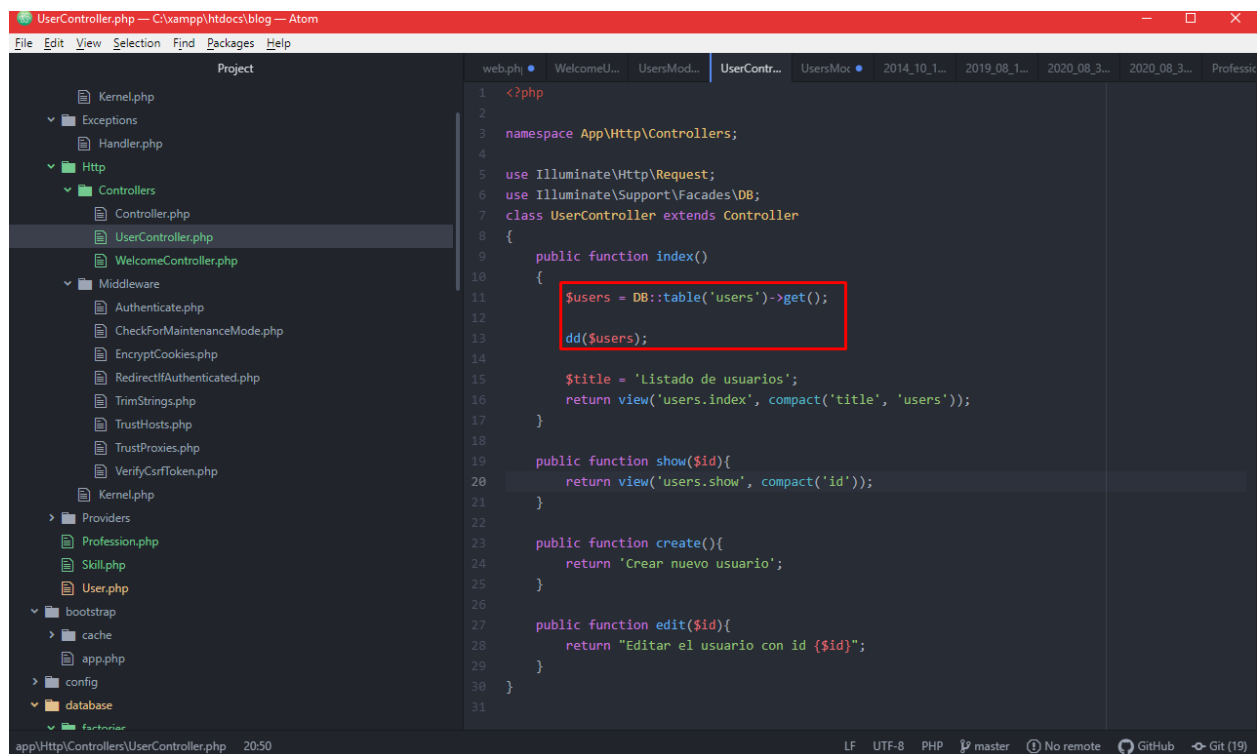


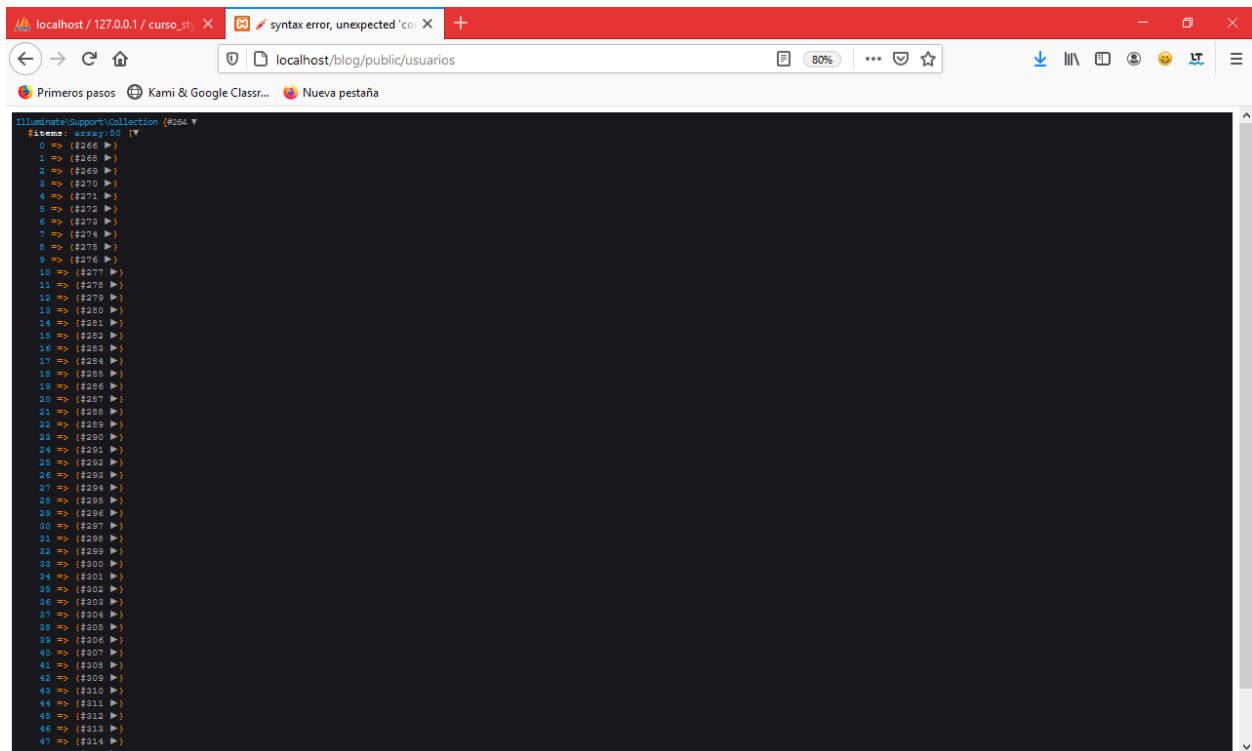
Actividad 3 Laravel: Listado básico

Rossana Palma López

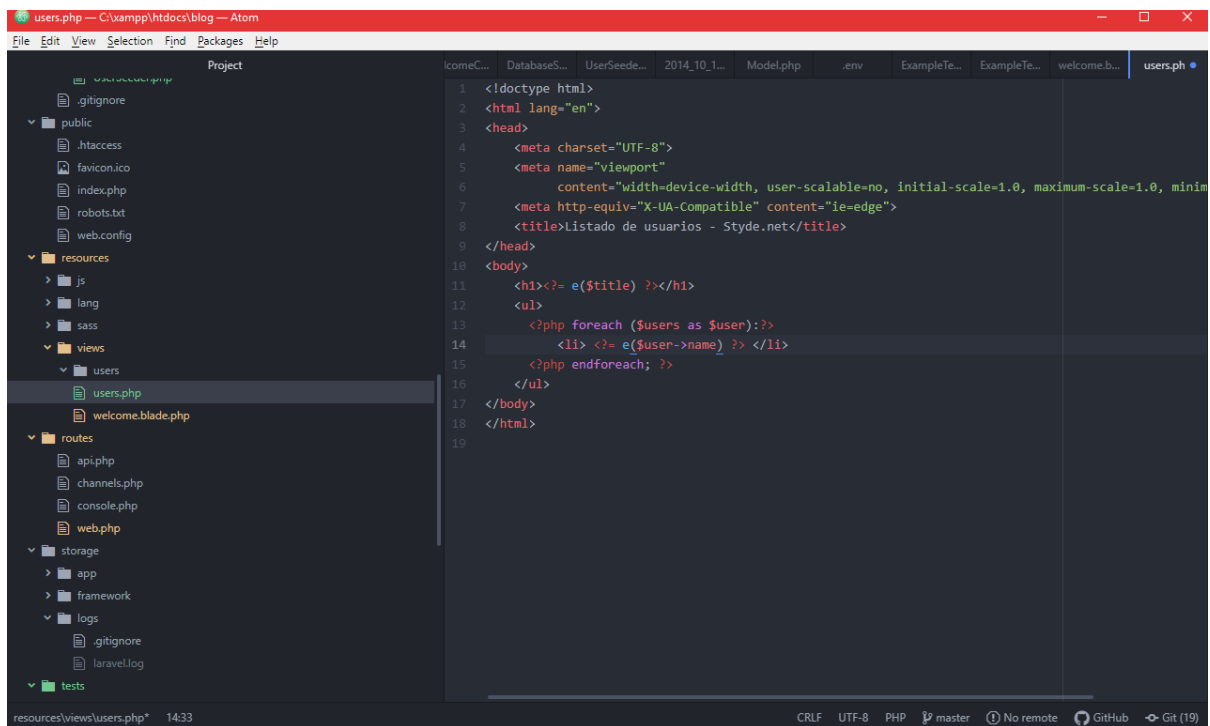


```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\DB;
7 class UserController extends Controller
8 {
9     public function index()
10     {
11         $users = DB::table('users')->get();
12         dd($users);
13
14         $title = 'Listado de usuarios';
15         return view('users.index', compact('title', 'users'));
16     }
17
18     public function show($id){
19         return view('users.show', compact('id'));
20     }
21
22     public function create(){
23         return 'Crear nuevo usuario';
24     }
25
26     public function edit($id){
27         return "Editar el usuario con id {$id}";
28     }
29 }
30
31
```

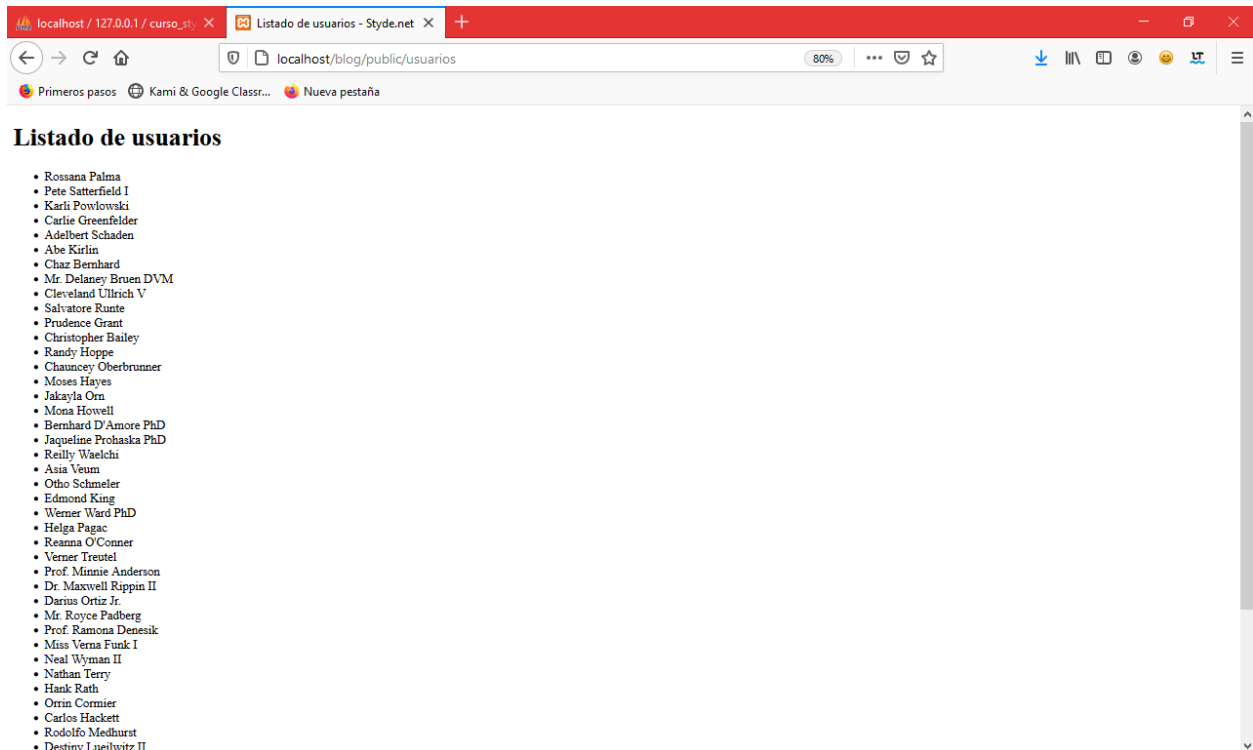
Modificamos el controlador *UserController* y declaramos una lista dinámica de usuarios, usando el constructor de consultas de Laravel, esta consulta será obtener todos los registros en la tabla **users**.



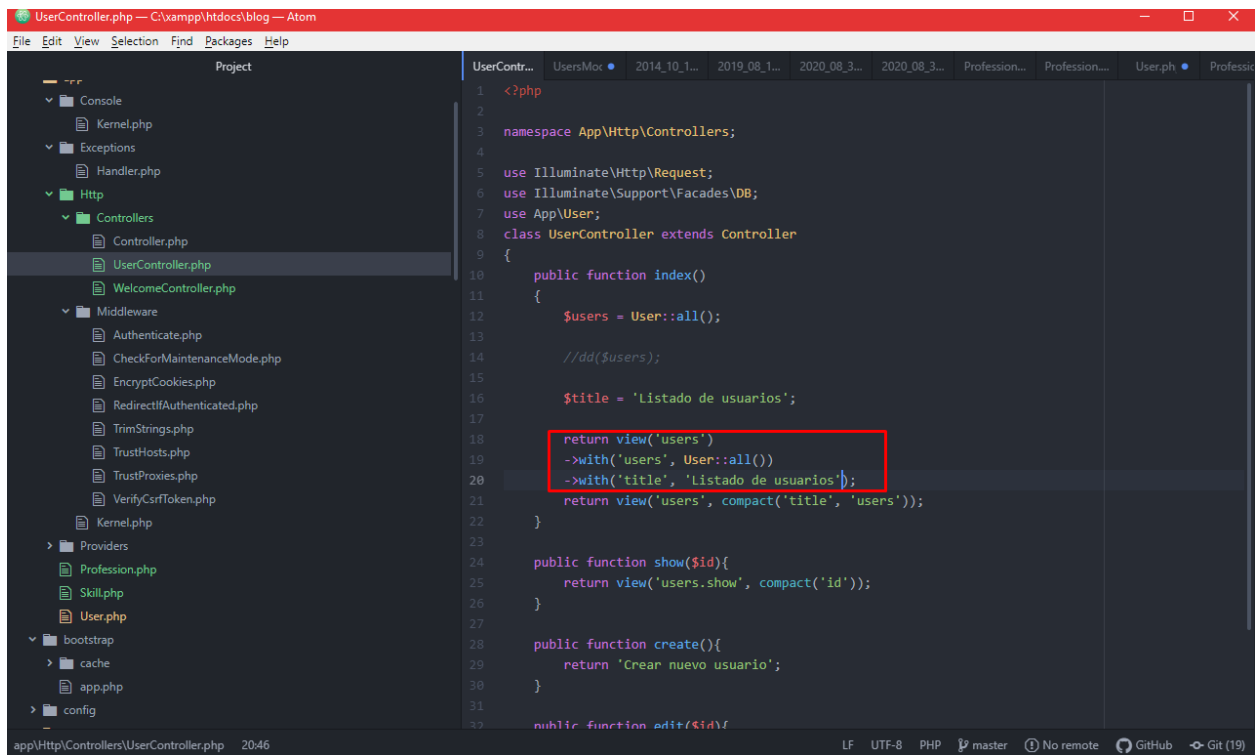
Gracias a `dd` podemos ver en el servidor local que obtenemos una colección con todos los usuarios.



Al ser del tipo User la colección que hemos obtenido, para que nuestra lista solo nos muestre el atributo name de un user.

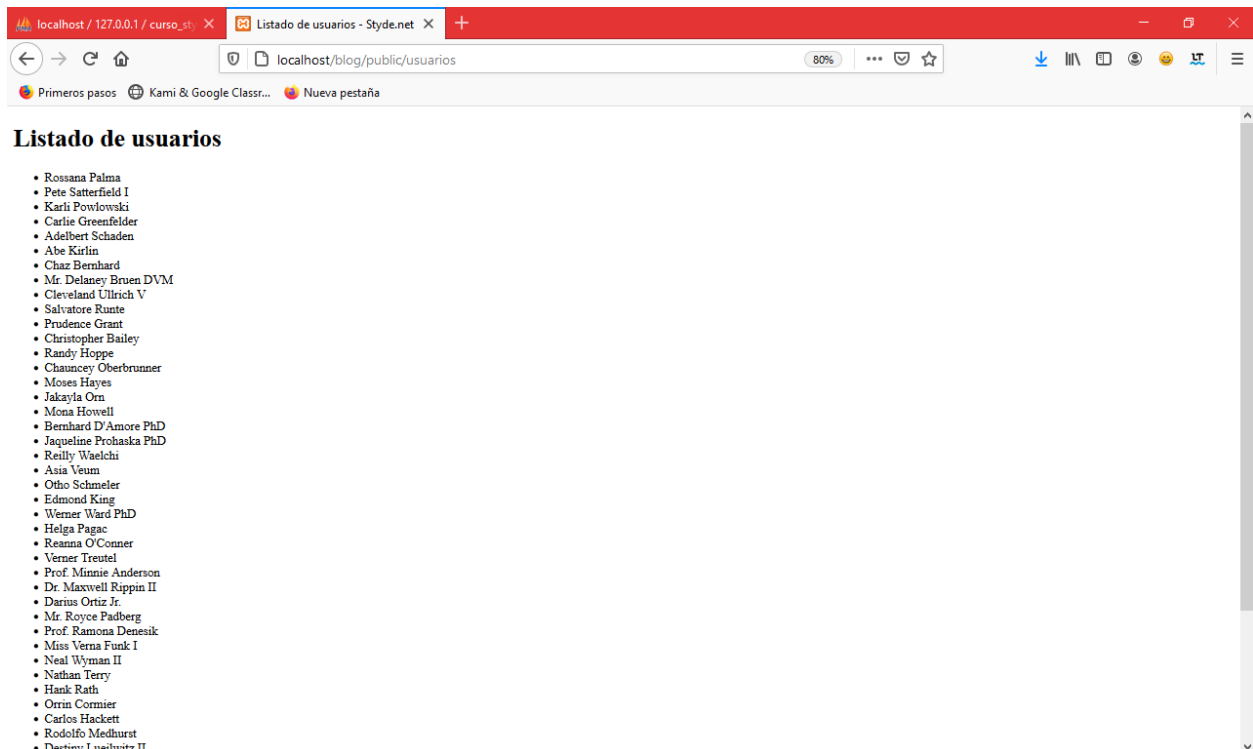


Ahora en la vista vemos reflejado los usuarios que se encuentran en la tabla **users**.



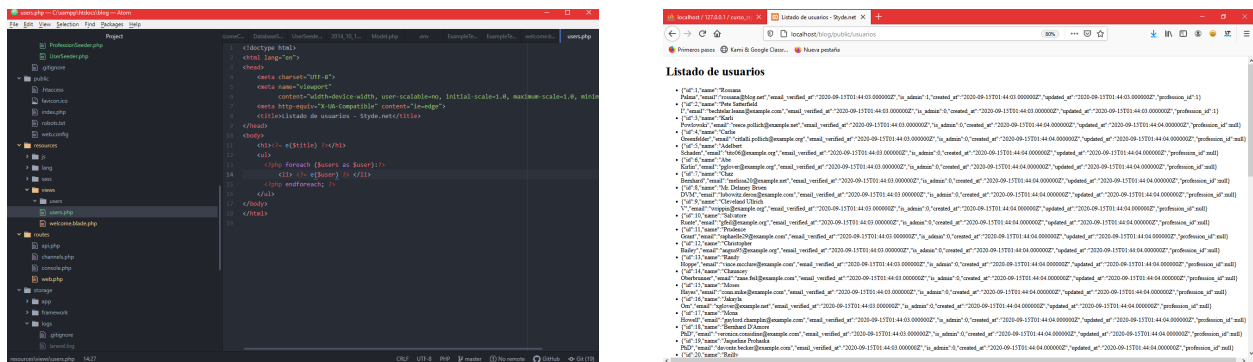
```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6 use Illuminate\Support\Facades\DB;
7 use App\User;
8 class UserController extends Controller
9 {
10     public function index()
11     {
12         $users = User::all();
13
14         //dd($users);
15
16         $title = 'Listado de usuarios';
17
18         return view('users')
19             ->with('users', User::all())
20             ->with('title', 'Listado de usuarios');
21     }
22
23     public function show($id){
24         return view('users.show', compact('id'));
25     }
26
27     public function create(){
28         return 'Crear nuevo usuario';
29     }
30
31     public function edit($id){
```

De la misma manera podemos obtener los registros de la tabla **users** usando Eloquent.

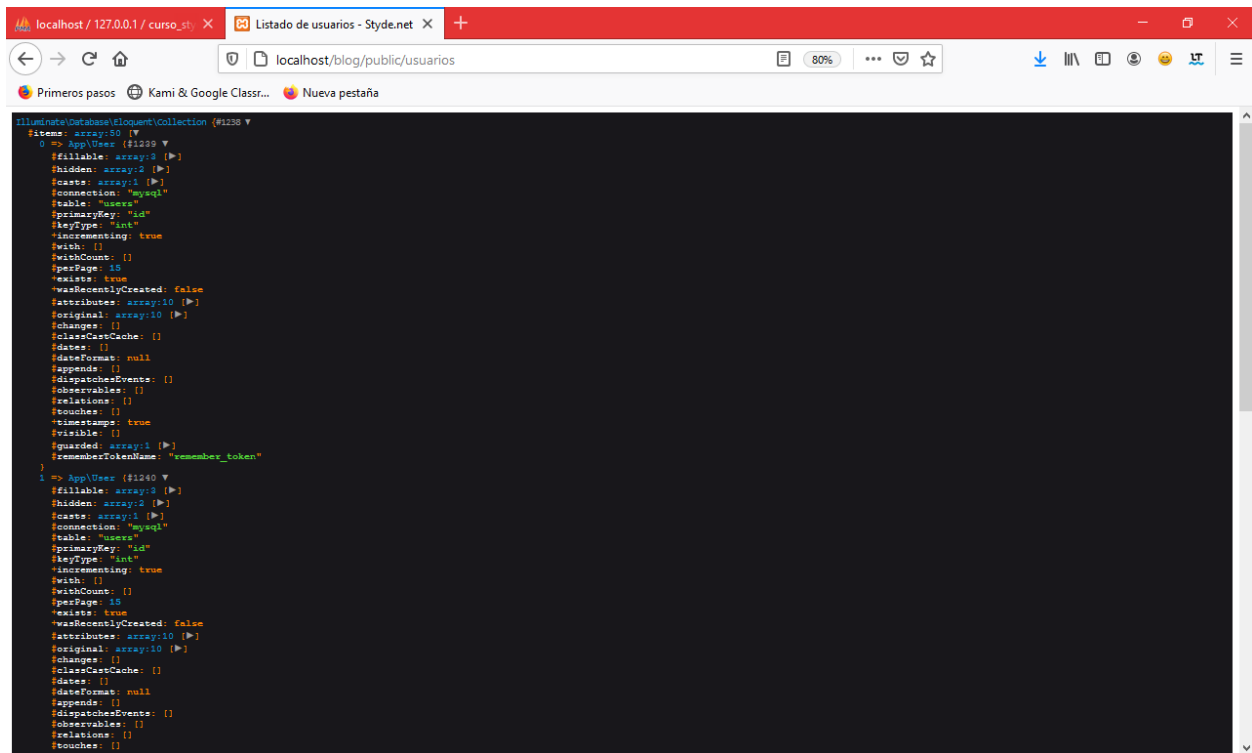


Y seguimos obteniendo en la vista la lista de todos los usuarios registrados en nuestra base de datos.

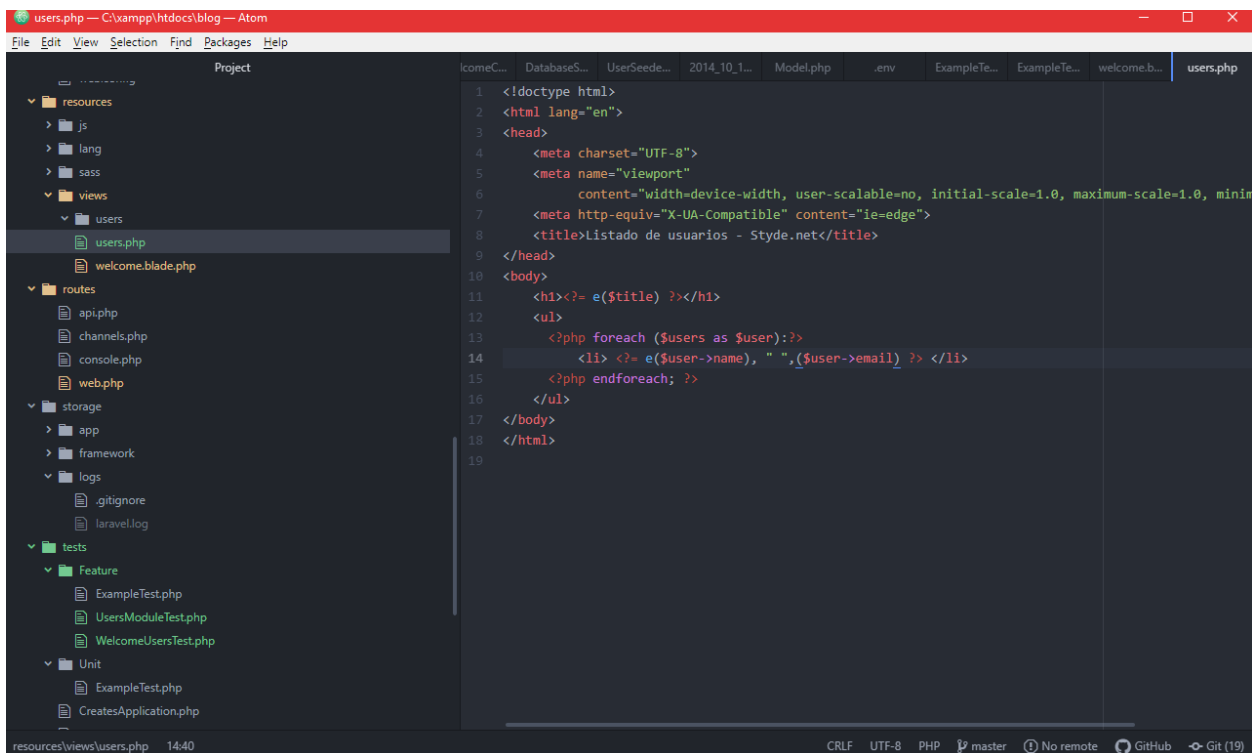
Ahora si dejamos de llamar a la propiedad *name* de `$user` en la vista obtenemos lo siguiente



Dado que cada usuario representa un ejemplar de la clase *User*, es decir de un Modelo de Eloquent. Esto es posible gracias a Eloquent. El método `__toString` de la clase Modelo de Eloquent que se encarga de convertir un modelo a json.



```
Illuminate\Database\Eloquent\Collection (#1238 ▼)
#items: array:10 (▼)
0 => App\User (#1239 ▼)
#fillable: array:3 (▼)
#hidden: array:2 (▼)
#casts: array:1 (▼)
#connection: "mysql"
#table: "users"
#primaryKey: "id"
#keyType: "int"
#incrementing: true
#with: []
#withCount: []
#perPage: 15
#exists: true
#wasRecentlyCreated: false
#attributes: array:10 (▼)
#original: array:10 (▼)
#changes: []
#classCastCache: []
#dates: []
#dateFormat: null
#appends: []
#dispatchesEvents: []
#observable: []
#relations: []
#touches: []
#timestamps: true
#visible: []
#guarded: array:1 (▼)
#rememberTokenName: "remember_token"
1 => App\User (#1240 ▼)
#fillable: array:3 (▼)
#hidden: array:2 (▼)
#casts: array:1 (▼)
#connection: "mysql"
#table: "users"
#primaryKey: "id"
#keyType: "int"
#incrementing: true
#with: []
#withCount: []
#perPage: 15
#exists: true
#wasRecentlyCreated: false
#attributes: array:10 (▼)
#original: array:10 (▼)
#changes: []
#classCastCache: []
#dates: []
#dateFormat: null
#appends: []
#dispatchesEvents: []
#observable: []
#relations: []
#touches: []
```



```
1 <doctype html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport"
6         content="width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0">
7     <meta http-equiv="X-UA-Compatible" content="ie=edge">
8     <title>Listado de usuarios - Styde.net</title>
9 </head>
10 <body>
11     <h1><?= e($title) ?></h1>
12     <ul>
13         <?php foreach ($users as $user):?>
14             <li><?= e($user->name), " ", ($user->email) ?> </li>
15         <?php endforeach; ?>
16     </ul>
17 </body>
18 </html>
19
```

Ahora le pedimos a la vista que nos muestre junto al nombre de usuario su correo.

