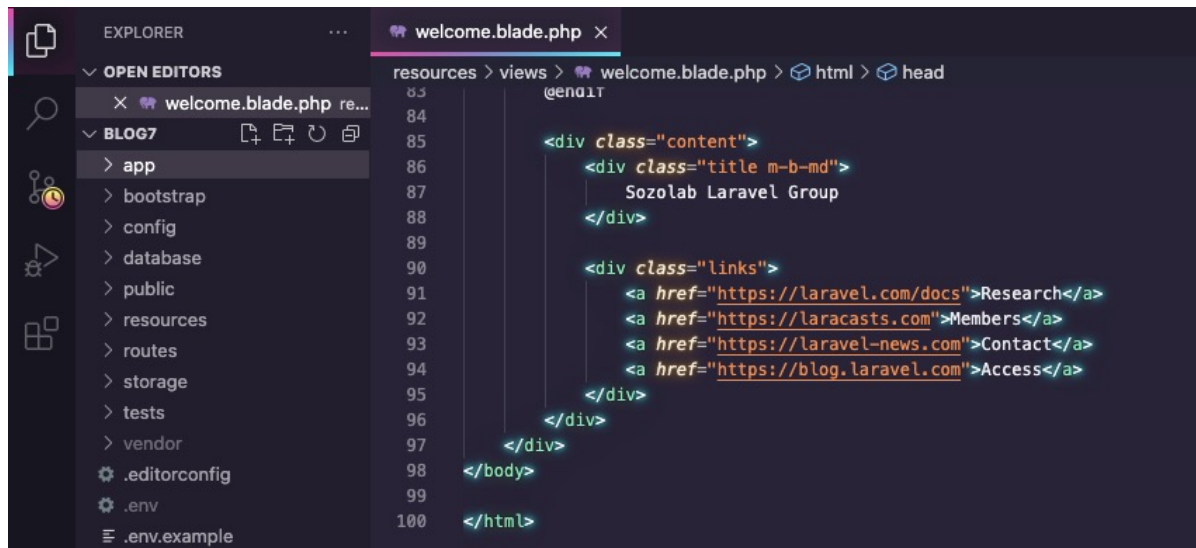




Route

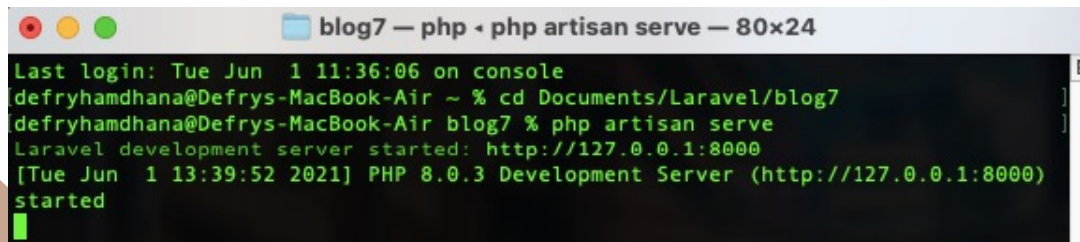
Lets open our Laravel project in text editor



The screenshot shows a text editor interface with a sidebar on the left displaying the project structure. The sidebar includes an 'EXPLORER' panel with a search icon, a list of 'OPEN EDITORS' (welcome.blade.php), and a 'BLOG7' directory containing 'app', 'bootstrap', 'config', 'database', 'public', 'resources', 'routes', 'storage', 'tests', 'vendor', '.editorconfig', '.env', and '.env.example'. The main editor area shows the 'welcome.blade.php' file with the following HTML code:

```
83 <@extends('layouts.app')>
84 <@section('content')>
85 <div class="content">
86 <div class="title m-b-md">
87     Sozolab Laravel Group
88 </div>
89
90 <div class="links">
91 <a href="https://laravel.com/docs">Research</a>
92 <a href="https://laracasts.com">Members</a>
93 <a href="https://laravel-news.com">Contact</a>
94 <a href="https://blog.laravel.com">Access</a>
95 </div>
96 </div>
97 </div>
98 </body>
99
100 </html>
```

2. Just for reminder, also running your web from terminal to view our website

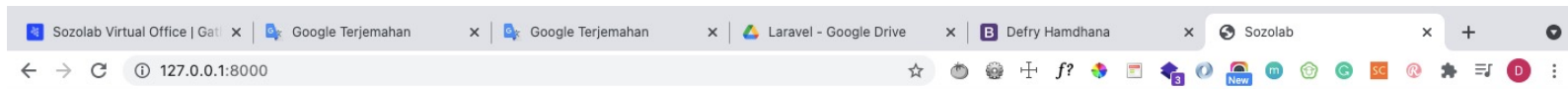


The screenshot shows a terminal window titled 'blog7 — php • php artisan serve — 80x24'. The terminal output is as follows:

```
Last login: Tue Jun  1 11:36:06 on console
defryhamdhana@Defrys-MacBook-Air ~ % cd Documents/Laravel/blog7
defryhamdhana@Defrys-MacBook-Air blog7 % php artisan serve
Laravel development server started: http://127.0.0.1:8000
[Tue Jun  1 13:39:52 2021] PHP 8.0.3 Development Server (http://127.0.0.1:8000)
started
```

NB: Paste the link, in this case (<http://127.0.0.1:8000>) in our browser.

And this is our dynamic website. Hehe. #jokes



Sozolab Laravel Group

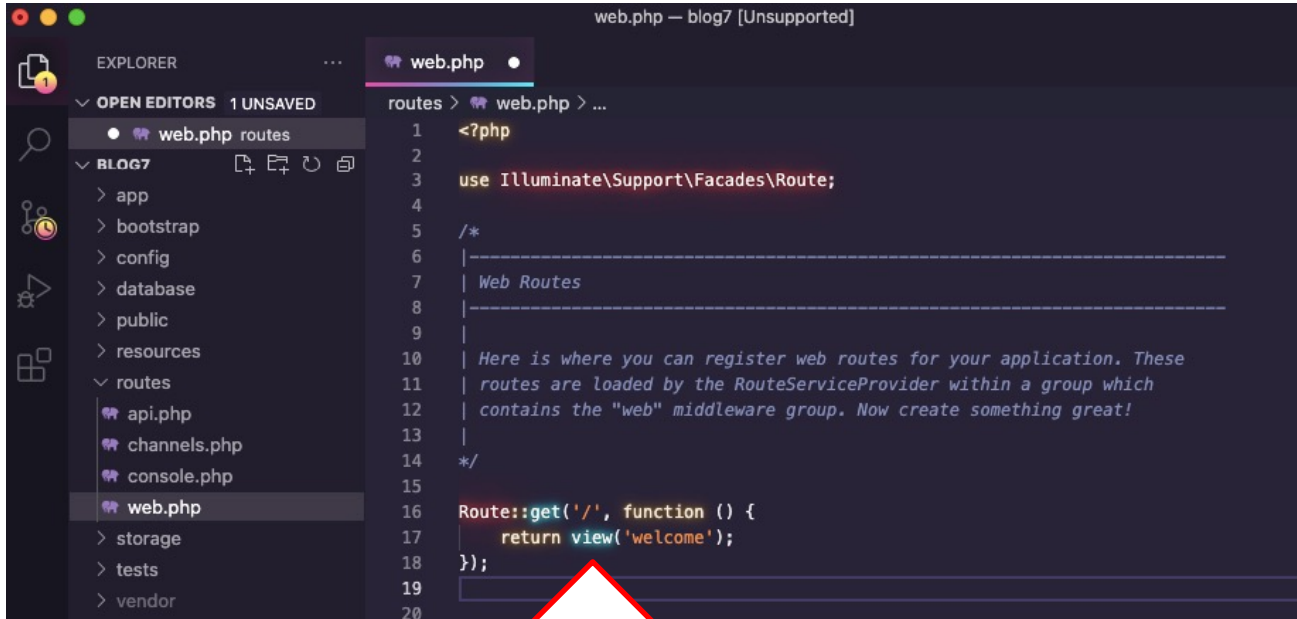
RESEARCH

MEMBERS

CONTACT

ACCESS

All basic laravel products start with the routing system. (routes -> web.php).

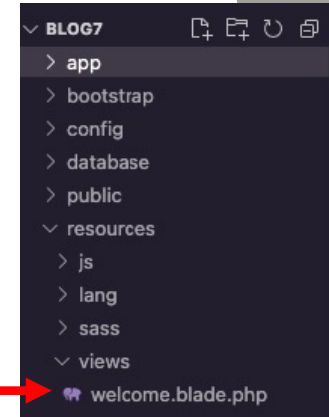


```
web.php — blog7 [Unsupported]

EXPLORER
  OPEN EDITORS 1 UNSAVED
    web.php routes
  BLOG7
    app
    bootstrap
    config
    database
    public
    resources
    routes
      api.php
      channels.php
      console.php
      web.php
    storage
    tests
    vendor

routes > web.php > ...
1  <?php
2
3  use Illuminate\Support\Facades\Route;
4
5  /*
6  |-----
7  | Web Routes
8  |-----
9  |
10 | Here is where you can register web routes for your application. These
11 | routes are loaded by the RouteServiceProvider within a group which
12 | contains the "web" middleware group. Now create something great!
13 |
14 */
15
16 Route::get('/', function () {
17     return view('welcome');
18 });
19
20
```

when a user asks to access / (the first part of) a website, then website will showing view → welcome

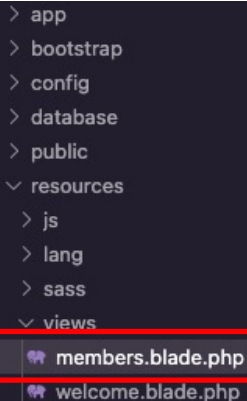


```
EXPLORER
  BLOG7
    app
    bootstrap
    config
    database
    public
    resources
    js
    lang
    sass
    views
      welcome.blade.php
```

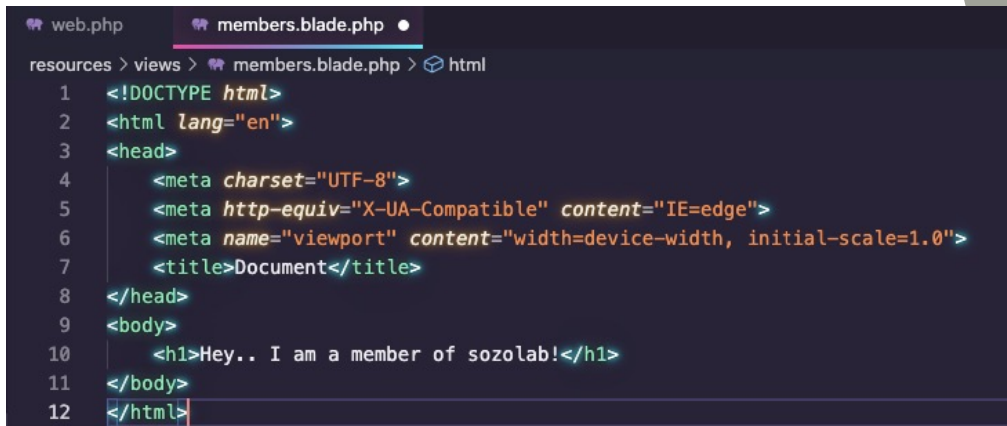
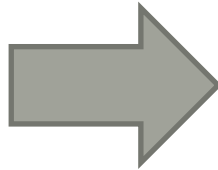
Now let's try to create our own route

```
16 Route::get('/', function () {  
17     return view('welcome');  
18 });  
19  
20 Route::get('/members', function () {  
21     return view('members');  
22 });  
23
```

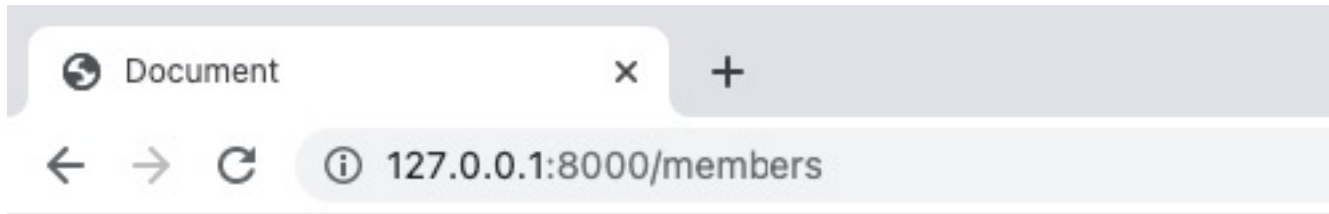
The above command asks the system to call a view named members, therefore we have to create members.blade.php in the views folder.



```
> app  
> bootstrap  
> config  
> database  
> public  
> resources  
  > js  
  > lang  
  > sass  
  > views  
    members.blade.php  
    welcome.blade.php
```

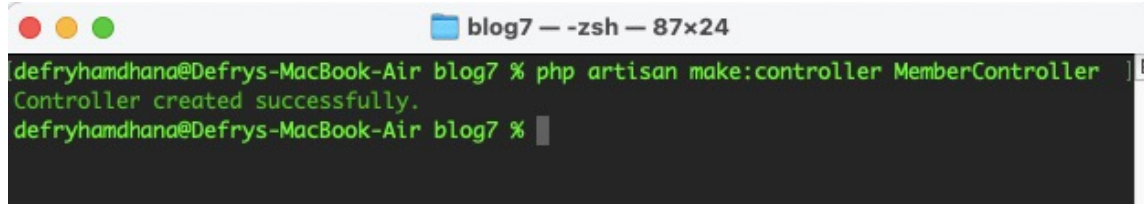


```
web.php  members.blade.php ●  
resources > views > members.blade.php > html  
1 <!DOCTYPE html>  
2 <html lang="en">  
3 <head>  
4     <meta charset="UTF-8">  
5     <meta http-equiv="X-UA-Compatible" content="IE=edge">  
6     <meta name="viewport" content="width=device-width, initial-scale=1.0">  
7     <title>Document</title>  
8 </head>  
9 <body>  
10     <h1>Hey.. I am a member of sozolab!</h1>  
11 </body>  
12 </html>
```



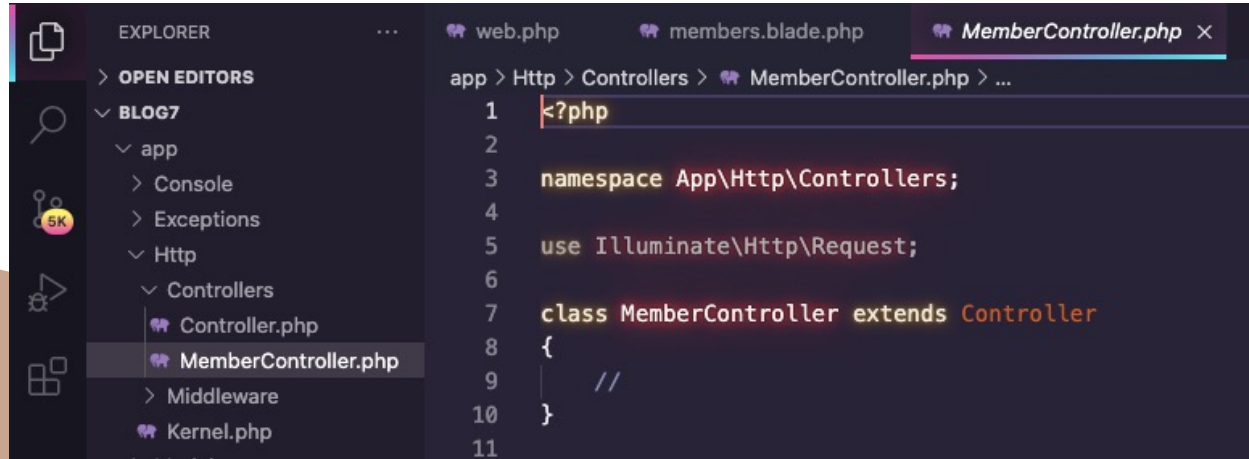
Hey.. I am a member of sozolab!

Now we will try to create a controller. To create a controller we can use composer with the command “php artisan make:controller Name_Controller.



```
blog7 — -zsh — 87x24
defryhamdhana@Defrys-MacBook-Air blog7 % php artisan make:controller MemberController
Controller created successfully.
defryhamdhana@Defrys-MacBook-Air blog7 %
```

We can see the result of the new controller in app/Http/Controllers



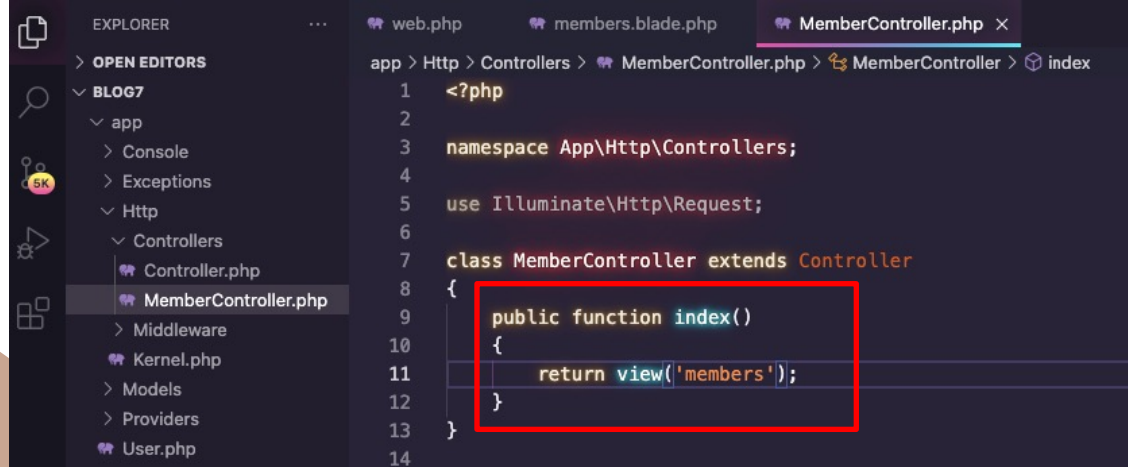
```
EXPLORER
OPEN EDITORS
BLOG7
  app
    Console
    Exceptions
    Http
      Controllers
        MemberController.php
        Controller.php
    Middleware
    Kernel.php

web.php  members.blade.php  MemberController.php x
app > Http > Controllers > MemberController.php > ...
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  class MemberController extends Controller
8  {
9      //
10 }
11
```

To access the controller, we will return to web.php, now we will modify the route members that we previously created.

```
15
16 Route::get('/', function () {
17     return view('welcome');
18 });
19
20 Route::get('/members', 'MemberController@index');
21
```

We can see the result of the new controller in `app/Http/Controllers/MemberController.php`. Then we will create a function index.



The screenshot shows an IDE with the Explorer panel on the left and the code editor on the right. The Explorer panel shows the project structure with 'app/Http/Controllers/MemberController.php' selected. The code editor shows the contents of 'MemberController.php', which includes a namespace declaration, a use statement for 'Illuminate\Http\Request', and a class 'MemberController' extending 'Controller'. The 'index()' method is highlighted with a red box.

```
app > Http > Controllers > MemberController.php > MemberController > index
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6
7  class MemberController extends Controller
8  {
9      public function index()
10     {
11         return view('members');
12     }
13 }
14
```

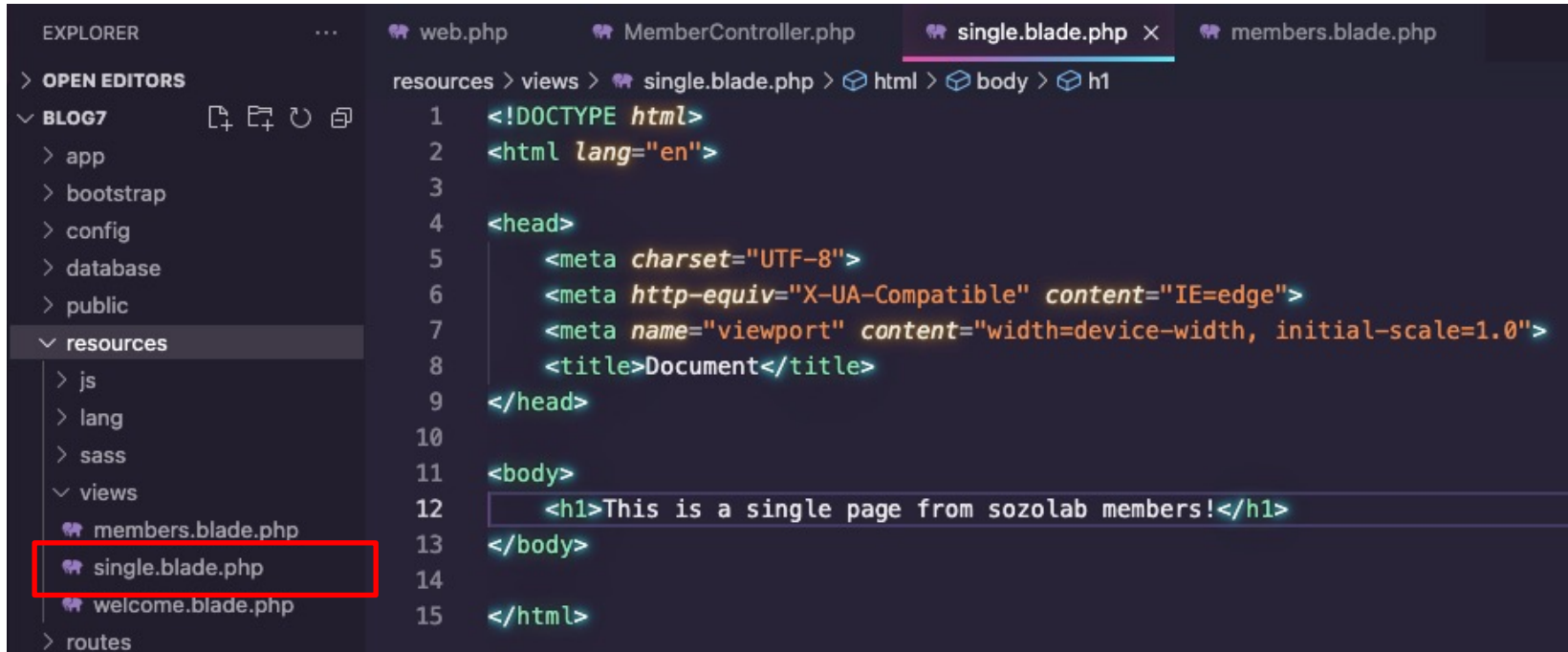

Now let's try to create another function to describe the individual members information.

```
16 Route::get('/', function () {  
17     return view('welcome');  
18 });  
19  
20 Route::get('/members', 'MemberController@index');  
21 Route::get('/members/1', 'MemberController@show');  
22
```

Back again to the controller to create a function show in MemberController

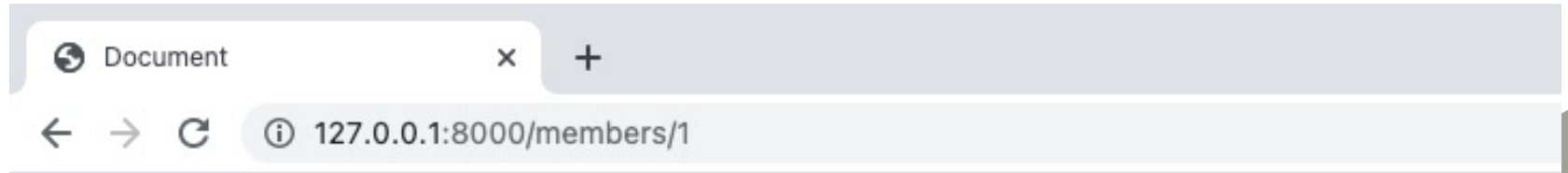
```
6  
7 class MemberController extends Controller  
8 {  
9     public function index()  
10    {  
11        return view('members');  
12    }  
13  
14    public function show()  
15    {  
16        return view('single');  
17    }  
18 }  
19
```

Let's create a view ➔ single.blade.php to showing a single menu on the website.



The screenshot shows a code editor with a dark theme. On the left, the 'EXPLORER' sidebar displays a file tree. Under the 'resources' folder, the 'views' subfolder is expanded, and 'single.blade.php' is highlighted with a red rectangle. The main editor area shows the content of 'single.blade.php' with the following code:

```
resources > views > single.blade.php > html > body > h1
1  <!DOCTYPE html>
2  <html lang="en">
3
4  <head>
5      <meta charset="UTF-8">
6      <meta http-equiv="X-UA-Compatible" content="IE=edge">
7      <meta name="viewport" content="width=device-width, initial-scale=1.0">
8      <title>Document</title>
9  </head>
10
11 <body>
12     <h1>This is a single page from sozolab members!</h1>
13 </body>
14
15 </html>
```



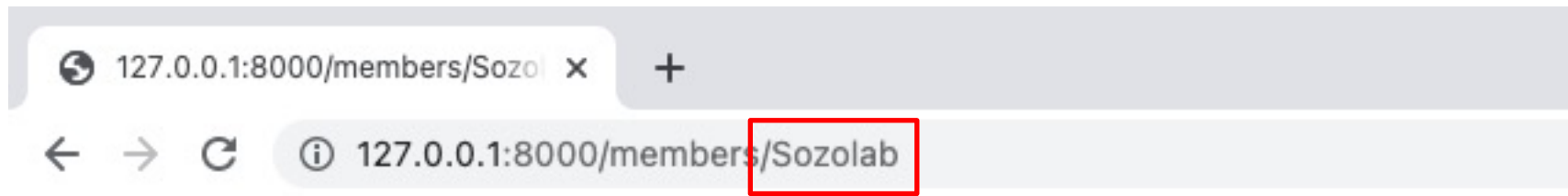
This is a single page from sozolab members!

Now we will try to make the pages of this website to be more dynamic. First, let's modify the route whose job is to showing a single menu on the website. Let's make a view → single.blade.php untuk show menu single in website.

```
20 Route::get('/members', 'MemberController@index');  
21 Route::get('/members/{slug}', 'MemberController@show');
```

After that we will also modify the show function in MemberController.php to see that the link we are calling is now dynamic.

```
14 public function show($slug)  
15 {  
16     die($slug);  
17     return view('single');  
18 }
```



Sozolab

Thank you 😊

