

Acesso ao sistema administrativo <https://search-employee.herokuapp.com/>

Acesso ao web app <https://search-employee.herokuapp.com/>

Documentação de rotas

Route::resource('employees', 'EmployeeController');

Employees.index - Retorna uma view com todos os funcionários

Employees.create - Retorna a view de criação do funcionário

Employee.store - Salva um novo funcionário

Employee.show - Retorna a view de visualização do funcionário

Employee.edit - Retorna a view de edição de um determinado funcionário

Employee.update - Atualiza no banco o funcionário editado

Employee.destroy - Exclui um determinado funcionado

GET	<code>/employees</code>	index	employees.index
GET	<code>/employees/create</code>	create	employees.create
POST	<code>/employees</code>	store	employees.store
GET	<code>/employees/{id}</code>	show	employees.show
GET	<code>/employees/{id}/edit</code>	edit	employees.edit
PUT/PATCH	<code>/employees/{id}</code>	update	employees.update
DELETE	<code>/employees/{id}</code>	destroy	employees.destroy

Route::*resource*('experiences', 'ExperienceController');

Employees.index - Retorna uma view com todos as experiências

Employees.create - Retorna a view de criação da experiência

Employee.store - Salva uma nova experiência

Employee.show - Retorna a view de visualização da experiência

Employee.edit - Retorna a view de edição de uma determinada experiência

Employee.update - Atualiza no banco a experiências editada

Employee.destroy - Exclui uma determinada experiência

GET	<code>/experiences</code>	index	experiences.index
GET	<code>/experiences/create</code>	create	experiences.create
POST	<code>/experiences</code>	store	experiences.store
GET	<code>/experiences/{id}</code>	show	experiences.show
GET	<code>/experiences/{id}/edit</code>	edit	experiences.edit
PUT/PATCH	<code>/experiences/{id}</code>	update	experiences.update
DELETE	<code>/experiences/{id}</code>	destroy	experiences.destroy

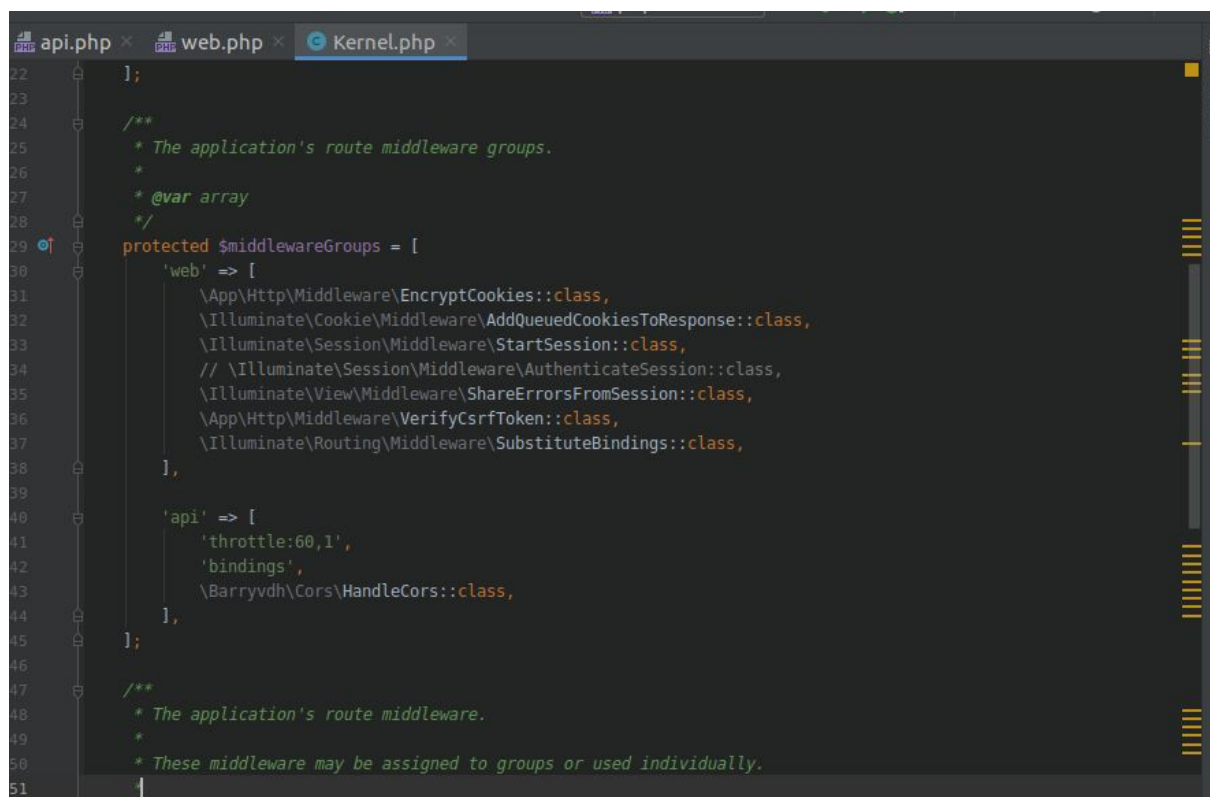
Route::*get*('modal', 'Controller@modal')->name('modal.delete'); - Retorna uma modal de confirmação de exclusão

Route::*get*('search', 'EmployeeController@search')->name('employees.search');

Retorna os funcionários pesquisados

Middleware web - Para uma solicitação da Web, você pode acessar o site e o aplicativo preservará esse estado de logon entre as rotas que você visitar durante a sessão. A API pode ter um middleware de aceleração, enquanto o próprio aplicativo da Web exigirá tokens CSRF, cookies (criptografados), sessões. Agora, você pode aplicar cada um deles independentemente um do outro com facilidade - aplicando um middleware de grupo da API às suas rotas de API e um middleware de grupo da Web às suas rotas da Web. Se você tiver rotas públicas, poderá acelerar o tempo de execução dessas páginas deixando de fora cookies e sessões por completo.

Middleware web definindo quais são os controles do sistema



```
1;
23
24 /**
25  * The application's route middleware groups.
26  *
27  * @var array
28  */
29 protected $middlewareGroups = [
30     'web' => [
31         \App\Http\Middleware\EncryptCookies::class,
32         \Illuminate\Cookie\Middleware\AddQueuedCookiesToResponse::class,
33         \Illuminate\Session\Middleware\StartSession::class,
34         // \Illuminate\Session\Middleware\AuthenticateSession::class,
35         \Illuminate\View\Middleware\ShareErrorsFromSession::class,
36         \App\Http\Middleware\VerifyCsrfToken::class,
37         \Illuminate\Routing\Middleware\SubstituteBindings::class,
38     ],
39
40     'api' => [
41         'throttle:60,1',
42         'bindings',
43         \Barryvdh\Cors\HandleCors::class,
44     ],
45 ];
46
47 /**
48  * The application's route middleware.
49  *
50  * These middleware may be assigned to groups or used individually.
51  */
```

Middleware groups may be assigned to routes and controller actions using the same syntax as individual middleware. Again, middleware groups make it more convenient to assign many middleware to a route at once:

```
Route::get('/', function () {  
    //  
})->middleware('web');  
  
Route::group(['middleware' => ['web']], function () {  
    //  
});
```



Out of the box, the `web` middleware group is automatically applied to your `routes/web.php` file by the `RouteServiceProvider`.

```
1  <?php  
2  
3  /* ... */  
13  
14  Route::get('/', function () {  
15      return redirect( to: '/employees');  
16  });  
17  
18  Route::resource('employees', 'EmployeeController');  
19  Route::resource('experiences', 'ExperienceController');  
20  Route::get('modal', 'Controller@modal')->name('modal.delete');  
21  Route::get('search', 'EmployeeController@search')->name('employees.search');  
22
```

Exemplo de requisição para api

```
script>
$(document).ready(function() {
  $.ajax({
    url: "https://employees-crud.herokuapp.com/api/employee/show/"+ "<?php print_r($_GET['data']);?>",
    contentType: 'application/json',
    method: 'get',
    success: function (result) {
      $('#name').append(result.name);
      $('#phone').append(result.phone);
      $('#email').append(result.email);
      $('#locale').append(""+result.address.city+"/"+result.address.state);

      console.log(result)

      for(let i = 0; i < result.experiences.length; i++) {
        $('#list').append('<li>'+result.experiences[i].description+' - Começou em '+result.experiences[i].starte
      }
    }
  });
});
```