В данной системе будет использоваться следующий **алгоритм разделения прав для dashboard**:

-пользователь с ролью **admin** может делать **все действия** со всеми записями (**create, store, edit, update, destroy**);

-пользователь с ролью **redac** может делать **только корректировку** любой записи (**edit, update**);

-все остальные пользователи (роль **user**) **не могут** работать с **dashboard**.

1.**Удалите** из БД таблицу **users** + из таблицы **migrations** запись ...\_create\_**users**\_table.

2.**Добавьте** в миграцию **...\_create\_users\_table.php** поле:

**$table->enum('role', array('user', 'redac', 'admin'))->default('user');**

3.Измените **database/seeds/DatabaseSeeder.php**, добавив пользователям **роль**:

...

**use App\Models\User;**

...

/\*\*

\* Seed the application's database.

\*

\* @return void

\*/

public function run()

{

// $this->call(UsersTableSeeder::class);

// Users

User::create(

[

**'name' => 'admin',**

'email' => 'admin@gmail.com',

'password' => bcrypt('**12345678**'),

'**role' => 'admin',**

'remember\_token' => str\_random(10),

]

);

User::create(

[

'**name' => 'alex',**

'email' => 'alex@gmail.com',

'password' => bcrypt('**12345678**'),

**'role' => 'redac',**

'remember\_token' => str\_random(10),

]

);

User::create(

[

**'name' => 'serg',**

'email' => 'serg@gmail.com',

'password' => bcrypt('**12345678**'),

**'role' => 'user',**

'remember\_token' => str\_random(10),

]

);

User::create(

[

**'name' => 'helen',**

'email' => 'helen@gmail.com',

'password' => bcrypt('**12345678**'),

**'role' => 'user',**

'remember\_token' => str\_random(10),

]

);

}

+сделайте **в консоле php artisan migrate --seed**

4.Для **custom-login and registration form** скопировать

**resourse\views\auth\login.blade.php + … register.blade.php** из проекта **laravel-cards**

**+php artisan view:clear**

5.Изменить **app\Http\Controllers\Auth\LoginController.php, RegisterController.php**

protected $redirectTo = **'/'**;

6.Для отображения пункта меню **Dashboard** только для пользователей с ролями **admin, redac** добавить в resources\views\**front\navbar.blade.php**

**@admin**

<a class="dropdown-item" href="#">**Dashboard**</a>

**@endadmin**

**@redac**

<a class="dropdown-item" href="#">**Dashboard**</a>

**@endredac**

+ в app\Providers\**AppServiceProvider.php**

...

**use Illuminate\Support\Facades\Blade;**

...

public function boot()

{

view()->composer('front/index', CountryComposer::class);

**Blade::if('admin', function () {**

**return auth()->user()->role === 'admin';**

**});**

**Blade::if('redac', function () {**

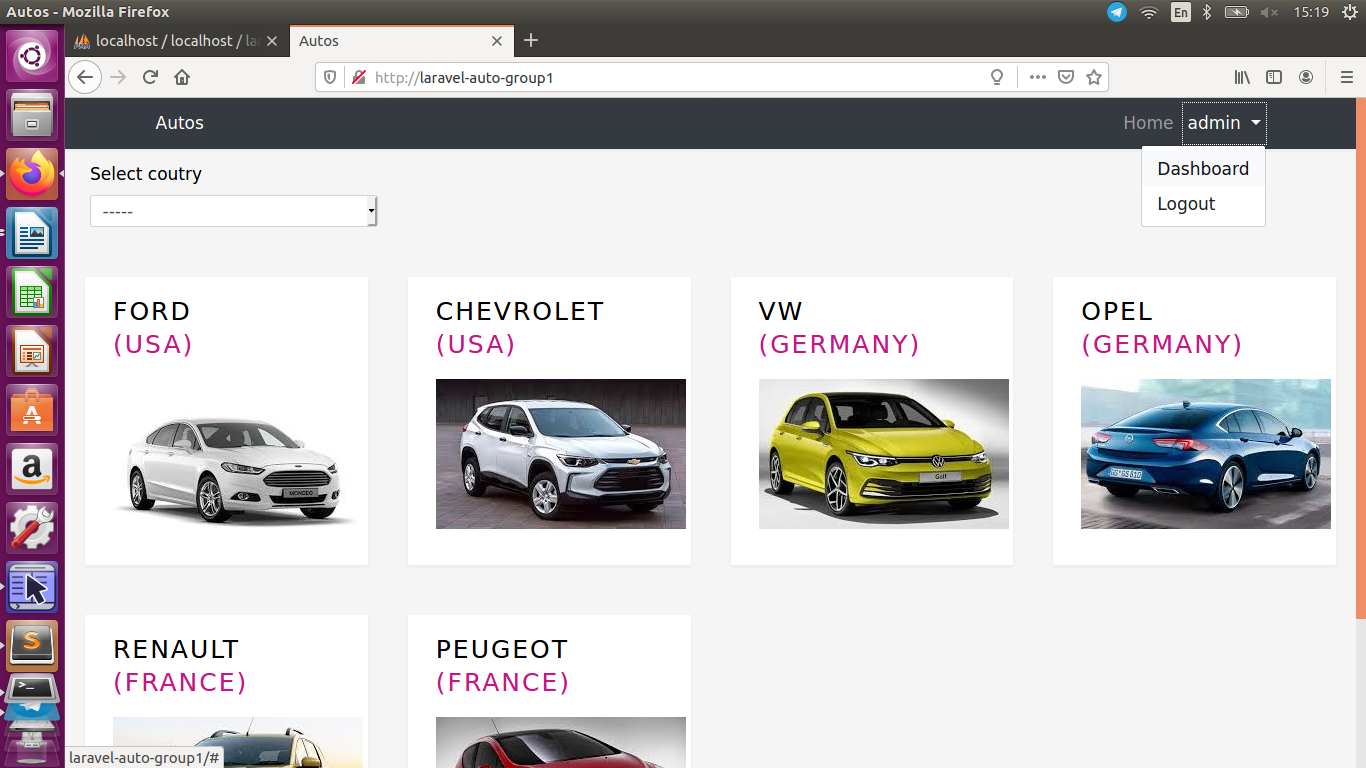
**return auth()->user()->role === 'redac';**

**});**

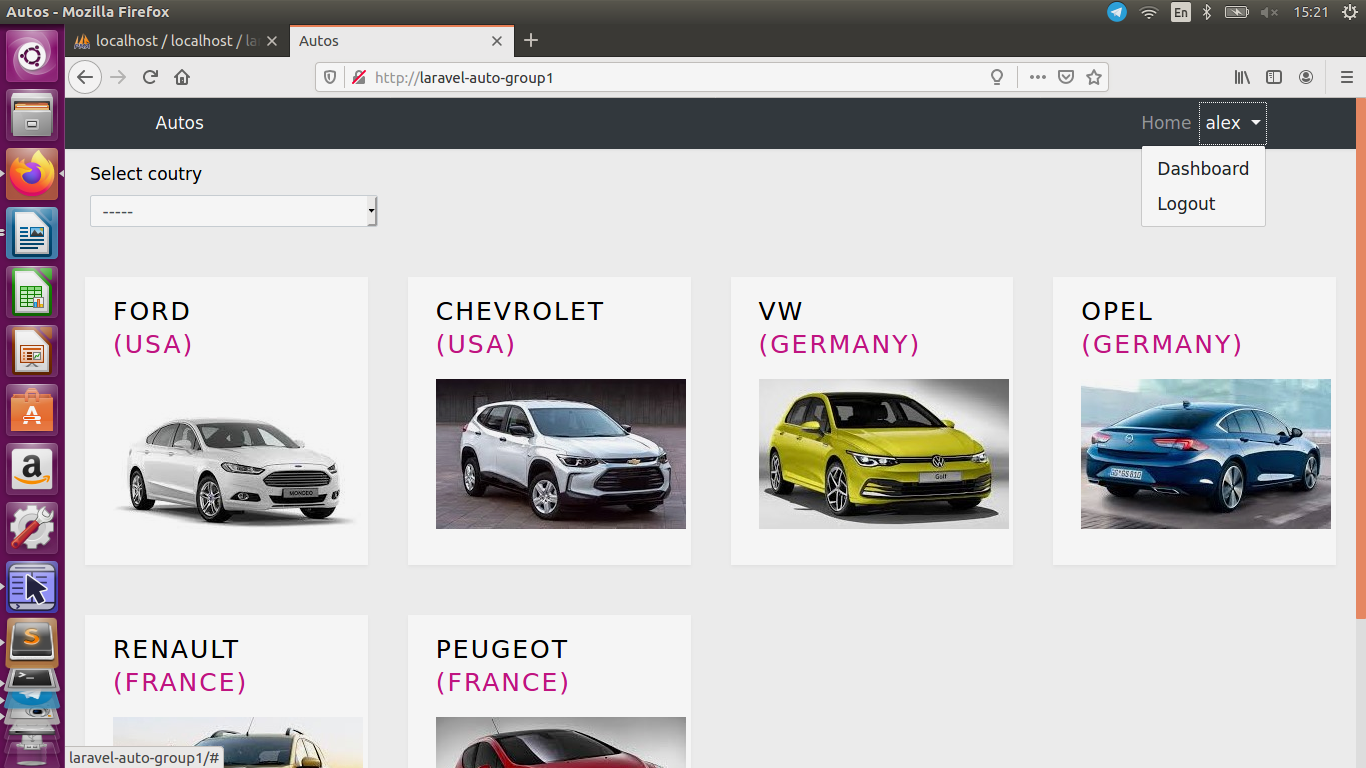
}

**+php artisan view:clear**

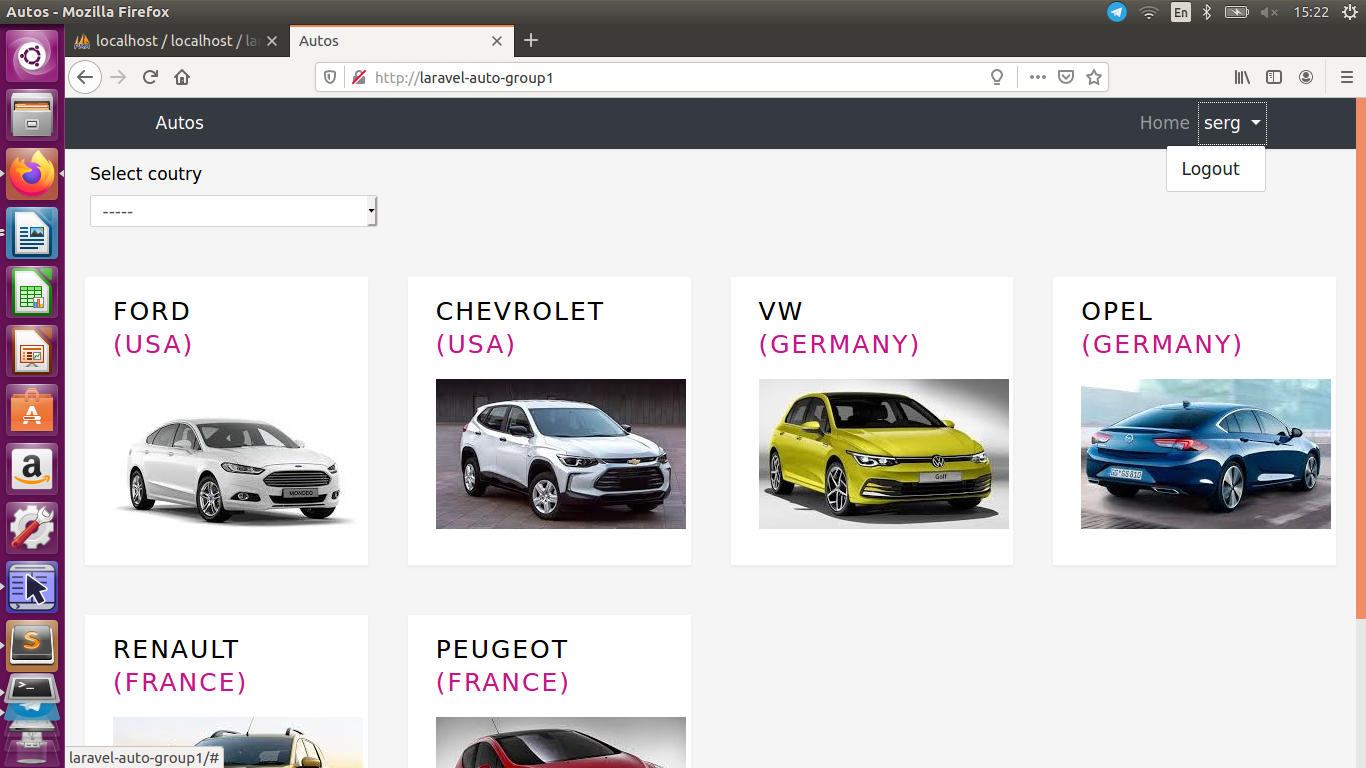
-пользователь с ролью **admin**:

****

-пользователь с ролью **redac**:

****

-пользователь с рольюобычный **user**:

****

7.Для отображения **данных в dashboard**:

-в resources\views\**front\navbar.blade.php**

@admin

<a class="dropdown-item" href="**{{ route('dashboard') }}**">Dashboard</a>

@endadmin

@redac

<a class="dropdown-item" href="**{{ route('dashboard') }}**">Dashboard</a>

@endredac

-в **routes\web.php**

...

/\*

|--------------------------------------------------------------------------

| Backend

|--------------------------------------------------------------------------|

\*/

**Route::prefix('')->namespace('Back')->group(function () {**

**Route::name('dashboard')->get('/dashboard', 'AdminController@index');**

**});**

-создать app\Http\Controllers\**Back\AdminController.php** (возможно **копирование из AutoConproller.php + измененить (ниже)**):

<?php

namespace App\Http\Controllers\**Back**;

use Illuminate\Http\Request;

use App\ {

Http\Controllers\Controller,

Repositories\AdminRepository,

Http\Controllers\Traits\Indexable

};

class **Admin**Controller extends Controller

{

use Indexable;

/\*\*

\* Create a new controller instance.

\*

\* @return void

\*/

public function \_\_construct(**Admin**Repository $repository)

{

//$this->middleware('auth');

$this->repository = $repository;

$this->namespace = '**back**';

}

}

-создать app\Http\**Repositories\AdminRepository.php** (возможно **копирование из AutoRepository.php + измененить (ниже)**):

<?php

namespace App\Repositories;

use App\Models\ {

Auto

};

class **Admin**Repository

{

/\*\*

\* The Model instance.

\*

\* @var \Illuminate\Database\Eloquent\Model

\*/

protected $model;

/\*\*

\* Create a new AutoRepository instance.

\*

\* @param \App\Models\Auto $auto

\*/

public function \_\_construct(Auto $auto)

{

$this->model= $auto;

}

/\*\*

\* Create a query for Autos.

\*

\* @return \Illuminate\Database\Eloquent\Builder

\*/

public function getData($request)

{

$query = $this->model

->select('id', 'country\_id', 'name', 'image')**;**

**//**->orderBy($parameters['order'], $parameters['direction']);

**//**if($request->country) $query->where('country\_id', $request->country);

return $query->get();

}

}

-создать тестовую resources\views\**back\index.blade.php** :

**@php**

**print\_r($autos);**

**@endphp**

