

Pràctica 4:

Objectiu: Fer servir el vostre projecte de Laravel per tal que altres aplicacions web emprin la vostra base de dades d'usuaris per a loguejar-se. És a dir, implementar un servidor oauth2.

Guies:

<https://blog.pusher.com/make-an-oauth2-server-using-laravel-passport/>

1 – Li diem al composer que ens instali el paquet passport:

composer require laravel/passport

```
adria@adria-HP-ProDesk-600-G1-SFF:~/Escritorio/Servidor/miweb/projecte_rellotges
$ composer require laravel/passport
Using version ^8.4 for laravel/passport
./composer.json has been updated
Loading composer repositories with package information
Updating dependencies (including require-dev)
Package operations: 13 installs, 0 updates, 0 removals
  - Installing symfony/psr-http-message-bridge (v2.0.0): Downloading (connecting
Downloading (100%)
```

2 – Un cop instal·lat, afegim el proveïdor al document config/app.php.

Allà dins de:

```
137     'providers' => [
138
```

Inserim la següent línia:

Laravel\Passport\PassportServiceProvider::class,

```
app.php  X  web.php
home > adria > Escritorio > Servidor > miweb > projecte_rellotges > config > app.php
177     App\Providers\RouteServiceProvider::class,
178     Laravel\Passport\PassportServiceProvider::class,
179
```

3 - A continuació fem un migrate perquè ens creei les taules necessaries:

php artisan migrate

```
adria@adria-HP-ProDesk-600-G1-SFF:~/Escritorio/Servidor/miweb/projecte_rellotges
$ php artisan migrate
Migrating: 2016_06_01_000001_create_oauth_auth_codes_table
Migrated: 2016_06_01_000001_create_oauth_auth_codes_table (0.21 seconds)
Migrating: 2016_06_01_000002_create_oauth_access_tokens_table
Migrated: 2016_06_01_000002_create_oauth_access_tokens_table (0.16 seconds)
Migrating: 2016_06_01_000003_create_oauth_refresh_tokens_table
Migrated: 2016_06_01_000003_create_oauth_refresh_tokens_table (0.17 seconds)
Migrating: 2016_06_01_000004_create_oauth_clients_table
Migrated: 2016_06_01_000004_create_oauth_clients_table (0.09 seconds)
Migrating: 2016_06_01_000005_create_oauth_personal_access_clients_table
Migrated: 2016_06_01_000005_create_oauth_personal_access_clients_table (0.05 seconds)
```

4 - Creem un parell de claus públiques i privades:

php artisan passport:install

```
adria@adria-HP-ProDesk-600-G1-SFF:~/Escritorio/Servidor/miweb/projecte_rellotges
$ php artisan passport:install
Encryption keys generated successfully.
Personal access client created successfully.
Client ID: 1
Client secret: v5YKZyRaFZDPkgHeJh8V8YPRshQxiBBnQ3ngu0EV
Password grant client created successfully.
Client ID: 2
Client secret: pcI1Zx1PsLjUFeZYAkcvstXPSCiQ8o4RHNwxde5D
```

5 – A continuació anem a app/User.php i en la línia 5 i en la línia 11 les deixem com es mostra la captura:

```
1  <?php
2
3  namespace App;
4
5  use Laravel\Passport\HasApiTokens;
6  use Illuminate\Foundation\Auth\User as Authenticatable;
7  use Illuminate\Notifications\Notifiable;
8
9  class User extends Authenticatable
10 {
11     use HasApiTokens, Notifiable;
```

6 - Després, anem a app/Providers/AuthServiceProvider.php afegim just al final de la funció boot:

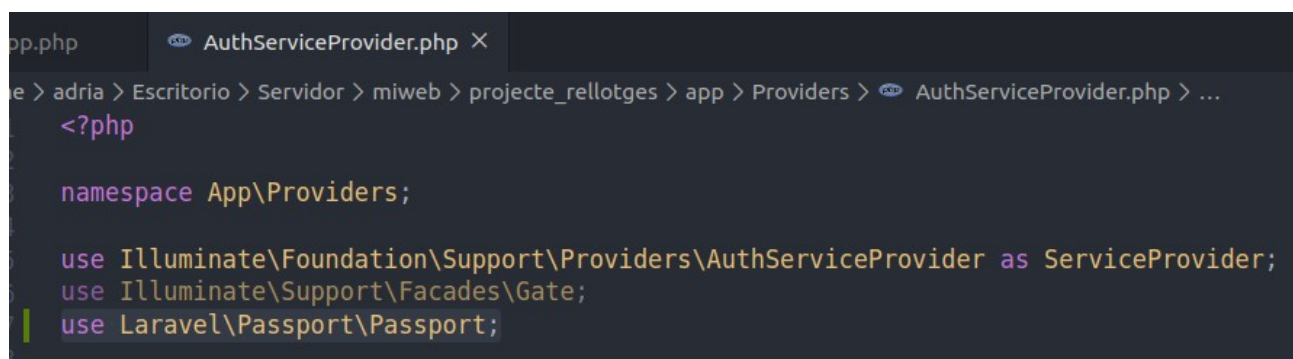
Passport::routes();



```
AuthServiceProvider.php X
home > adria > Escritorio > Servidor > miweb > projecte_relo
22      * @return void
23      */
24      public function boot()
25      {
26          $this->registerPolicies();
27
28          Passport::routes();
29      }
30  }
31
```

I posem el use corresponent al principi del document:

use Laravel\Passport\Passport;

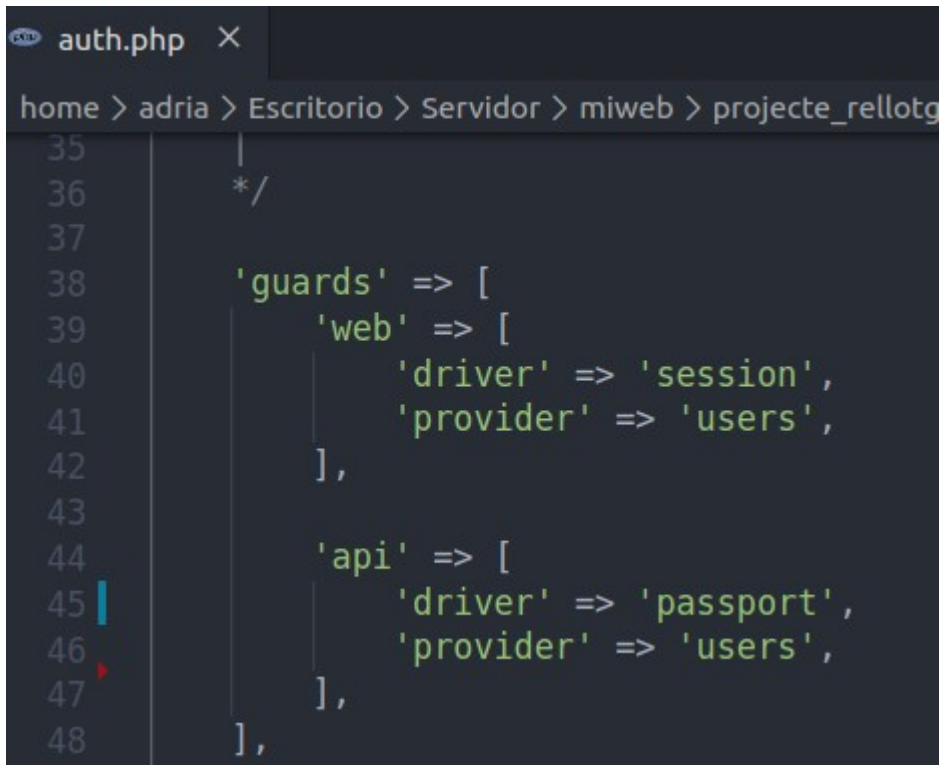


```
AuthServiceProvider.php X
e > adria > Escritorio > Servidor > miweb > projecte_rellotges > app > Providers > AuthServiceProvider.php > ...
<?php

namespace App\Providers;

use Illuminate\Foundation\Support\Providers\AuthServiceProvider as ServiceProvider;
use Illuminate\Support\Facades\Gate;
use Laravel\Passport\Passport;
```

7- Un cop fet, anem a config/auth.php i en la línia 45, on diu token possem passport i la línia 47 la borrem:



```
auth.php x
home > adria > Escritorio > Servidor > miweb > projecte_rellotg
35 |
36 | */
37 |
38 | 'guards' => [
39 |     'web' => [
40 |         'driver' => 'session',
41 |         'provider' => 'users',
42 |     ],
43 |
44 |     'api' => [
45 |         'driver' => 'passport',
46 |         'provider' => 'users',
47 |     ],
48 | ],
```

8- A continuació publiquem alguns components Vue amb la següent comanda:

php artisan vendor:publish --tag=passport-components

```
adria@adria-HP-ProDesk-600-G1-SFF:~/Escritorio/Servidor/miweb/proyecto_rellotges
$ php artisan vendor:publish --tag=passport-components
Copied Directory [/vendor/laravel/passport/resources/js/components] To [/resources/js/components/passport]
Publishing complete.
Publishing complete.
```

9- Anem al document resources/js/app.js i afegim al final del document, abans de 'const app = new Vue({', les següents línies de codi:

```
auth_redirection.php User.php JS app.js callback.php
home > adria > Escritorio > Servidor > miweb > proyecto_rellotges > resources > js > JS app.js > .
30
31 Vue.component(
32   'passport-authorized-clients',
33   require('./components/passport/AuthorizedClients.vue')
34 );
35
36 Vue.component(
37   'passport-clients',
38   require('./components/passport/Clients.vue')
39 );
40
41 Vue.component(
42   'passport-personal-access-tokens',
43   require('./components/passport/PersonalAccessTokens.vue')
44 );
45
46
47 const app = new Vue({
48   el: '#app',
49 });
50
```


10- Instal·lem i executem el paquet Node:

npm install

```
adria@adria-HP-ProDesk-600-G1-SFF:~/Escritorio/Servidor/miweb/projecte_rellotges
$ npm install
npm WARN deprecated popper.js@1.16.1: You can find the new Popper v2 at @popperjs/core, this package is dedicated to the legacy v1
npm WARN deprecated urix@0.1.0: Please see https://github.com/lydell/urix#deprecated
npm WARN deprecated resolve-url@0.2.1: https://github.com/lydell/resolve-url#deprecated
/home/adria/Escritorio/Servidor/miweb/projecte_rellotges
```

npm run dev

```
adria@adria-HP-ProDesk-600-G1-SFF:~/Escritorio/Servidor/miweb/projecte_rellotges
$ npm run dev

> @ dev /home/adria/Escritorio/Servidor/miweb/projecte_rellotges
> npm run development

> @ development /home/adria/Escritorio/Servidor/miweb/projecte_rellotges
> cross-env NODE_ENV=development node_modules/webpack/bin/webpack.js --progress
--hide-modules --config=node_modules/laravel-mix/setup/webpack.config.js

DONE Compiled successfully in 6154ms 12:01:31

    Asset      Size  Chunks             Chunk Names
/css/app.css  177 KiB  /js/app [emitted]  /js/app
  /js/app.js   1.5 MiB  /js/app [emitted]  /js/app
adria@adria-HP-ProDesk-600-G1-SFF:~/Escritorio/Servidor/miweb/projecte_rellotges
$
```

11- Creem un nou control·lador anomeat SettingsController:

php artisan make:controller SettingsController

```
adria@adria-HP-ProDesk-600-G1-SFF:~/Escritorio/Servidor/miweb/projecte_rellotges
$ php artisan make:controller SettingsController
Controller created successfully.
```


12-Anem al control·lador en qüestió i el deixem de la següent manera:

```
SettingsController.php X
home > adria > Escritorio > Servidor > miweb > projecte_rellotges > app > Http\Controllers

1  <?php
2
3      namespace App\Http\Controllers;
4
5      use Illuminate\Http\Request;
6
7      class SettingsController extends Controller
8      {
9          public function __construct()
10         {
11             $this->middleware('auth');
12         }
13
14         public function index()
15         {
16             return view('settings');
17         }
18     }
19
```

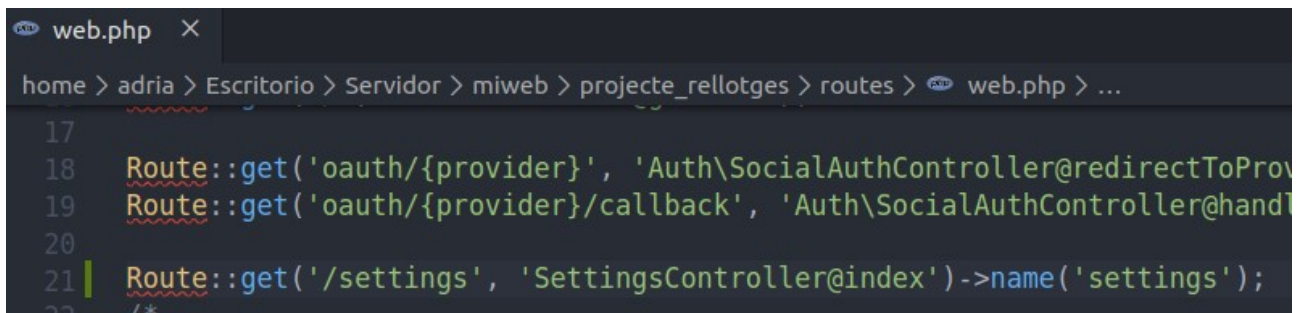
13-Creem una nova view anomenada settings i el deixem de la següent manera:

```
SettingsController.php  settings.blade.php X
me > adria > Escritorio > Servidor > miweb > projecte_rellotges > resources > views > settings.blade.php > ...

1  @extends('layouts.app')
2
3      @section('content')
4          <div class="container">
5              <div class="row">
6                  <div class="col-md-8 col-md-offset-2">
7                      <passport-authorized-clients></passport-authorized-clients>
8                      <passport-clients></passport-clients>
9                      <passport-personal-access-tokens></passport-personal-access-tokens>
10                 </div>
11             </div>
12         </div>
13     @endsection
14
```

14-Anem al document routes/web.php i hi afegim la següent ruta:

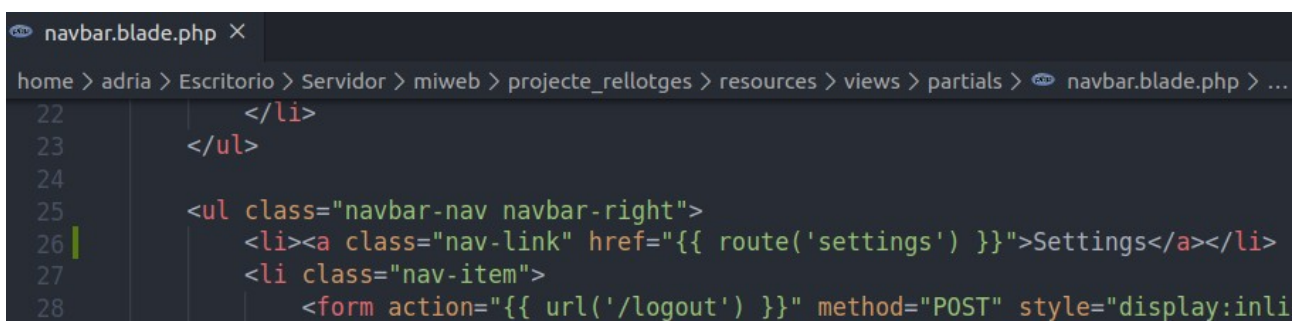
Route::get('/settings', 'SettingsController@index')->name('settings');



The screenshot shows a code editor with a file named 'web.php'. The breadcrumb navigation at the top reads: 'home > adria > Escritorio > Servidor > miweb > projecte_rellotges > routes > web.php > ...'. The code content shows lines 17 through 23. Line 21 contains the route definition: `Route::get('/settings', 'SettingsController@index')->name('settings');`. The cursor is positioned at the end of line 21.

15-Finalment anem a la view navbar i a la línia 26 li afegim el següent codi:

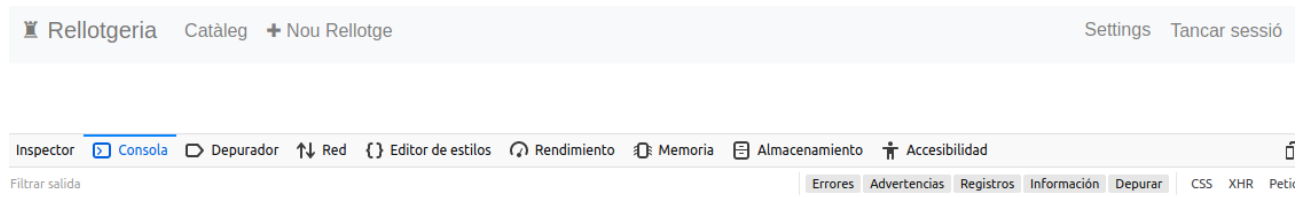
Settings



The screenshot shows a code editor with a file named 'navbar.blade.php'. The breadcrumb navigation at the top reads: 'home > adria > Escritorio > Servidor > miweb > projecte_rellotges > resources > views > partials > navbar.blade.php > ...'. The code content shows lines 22 through 28. Line 26 contains the HTML link: `Settings`. The cursor is positioned at the end of line 26.

16-Executa el serve, logejat amb algun usuari i vés a Settings:

NI DONA ERROR NI SURT RES!



Per a solucionar aquest problema, es fa el següent:

1) En la blade settings se li té que afegir al <div> que conté els <passport---> la id="app" i al final de la secció el <script> següent:

<script src="{{ asset('js/app.js') }}" ></script>

```
web.php  SettingsController.php  JS app.js  settings.blade.php X
home > adria > Escritorio > Servidor > miweb > projecte_rellotges > resources > views > settings.blade.php > ...
1  @extends('layouts.master')
2
3  @section('content')
4      <div class="container">
5          <div class="row">
6              <div class="col-md-8 col-md-offset-2" id="app">
7                  <passport-authorized-clients></passport-authorized-clients>
8                  <passport-clients></passport-clients>
9                  <passport-personal-access-tokens></passport-personal-access-tokens>
10             </div>
11         </div>
12     </div>
13     <script src="{{ asset('js/app.js') }}" ></script>
14 @endsection
```

2) En el document config/app.js afegim al final de cada require un `'default'`:

```
Vue.component(
  'passport-authorized-clients',
  require('./components/passport/AuthorizedClients.vue').default
);

Vue.component(
  'passport-clients',
  require('./components/passport/Clients.vue').default
);

Vue.component(
  'passport-personal-access-tokens',
  require('./components/passport/PersonalAccessTokens.vue').default
);
```

3) I lo més important, aplicar els canvis amb:

npm run dev

```
DONE Compiled successfully in 8117ms 19:49:02

  Asset      Size  Chunks             Chunk Names
/css/app.css 177 KiB /js/app [emitted]  /js/app
/js/app.js   1.5 MiB /js/app [emitted]  /js/app
adria@adria-HP-ProDesk-600-G1-SFF:~/Escritorio/Servidor/miweb/proyecto_rellotges
/resources/js$
```

4) I així ens funcionarà:

🏠 Rellotgeria

Catàleg

+ Nou Rellotge

Settings

Tancar sessió

OAuth Clients

Create New Client

You have not created any OAuth clients.

Personal Access Tokens

Create New Token

You have not created any personal access tokens.

17- Anem a crear un nou Oauth Client:

Create Client



Name

rellotgeria

Something your users will recognize and trust.

Redirect
URL

http://localhost:8000/callback

Your application's authorization callback URL.

Confidential



Require the client to authenticate with a secret. Confidential clients can hold credentials in a secure way without exposing them to unauthorized parties. Public applications, such as native desktop or JavaScript SPA applications, are unable to hold secrets securely.

Close

Create

OAuth Clients

[Create New Client](#)

Client ID	Name	Secret
-----------	------	--------

7

rellotgeria

WgxMOMWsaTXsd5Q0PfjvKOW6jMwhj2EfBhi9Cq8F

[Edit](#)[Delete](#)

18-A continuació, obrim el document routes/api.php i afegim lo següent:

```
Route::middleware('auth:api')->get('/todos', function (Request  
$request) {  
    return $request->user()->todos;  
});
```

A screenshot of a code editor window titled 'api.php'. The breadcrumb navigation shows the path: 'home > adria > Escritorio > Servidor > miweb > projecte_rellotges > routes > api.php > ...'. The code is as follows:

```
19  
20 Route::middleware('auth:api')->get('/todos', function (Request $request) {  
21     return $request->user()->todos;  
22 });|  
23
```

18-Començem per descarregar-nos el projecte del nostre company:

He volgut fer aquesta part de la pràctica amb el company de classe anomenat David, però que a causa de molts errors que ha tingut, tot i haver conjuntament arreglat molts dels seus problemes, no ha pogut arribar a aquest punt del exercici. Al no tenir amb qui fer les proves, no he tingut cap altre opció que parar aquí la pràctica.