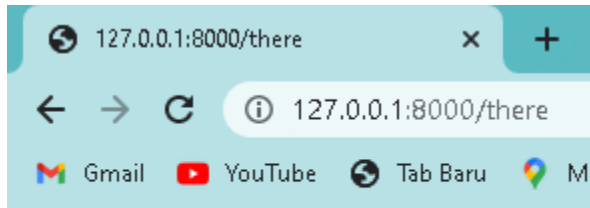


4. **Redirect route**

Berfungsi untuk mengalihkan pengguna ke URL lain dengan metode menggunakan metode **Route::redirect**

```
Route::get('/there', function () {  
    return "ini page untuk latihan route redirect";  
});  
  
Route::redirect('/here', '/there');  
// '/here' merupakan URL asal  
// '/there' merupakan URL tujuan
```

Maka akan diarahkan ke URL berikut

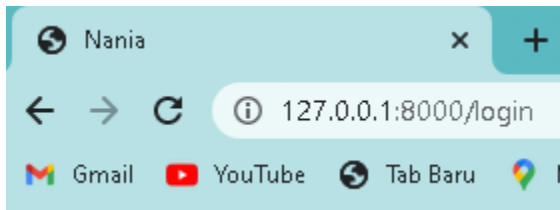


ini page untuk latihan route redirect

5. Route View

```
Route::view('/welcome', 'welcome', ['title' => 'Nania']);  
// route akan menjalankan ke folder view yang mengarah pada  
// file welcome dan mengubah title menjadi Nania.
```

Maka akan muncul halaman welcome dengan title nania



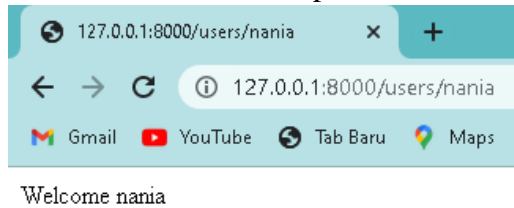
Acara 6

Router Parameter

Digunakan untuk menambahkan parameter dinamis ke dalam URL agar kita dapat mengakses data parameter tersebut dari controller

```
Route::get('users/{name?}', function($name = "nania") {  
    return "Welcome " . $name;  
});
```

Maka akan muncul output berikut



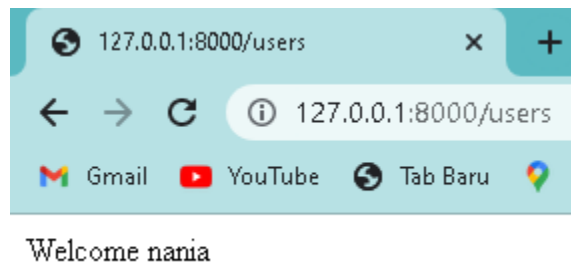
1. Parameter opsional

Bertugas untuk memberi nilai default ketika parameter tersebut tidak mempunyai nilai

Code 1:

```
Route::get ('users/{id?}', function($id = null) {  
    return " Welcome " . $id;  
});
```

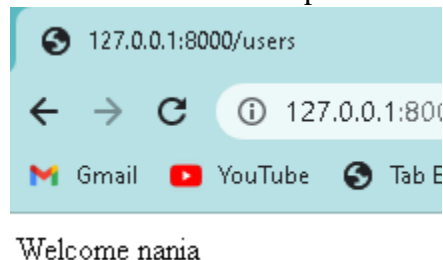
Maka akan muncul hasil berikut



Code 2

```
Route::get ('users/{id?}', function($id = nania) {  
    return " Welcome " . $id;  
});
```

Maka akan muncul output berikut:

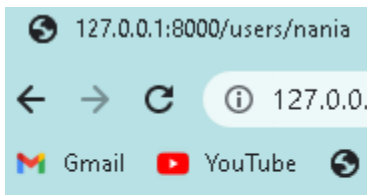


2. Regular Expression Constraints

Berfungsi untuk membatasi format penulisan nilai parameter route dengan metode where pada Route instance

```
Route::get('users/{name?}', function($name) {  
    return "Welcome " . $name;  
})->where('name', '[A-Za-z]+');
```

Maka akan muncul output berikut



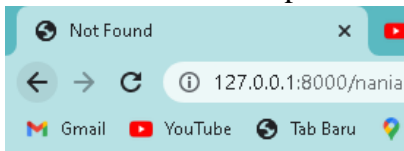
Welcome nania

3. Global Constraints

Berfungsi untuk selalu membatasi format penulisan nilai parameter route dengan method pattern.

```
public function boot()  
{  
    Route::pattern('name', '[A-Za-z]+');  
    parent::boot();  
}  
  
Route::get('users/{name}', function($name) {  
    return "Welcome " . $name;  
}));
```

Maka akan muncul output berikut

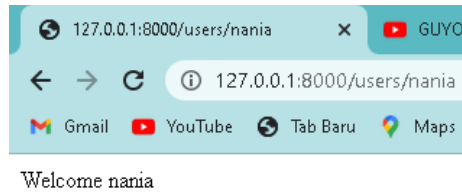


4. Encoded Forward Slashes

Pada Laravel nilai parameter pada routing tidak diizinkan menggunakan karakter '/' karena akan mencari rute baru, untuk mengizinkan '/' menjadi bagian dari placeholder kita dapat menggunakan kondisi expression regular where.

```
Route::get('users/{name?}', function($name = "nania") {  
    return "Welcome " . $name;  
})->where('name', '.*');
```

Maka, akan muncul output berikut:

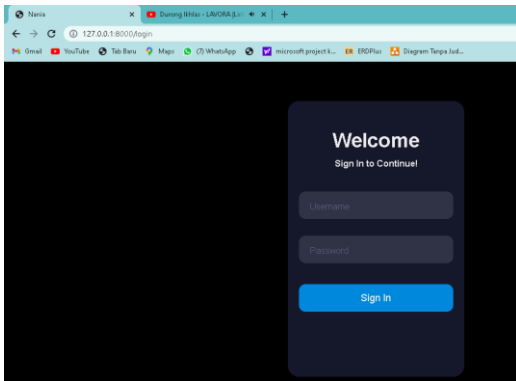


Acara 7

Named Route

Named Route atau bisa disebut dengan Route Bernama memungkinkan pembuatan nama URL yang mudah untuk route tertentu. Kita dapat menentukan nama route dengan merangka metode nama ke definisi route.

```
Route::view('/login', 'login', ['title' => "Login Page"])  
->name('login');
```



1. Generate URL ke Route Bernama

Untuk menetapkan nama untuk route tertentu, kita dapat menggunakan route tersebut dengan membuat URL atau redirect URL seperti berikut:

```
Route::get('/user', function() {  
    //generating URLs  
    $url = route('users');  
})->where('id', '.*');
```

```
Route::get('users/{id}', function($id = "nania"){  
    return "Welcome ".$id;  
})->where('id', '.*')->name('users');
```

Pada code di atas jika kita menuliskan url /user maka akan dialihkan ke route yang bernama users.

Jika route bernama mempunyai parameter, kita dapat menggunakan parameter tersebut sebagai argumen kedua dalam fungsi route seperti berikut:


```
Route::get('/user', function($id = "nania") {  
    //generating URLs  
    $url = route('users', [  
    ]);  
    //generating redirect  
    return redirect()->route('users', [  
    ]);  
})->where('id','.*');
```

2. Memeriksa Rute Saat Ini

Untuk menentukan apakah permintaan saat ini dirutekan ke route bernama tertentu, maka dapat digunakan metode named Route instance.

Acara 8

Group Routing

Group Routing bertujuan untuk menggabungkan beberapa rute yang serupa dalam satu tempat yang sama, sehingga lebih mudah dalam melakukan pengaturan atau pemeliharaan.

1. Middleware

Pada Laravel Middleware memungkinkan kita untuk memodifikasi request sebelum sampai ke controller dan response setelah keluar dari controller. Dalam Laravel Middleware sendiri biasa digunakan untuk melakukan validasi, otentikasi, pengaturan session dan lain sebagainya.

```
Route::prefix('admin')->middleware(['auth', 'admin'])->group(function () {
    Route::get('/users', function(){
        return 'ini halaman /admin/users';
    });
    Route::get('/home', function(){
        return 'ini halaman /admin/home';
    });
    Route::get('/report', function(){
        return 'ini halaman /admin/report';
    });
});
```

2. Namespace

Namespace pada Laravel dapat memungkinkan kita untuk mengorganisasi code-code yang kita buat menjadi kelompok-kelompok tertentu yang terpisah, sehingga memudahkan dalam pengelolaan dan pemeliharaan kode.

```
Route::namespace('Admin')->group(function () {
    // Controllers within the "App\Http\Controllers\Admin" namespace
    Route::get('Home', 'HomeController@index');
    Route::get('Report', 'ReportController@index');
});
```

3. Subdomain Routing

Subdomain pada Laravel memungkinkan kita untuk memisahkan rute-rute berdasarkan subdomain dan mengarahkannya ke controller yang berbeda-beda. Dalam menggunakan subdomain ini, kita menggunakan method domain. Ketika subdomain diakses Laravel akan mencocokkan subdomain tersebut dengan konfigurasi yang telah ditentukan dan akan mengarahkannya ke rute-rute yang sesuai.

```
Route::domain('nania.myapp.com')->group(function() {
    Route::get('user/{id}', function () {
        return 'coba';
    });
});
```

4. Route Prefix

Dengan menggunakan Route Prefix kita dapat mengelompokkan rute-rute yang memiliki kesamaan pada awal URL sehingga memudahkan dalam pengelolaan rute tersebut.

```
Route::prefix('admin')->group(function () {
    Route::get('/users', function(){
        return 'ini halaman /admin/users';
    });
    Route::get('/home', function(){
        return 'ini halaman /admin/home';
    });
    Route::get('/report', function(){
        return 'ini halaman /admin/report';
    });
});
```

Maka, akan muncul output berikut

