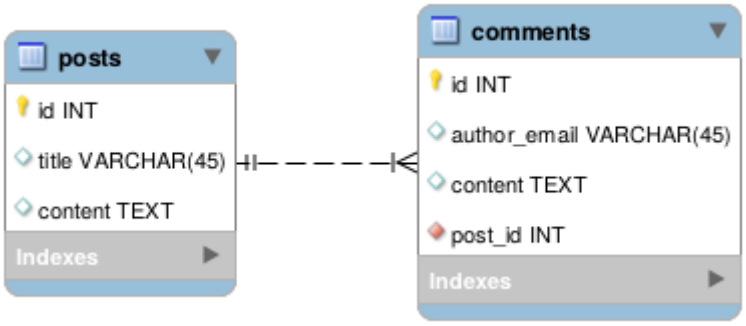
LARAVEL 5.6
USP dev (Leandro Ramos)

https://www.youtube.com/watch?v=lO_udEy9I4U&list=PLIFOx3X8xDut2X2ydevlVvFKkqFxZ4SX9

Resumo do curso feito por Roberto Pinheiro

Aula 01 - hasMany Relationships e Tinker

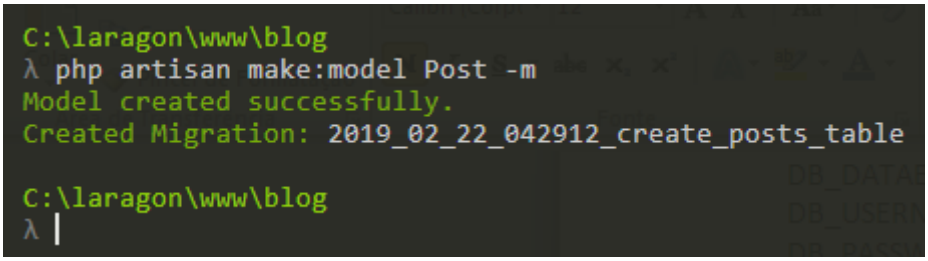


Configurando o acesso ao banco de dados

DB_CONNECTION=mysql
DB_HOST=localhost
DB_PORT=3306
DB_DATABASE=blog
DB_USERNAME=root
DB_PASSWORD=

Criando os models e migrations

php artisan make:model Post -m



database/migrations/2019_02_22_042912_create_posts_table.php

```php
<?php

use Illuminate\Support\Facades\Schema;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Database\Migrations\Migration;

class CreatePostsTable extends Migration
{
    public function up()
    {
        Schema::create('posts', function (Blueprint $table) {
            $table->increments('id');
            $table->string('title');
            $table->text('content');
            $table->timestamps();
        });
    }

    public function down()
    {
        Schema::dropIfExists('posts');
    }
}
```

php artisan make:model Comment -m

```
C:\laragon\www\blog
λ php artisan make:model Comment -m
Model created successfully.
Created Migration: 2019_02_22_043021_create_comments_table

C:\laragon\www\blog
λ
```

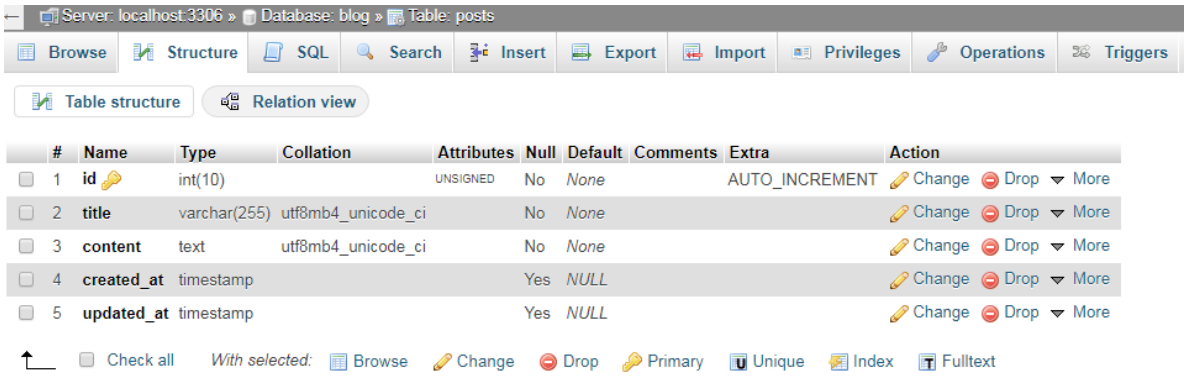
database/migrations/2019_02_22_043021_create_comments_table.php

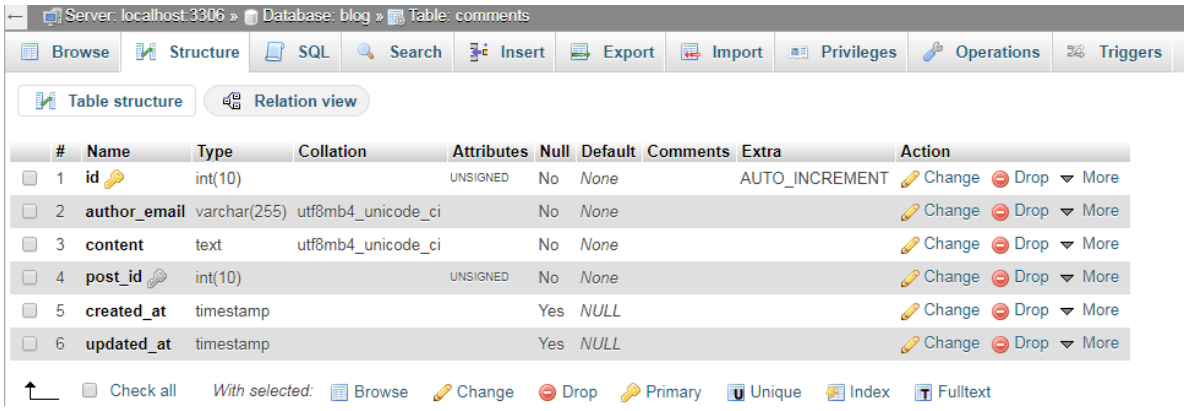
```
<?php

use Illuminate\Support\Facades\Schema;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Database\Migrations\Migration;

class CreateCommentsTable extends Migration
{
    public function up()
    {
        Schema::create('comments', function (Blueprint $table) {
            $table->increments('id');
            $table->string('author_email');
            $table->text('content');
            $table->integer('post_id')->unsigned();
            $table->timestamps();

            $table->foreign('post_id')->references('id')->on('posts');
        });
    }

    public function down()
    {
        Schema::dropIfExists('comments');
    }
}
```

Criando as tabelas

php artisan migrate

```
C:\laragon\www\blog
λ php artisan migrate
Migrating: 2014_10_12_000000_create_users_table
Migrated:  2014_10_12_000000_create_users_table
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated:  2014_10_12_100000_create_password_resets_table
Migrating: 2019_02_22_042912_create_posts_table
Migrated:  2019_02_22_042912_create_posts_table
Migrating: 2019_02_22_043021_create_comments_table
Migrated:  2019_02_22_043021_create_comments_table

C:\laragon\www\blog
λ
```





app/Post.php

```php
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Post extends Model
{
   public function comments()
   {
      // Um post tem vários comentários
      return $this->hasMany('App\Comment');
   }
}
```

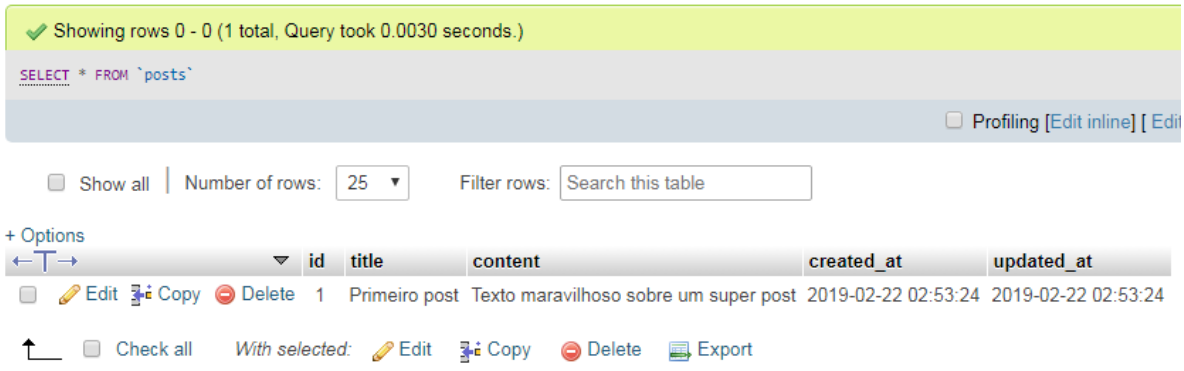
app/Comment.php

```php
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Comment extends Model
{
   public function post()
   {
      // Um comentário pertence a um post
      return $this->belongsTo('App\Post');
   }
}
```

Testando os relacionamentos com o Tinker

php artisan tinker

```
C:\laragon\www\blog
λ php artisan tinker
Psy Shell v0.9.6 (PHP 7.2.11 — cli) by Justin Hileman
>>> $post = new Post
[!] Aliasing 'Post' to 'App\Post' for this Tinker session.
=> App\Post {#2865}
>>> $post->title = "Primeiro post"
=> "Primeiro post"
>>> $post->content = "Texto maravilhoso sobre um super post"
=> "Texto maravilhoso sobre um super post"
>>> $post->save()
=> true
>>> Post::all()
=> Illuminate\Database\Eloquent\Collection {#2874
     all: [
       App\Post {#2875
         id: 1,
         title: "Primeiro post",
         content: "Texto maravilhoso sobre um super post",
         created_at: "2019-02-22 02:53:24",
         updated_at: "2019-02-22 02:53:24",
       },
     ],
   }
>>>
```



```
>>> $comment = new Comment
[!] Aliasing 'Comment' to 'App\Comment' for this Tinker session.
=> App\Comment {#2872}
>>> $comment->author_email = "fulano@eu.com"
=> "fulano@eu.com"
>>> $comment->content = "Gostei muito do seu post"
=> "Gostei muito do seu post"
>>> $comment->post_id = 1
=> 1
>>> $comment->save()
=> true
>>> $comment2 = new Comment
=> App\Comment {#2869}
>>> $comment2->author_email = "ciclano@eu.com"
=> "ciclano@eu.com"
>>> $comment2->content = "Eu não gostei"
=> "Eu não gostei"
>>> $comment2->post_id = 1
=> 1
>>> $comment2->save()
=> true
```

```
>>> Comment::all()
=> Illuminate\Database\Eloquent\Collection {#2880
     all: [
       App\Comment {#2881
         id: 1,
         author_email: "fulano@eu.com",
         content: "Gostei muito do seu post",
         post_id: 1,
         created_at: "2019-02-22 02:56:20",
         updated_at: "2019-02-22 02:56:20",
       },
       App\Comment {#2882
         id: 2,
         author_email: "ciclano@eu.com",
         content: "Eu não gostei",
         post_id: 1,
         created_at: "2019-02-22 02:57:56",
         updated_at: "2019-02-22 02:57:56",
       },
     ],
   }
>>>
```

Showing rows 0 - 1 (2 total, Query took 0.0020 seconds.)

```
SELECT * FROM `comments`
```

Profiling [Edit inline] [Edit] [Explain SQL]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

+ Options

	id	author_email	content	post_id	created_at	updated_at
Edit Copy Delete	1	fulano@eu.com	Gostei muito do seu post	1	2019-02-22 02:56:20	2019-02-22 02:56:20
Edit Copy Delete	2	ciclano@eu.com	Eu não gostei	1	2019-02-22 02:57:56	2019-02-22 02:57:56

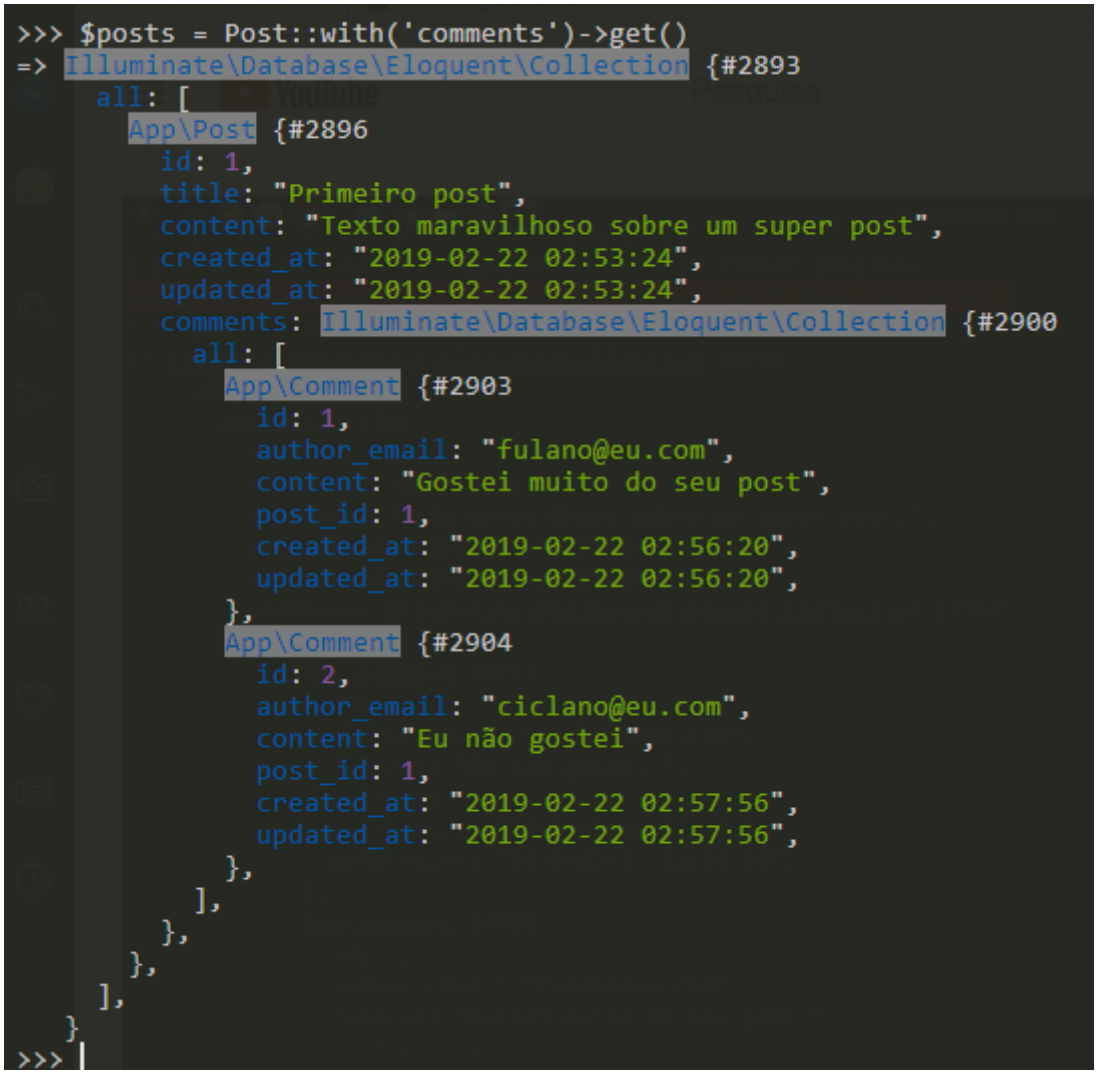
Check all With selected: Edit Copy Delete Export

```
>>> Post::first()
=> App\Post {#2873
     id: 1,
     title: "Primeiro post",
     content: "Texto maravilhoso sobre um super post",
     created_at: "2019-02-22 02:53:24",
     updated_at: "2019-02-22 02:53:24",
   }
>>> $post->comments
=> Illuminate\Database\Eloquent\Collection {#2870
     all: [
       App\Comment {#2875
         id: 1,
         author_email: "fulano@eu.com",
         content: "Gostei muito do seu post",
         post_id: 1,
         created_at: "2019-02-22 02:56:20",
         updated_at: "2019-02-22 02:56:20",
       },
       App\Comment {#2859
         id: 2,
         author_email: "ciclano@eu.com",
         content: "Eu não gostei",
         post_id: 1,
         created_at: "2019-02-22 02:57:56",
         updated_at: "2019-02-22 02:57:56",
       },
     ],
   }
>>>
```

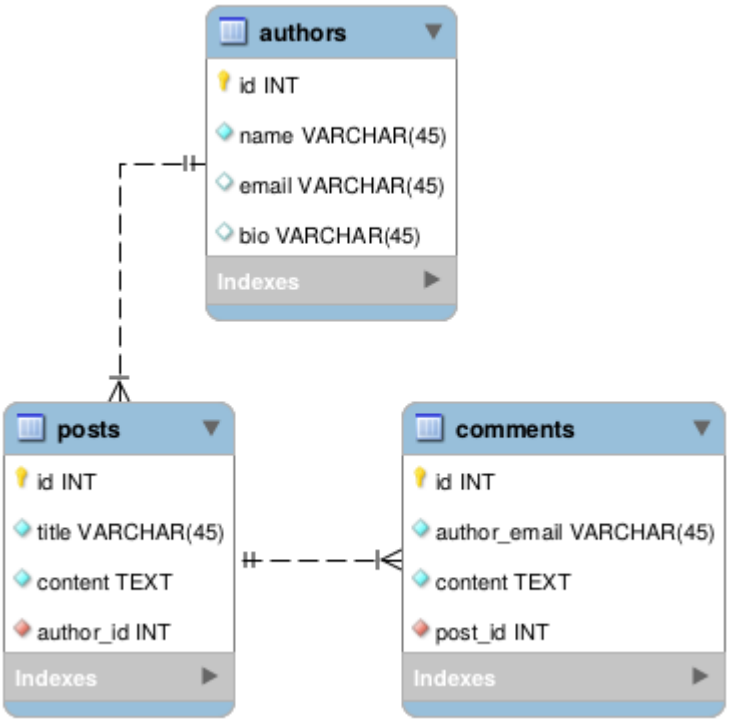
```
>>> $comment = Comment::find(2)
=> App\Comment {#2877
     id: 2,
     author_email: "ciclano@eu.com",
     content: "Eu não gostei",
     post_id: 1,
     created_at: "2019-02-22 02:57:56",
     updated_at: "2019-02-22 02:57:56",
   }
>>> $comment -> post
=> App\Post {#2873
     id: 1,
     title: "Primeiro post",
     content: "Texto maravilhoso sobre um super post",
     created_at: "2019-02-22 02:53:24",
     updated_at: "2019-02-22 02:53:24",
   }
>>>
```

```
>>> $comment -> post -> title
=> "Primeiro post"
>>> $post
=> App\Post {#2865
     title: "Primeiro post",
     content: "Texto maravilhoso sobre um super post",
     updated_at: "2019-02-22 02:53:24",
     created_at: "2019-02-22 02:53:24",
     id: 1,
     comments: Illuminate\Database\Eloquent\Collection {#2870
       all: [
         App\Comment {#2875
           id: 1,
           author_email: "fulano@eu.com",
           content: "Gostei muito do seu post",
           post_id: 1,
           created_at: "2019-02-22 02:56:20",
           updated_at: "2019-02-22 02:56:20",
         },
         App\Comment {#2859
           id: 2,
           author_email: "ciclano@eu.com",
           content: "Eu não gostei",
           post_id: 1,
           created_at: "2019-02-22 02:57:56",
           updated_at: "2019-02-22 02:57:56",
         },
       ],
     },
   }
>>>
```

```
>>> $post->comments->where('id',2)
=> Illuminate\Database\Eloquent\Collection {#2881
     all: [
       1 => App\Comment {#2859
         id: 2,
         author_email: "ciclano@eu.com",
         content: "Eu não gostei",
         post_id: 1,
         created_at: "2019-02-22 02:57:56",
         updated_at: "2019-02-22 02:57:56",
       },
     ],
   }
>>>
```


>>> $posts = Post::with('comments')->get()
=> Illuminate\Database\Eloquent\Collection {#2893
all: [
App\Post {#2896
id: 1,
title: "Primeiro post",
content: "Texto maravilhoso sobre um super post",
created_at: "2019-02-22 02:53:24",
updated_at: "2019-02-22 02:53:24",
comments: Illuminate\Database\Eloquent\Collection {#2900
all: [
App\Comment {#2903
id: 1,
author_email: "fulano@eu.com",
content: "Gostei muito do seu post",
post_id: 1,
created_at: "2019-02-22 02:56:20",
updated_at: "2019-02-22 02:56:20",
},
App\Comment {#2904
id: 2,
author_email: "ciclano@eu.com",
content: "Eu não gostei",
post_id: 1,
created_at: "2019-02-22 02:57:56",
updated_at: "2019-02-22 02:57:56",
},
],
},
},
],
}
>>>

Aula 02 - hasManyThrough Relationships



Criando o model para os autores

php artisan make:model Author -m

```
C:\laragon\www\blog
λ php artisan make:model Author -m
Model created successfully.
Created Migration: 2019_02_22_044010_create_authors_table

C:\laragon\www\blog
λ
```

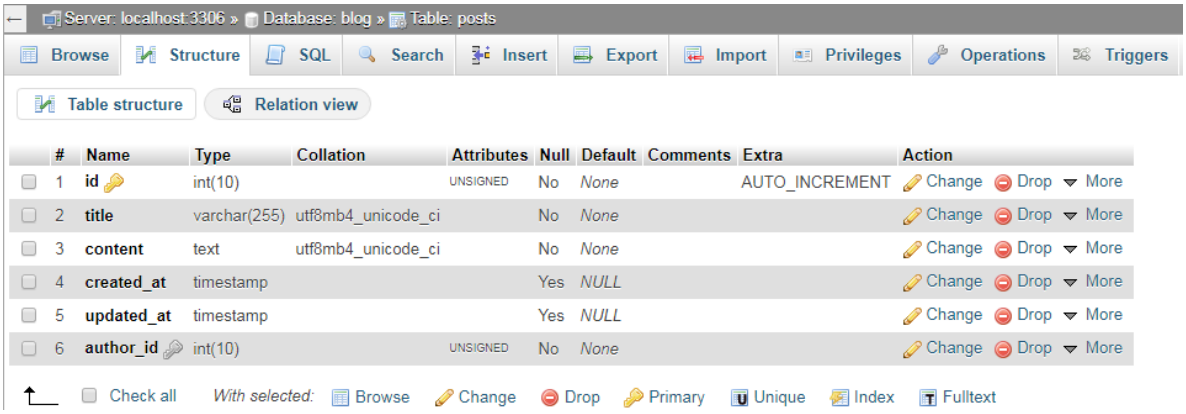
database/migrations/2019_02_22_044010_create_authors_table.php

```
<?php

use Illuminate\Support\Facades\Schema;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Database\Migrations\Migration;

class CreateAuthorsTable extends Migration
{
    public function up()
    {
        Schema::create('authors', function (Blueprint $table) {
            $table->increments('id');
            $table->string('name');
            $table->string('email');
            $table->string('bio')->nullable();
            $table->timestamps();
        });
    }

    public function down()
    {
        Schema::dropIfExists('authors');
    }
}
```

Criando uma migration para adicionar um campo na tabela posts

php artisan make:migration add_author_id_to_posts --table=posts

```
C:\laragon\www\blog
λ php artisan make:migration add_author_id_to_posts --table=posts
Created Migration: 2019_02_22_044234_add_author_id_to_posts
```



database/migrations/2019_02_22_044234_add_author_id_to_posts.php

```
<?php

use Illuminate\Support\Facades\Schema;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Database\Migrations\Migration;

class AddAuthorIdToPosts extends Migration
{
    public function up()
    {
        Schema::table('posts', function (Blueprint $table) {
            $table->integer('author_id')->unsigned();
            $table->foreign('author_id')->references('id')->on('authors');
        });
    }

    public function down()
    {
        Schema::table('posts', function (Blueprint $table) {
            $table->dropForeign('posts_author_id_foreign');
        });
    }
}
```

Criando as tabelas

php artisan migrate

```
C:\laragon\www\blog
λ php artisan migrate
Migrating: 2019_02_22_044010_create_authors_table
Migrated:  2019_02_22_044010_create_authors_table
Migrating: 2019_02_22_044234_add_author_id_to_posts
Migrated:  2019_02_22_044234_add_author_id_to_posts
```

app/Post.php

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Post extends Model
{
   public function comments()
   {
      // Um post tem vários comentários
      return $this->hasMany('App\Comment');
   }

   public function author()
   {
      // Um post pertence a um autor
      return $this->belongsTo('App\Author');
   }
}
```

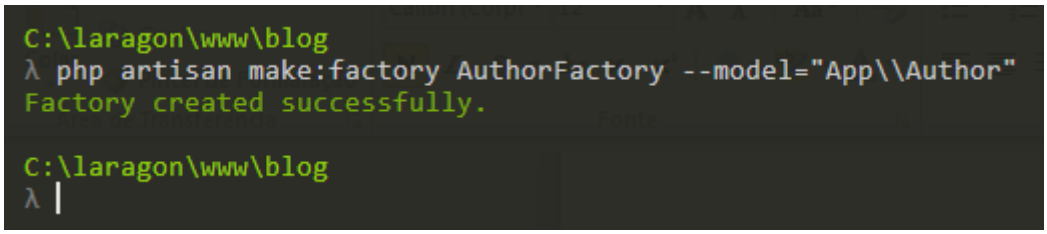
app/Author.php

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Author extends Model
{
   public function posts()
   {
         // Um autor tem muitos posts
      return $this->hasMany('App\Post');
   }
   public function comments()
   {
         // Um autor tem muitos comentários, passando pelos posts
      return $this->hasManyThrough('App\Comment', 'App\Post');
   }
}
```

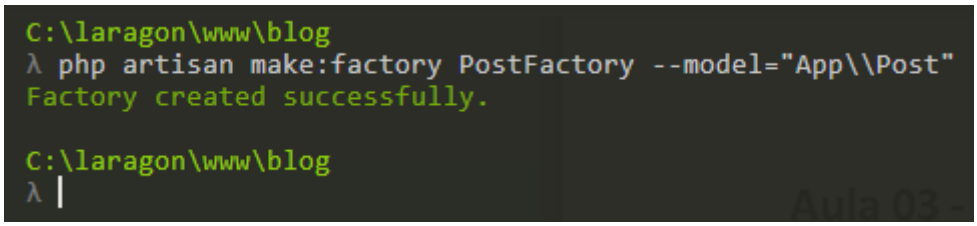
Aula 03 - Factories

Criando os factories

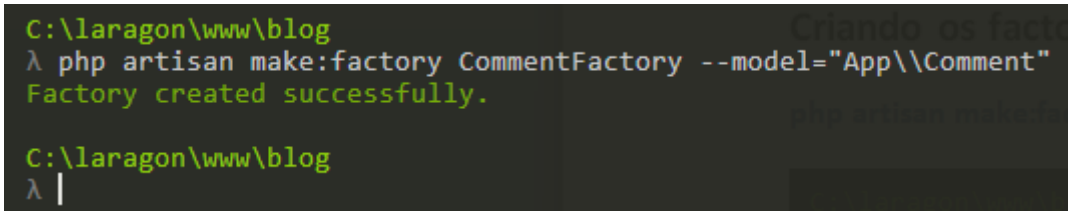
php artisan make:factory AuthorFactory --model="App\\Author"



php artisan make:factory PostFactory --model="App\\Post"



php artisan make:factory CommentFactory --model="App\\Comment"



database/factory/AuthorFactory.php

```
<?php

use Faker\Generator as Faker;

$factory->define(App\Author::class, function (Faker $faker) {
   return [
      'name'  => $faker->name,
      'email' => $faker->unique()->safeEmail,
      'bio'   => $faker->paragraph(1),
   ];
});
```

database/factory/PostFactory.php

```
<?php

use App\Author;
use Faker\Generator as Faker;

$factory->define(App\Post::class, function (Faker $faker) {
   return [
      'title'     => $faker->sentence(4),
      'content'   => $faker->paragraph(4),

      'author_id' => function () {
         return Author::orderByRaw("RAND()")
            ->take(1)
            ->first()
            ->id;
      }
   ];
});
```

database/factory/CommentFactory.php

```
<?php

use App\Post;
use Faker\Generator as Faker;

$factory->define(App\Comment::class, function (Faker $faker) {
   return [
      'author_email'  => $faker->unique()->safeEmail,
      'content'       => $faker->paragraph(2),

      'post_id' => function () {
         return Post::orderByRaw("RAND()")
            ->take(1)
            ->first()
            ->id;
      }
   ];
});
```

database/seeds/DatabaseSeeder.php

```
<?php

use Illuminate\Database\Seeder;

class DatabaseSeeder extends Seeder
{
   public function run()
   {
      // $this->call(UsersTableSeeder::class);
      echo "Creating 10 authors...\n";
      factory(App\Author::class, 10)->create();

      echo "Creating 36 posts related to random authors...\n";
      factory(App\Post::class, 36)->create();

      echo "Creating 67 comments related to random posts...\n";
      factory(App\Comment::class, 67)->create();
   }
}
```

Populando as tabelas

drop database blog;
create database blog;

No terminal:

php artisan migrate

```
C:\laragon\www\blog
λ php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated:  2014_10_12_000000_create_users_table
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated:  2014_10_12_100000_create_password_resets_table
Migrating: 2019_02_22_042912_create_posts_table
Migrated:  2019_02_22_042912_create_posts_table
Migrating: 2019_02_22_043021_create_comments_table
Migrated:  2019_02_22_043021_create_comments_table
Migrating: 2019_02_22_044010_create_authors_table
Migrated:  2019_02_22_044010_create_authors_table
Migrating: 2019_02_22_044234_add_author_id_to_posts
Migrated:  2019_02_22_044234_add_author_id_to_posts

C:\laragon\www\blog
λ
```

php artisan db:seed

```
C:\laragon\www\blog
λ php artisan db:seed
Creating 10 authors...
Creating 36 posts related to random authors...
Creating 67 comments related to random posts...

C:\laragon\www\blog
λ
```

Showing rows 0 - 9 (10 total, Query took 0.0020 seconds.)

```
SELECT * FROM `authors`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

+ Options

	id	name	email	bio	created_at	updated_at
Edit Copy Delete	1	Keyon Block	roxane85@example.org	Vero cupiditate perferendis id.	2019-02-22 15:34:46	2019-02-22 15:34:46
Edit Copy Delete	2	Lyla Zemlak	laila65@example.net	Labore qui quod magnam aut aut repellat pariatur.	2019-02-22 15:34:46	2019-02-22 15:34:46
Edit Copy Delete	3	Prof. Adrien Hand Sr.	littel.reyes@example.net	Optio ipsam minima sapiente fugit voluptatem quam.	2019-02-22 15:34:46	2019-02-22 15:34:46
Edit Copy Delete	4	Bertha Feeney II	isai73@example.net	Reiciendis accusantium aut facilis molestiae.	2019-02-22 15:34:46	2019-02-22 15:34:46
Edit Copy Delete	5	Mr. Nels Heller	davis.cedrick@example.com	Quibusdam quis sunt repellendus dolor reiciendis d...	2019-02-22 15:34:46	2019-02-22 15:34:46
Edit Copy Delete	6	Reynold Osinski	dasia75@example.net	Magni excepturi harum quod vel voluptatem nihil qu...	2019-02-22 15:34:46	2019-02-22 15:34:46
Edit Copy Delete	7	Lysanne Feest	laron56@example.com	Debitis eos aut eligendi fugiat adipisci rerum ill...	2019-02-22 15:34:46	2019-02-22 15:34:46
Edit Copy Delete	8	Chris Rath Sr.	reynolds.rafaela@example.org	Non dolore aspernatur quae aut eligendi veniam. Cu...	2019-02-22 15:34:46	2019-02-22 15:34:46
Edit Copy Delete	9	Aiyana Leuschke II	isobel76@example.com	Accusamus minus occaecati mollitia sed voluptatum ...	2019-02-22 15:34:46	2019-02-22 15:34:46
Edit Copy Delete	10	Prof. Margarette Nader	darrin06@example.org	Iure est ut officiis culpa esse. Dolorum et deseru...	2019-02-22 15:34:46	2019-02-22 15:34:46

Check all With selected: Edit Copy Delete Export

Showing rows 0 - 35 (36 total, Query took 0.0010 seconds.)

```
SELECT * FROM `posts`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 50 Filter rows: Search this table Sort by key: None

+ Options

	id	title	content	created_at	updated_at	author_id
Edit Copy Delete	1	Eligendi explicabo eos animi et.	Voluptatem soluta tenetur officiis mollitia doloru...	2019-02-22 15:34:46	2019-02-22 15:34:46	4
Edit Copy Delete	2	Harum corporis soluta inventore.	Voluptatem dolores quaerat totam ipsam. Molestiae ...	2019-02-22 15:34:46	2019-02-22 15:34:46	6
Edit Copy Delete	3	Explicabo qui quibusdam.	Enim voluptatem earum corporis saepe aperiam. Anim...	2019-02-22 15:34:46	2019-02-22 15:34:46	3
Edit Copy Delete	4	Dolor quod fugit.	Harum vel nihil amet saepe architecto dolore simil...	2019-02-22 15:34:46	2019-02-22 15:34:46	7
Edit Copy Delete	5	Voluptas id ipsum nam id fugit.	Dolor est et autem blanditiis. Molestiae quibusdam...	2019-02-22 15:34:46	2019-02-22 15:34:46	1
Edit Copy Delete	6	Sit quisquam qui et.	Et enim maiores magnam rem. Est at cumque sint aut...	2019-02-22 15:34:46	2019-02-22 15:34:46	8
Edit Copy Delete	7	Tempora qui ut consequatur dignissimos possimus.	Repellendus accusamus deleniti consequatur quas. F...	2019-02-22 15:34:46	2019-02-22 15:34:46	9
Edit Copy Delete	8	Laboriosam officia magni dolores eum fugiat.	Tempora beatae facilis distinctio aut. Ut accusamu...	2019-02-22 15:34:46	2019-02-22 15:34:46	9
Edit Copy Delete	9	Debitis amet sed sunt est omnis.	Nobis qui omnis possimus fuga. Sed excepturi paria...	2019-02-22 15:34:46	2019-02-22 15:34:46	7
Edit Copy Delete	10	Earum facilis minima dignissimos.	Adipisci molestias fugiat corporis iste. Veritatis...	2019-02-22 15:34:46	2019-02-22 15:34:46	6
Edit Copy Delete	11	Temporibus in suscipit nihil nihil.	Vitae aliquid magnam dolorem maiores. Beatae offic...	2019-02-22 15:34:46	2019-02-22 15:34:46	7
Edit Copy Delete	12	Dolores sit laudantium laborum possimus.	Asperiores iste quos sit similique. Dolorem dignis...	2019-02-22 15:34:46	2019-02-22 15:34:46	6

Showing rows 0 - 66 (67 total, Query took 0.0020 seconds.)

```
SELECT * FROM `comments`
```

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 250 Filter rows: Search this table Sort by key: None

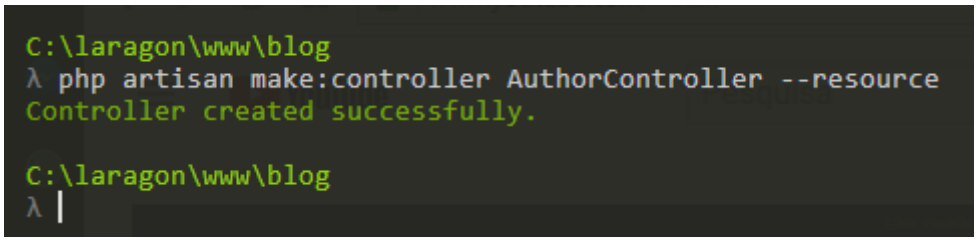
+ Options

	id	author_email	content	post_id	created_at	updated_at
Edit Copy Delete	1	antonette.bahringer@example.net	Consequatur vero quae explicabo est earum. Volupta...	20	2019-02-22 15:34:47	2019-02-22 15:34:47
Edit Copy Delete	2	nmoen@example.org	Expedita laudantium cupiditate deserunt et. Cum au...	19	2019-02-22 15:34:47	2019-02-22 15:34:47
Edit Copy Delete	3	wuckert.merl@example.org	Placeat eos quisquam natus facilis voluptatem. Odi...	2	2019-02-22 15:34:47	2019-02-22 15:34:47
Edit Copy Delete	4	reanna.johnston@example.net	Perferendis est et et et explicabo fuga corporis. ...	21	2019-02-22 15:34:48	2019-02-22 15:34:48
Edit Copy Delete	5	vnolan@example.org	Cupiditate qui blanditiis ea voluptatem dignissimo...	13	2019-02-22 15:34:48	2019-02-22 15:34:48
Edit Copy Delete	6	maximilian.vandervort@example.com	Sunt sapiente itaque veniam. Eos aut odit exercita...	4	2019-02-22 15:34:48	2019-02-22 15:34:48
Edit Copy Delete	7	laury14@example.net	Placeat cupiditate illum ut consequatur sunt debit...	32	2019-02-22 15:34:48	2019-02-22 15:34:48
Edit Copy Delete	8	hadley.mcclure@example.org	Qui voluptatibus alias recusandae maxime beatae es...	18	2019-02-22 15:34:48	2019-02-22 15:34:48
Edit Copy Delete	9	uwiza@example.org	Ab omnis nemo voluptas explicabo totam eaque aut. ...	14	2019-02-22 15:34:48	2019-02-22 15:34:48
Edit Copy Delete	10	alexanne30@example.org	Sapiente iste in neque omnis totam unde aliquam. N...	14	2019-02-22 15:34:48	2019-02-22 15:34:48
Edit Copy Delete	11	hansen.yvonne@example.org	Aut id quidem unde. Ut cumque sed quasi.	7	2019-02-22 15:34:48	2019-02-22 15:34:48
Edit Copy Delete	12	pswift@example.net	Adipisci quasi optio nesciunt id distinctio libero...	7	2019-02-22 15:34:48	2019-02-22 15:34:48

Aula 04 - Controllers e Rotas

Criando o controller para os autores

php artisan make:controller AuthorController --resource

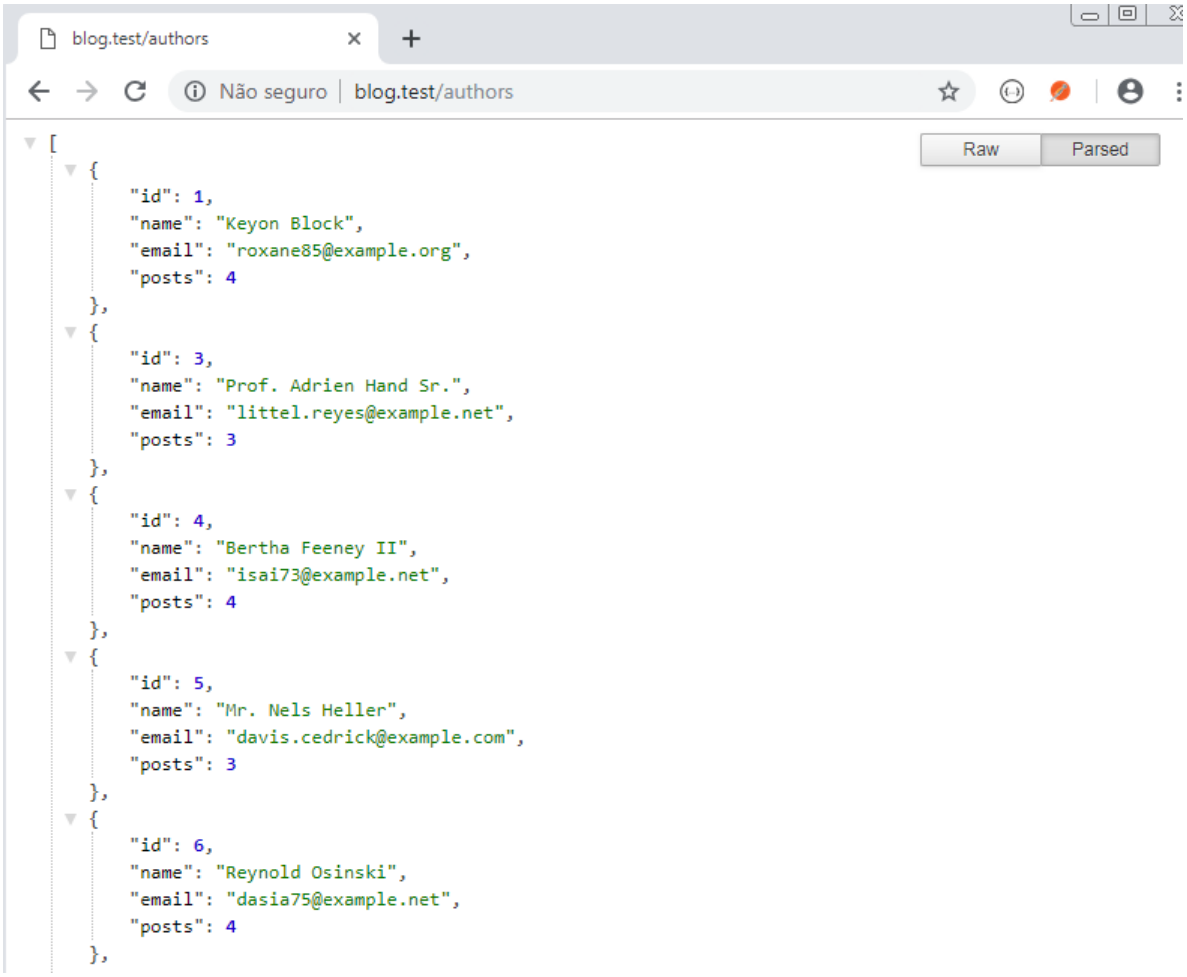


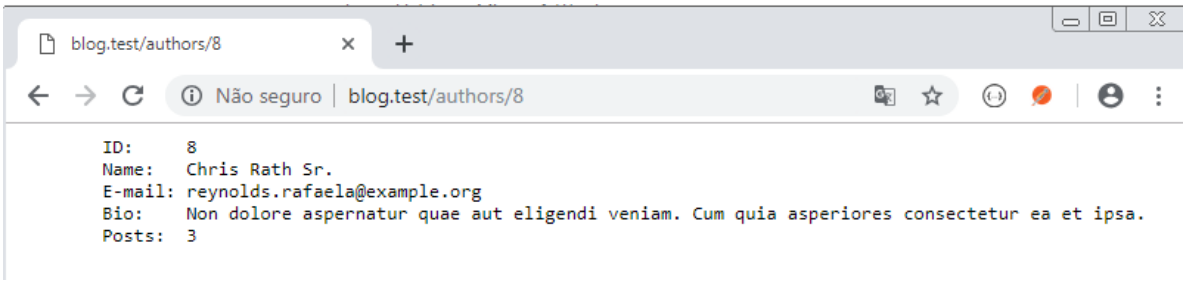
routes/web.php

Route::resource('authors','AuthorController');

app/Http/Controllers/AuthorController.php

```
<?php

namespace App\Http\Controllers;

use App\Author;
use Illuminate\Support\Facades\DB;
use Illuminate\Http\Request;

class AuthorController extends Controller
{
   public function index()
   {
      //$authors = json_encode(Author::all(), JSON_PRETTY_PRINT);
      $authors = Author::join('posts', 'posts.author_id', '=', 'authors.id')
         ->groupBy('authors.id')
         ->get([
            'authors.id',
            'authors.name',
            'authors.email',
            DB::raw('count(posts.id) as posts')
         ]
      );

      $authors = json_encode($authors, JSON_PRETTY_PRINT);

      return "<pre>$authors</pre>";
   }
   public function show(Author $author)
   {
      $posts = $author->posts->count();

      $text = <<<TEXT
      ID:    $author->id
      Name:   $author->name
      E-mail: $author->email
      Bio:    $author->bio
      Posts:  $posts
TEXT;
      /* A marca de fim do heredoc (TEXT;)
         deve ficar na coluna zero da linha */

      return "<pre>$text</pre>";
   }
}
```

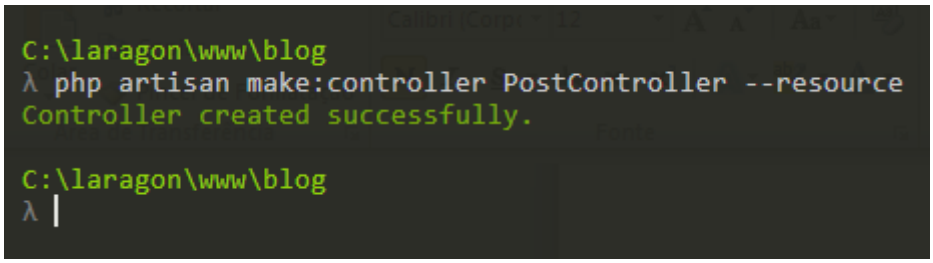

blog.test/authors
Não seguro | blog.test/authors
Raw
Parsed
[
{
"id": 1,
"name": "Keyon Block",
"email": "roxane85@example.org",
"posts": 4
},
{
"id": 3,
"name": "Prof. Adrien Hand Sr.",
"email": "littel.reyes@example.net",
"posts": 3
},
{
"id": 4,
"name": "Bertha Feeney II",
"email": "isai73@example.net",
"posts": 4
},
{
"id": 5,
"name": "Mr. Nels Heller",
"email": "davis.cedrick@example.com",
"posts": 3
},
{
"id": 6,
"name": "Reynold Osinski",
"email": "dasia75@example.net",
"posts": 4
},


blog.test/authors/8
Não seguro | blog.test/authors/8
ID: 8
Name: Chris Rath Sr.
E-mail: reynolds.rafaela@example.org
Bio: Non dolore aspernatur quae aut eligendi veniam. Cum quia asperiores consectetur ea et ipsa.
Posts: 3

Aula 05 - PostController

Criando o controller para os posts

php artisan make:controller PostController --resource

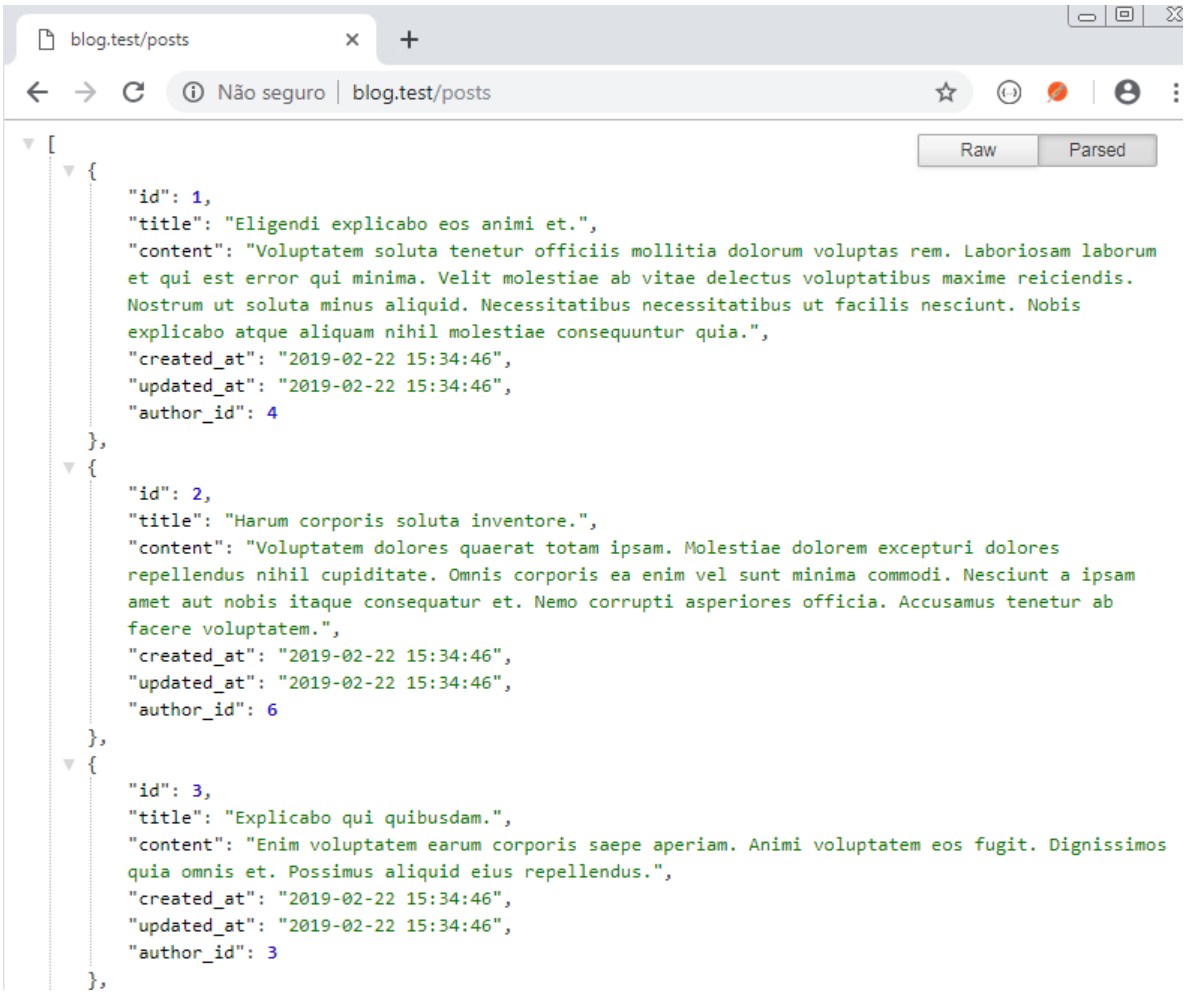


routes/web.php

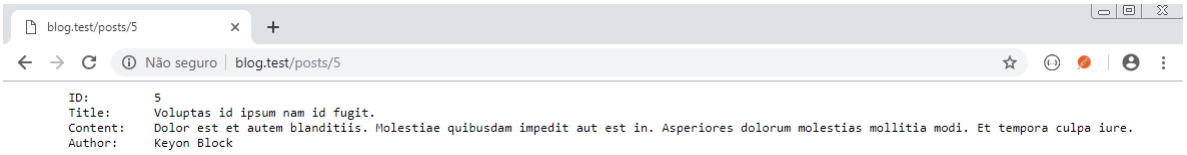
```
Route::resources([
   'authors' => 'AuthorController',
   'posts' => 'PostController'
]);
```

app/Http/Controllers/PostController.php

```
<?php

namespace App\Http\Controllers;

use App\Post;
use Illuminate\Support\Facades\DB;
use Illuminate\Http\Request;

class PostController extends Controller
{
   public function index()
   {
      $posts = json_encode(Post::all(), JSON_PRETTY_PRINT);
      return "<pre>$posts</pre>";
   }
   public function show(Post $post)
   {
      $author = $post->author->name;

      $text = <<<TEXT
      ID:        $post->id
      Title:      $post->title
      Content:    $post->content
      Author:     $author
TEXT;
      // A marca do fim do heredoc (TEXT;) deve ficar na coluna zero da linha.
      // ou teremos um erro - Unexpected end of file
      return "<pre>$text</pre>";
   }
}
```



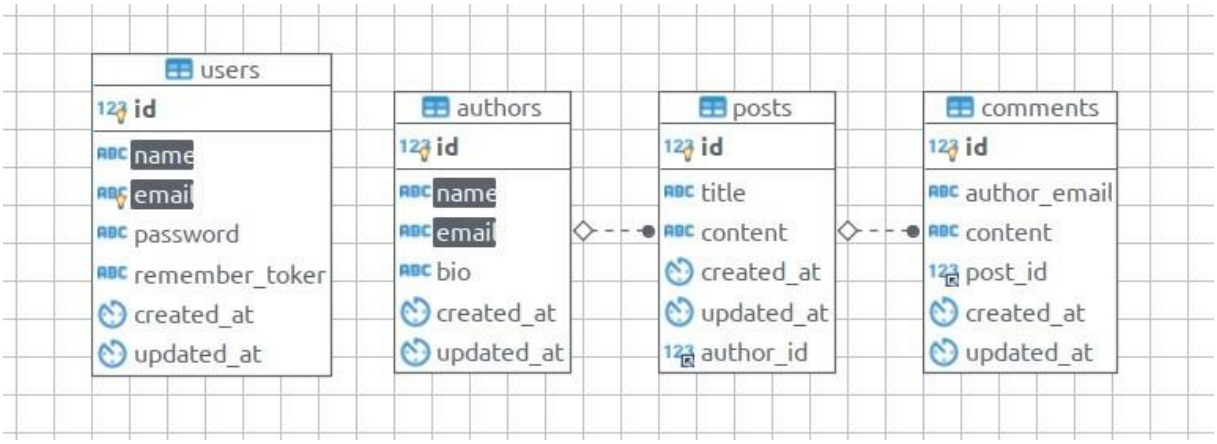

```
[
  {
    "id": 1,
    "title": "Eligendi explicabo eos animi et.",
    "content": "Voluptatem soluta tenetur officiis mollitia dolorum voluptas rem. Laboriosam laborum et qui est error qui minima. Velit molestiae ab vitae delectus voluptatibus maxime reiciendis. Nostrum ut soluta minus aliquid. Necessitatibus necessitatibus ut facilis nesciunt. Nobis explicabo atque aliquam nihil molestiae consequuntur quia.",
    "created_at": "2019-02-22 15:34:46",
    "updated_at": "2019-02-22 15:34:46",
    "author_id": 4
  },
  {
    "id": 2,
    "title": "Harum corporis soluta inventore.",
    "content": "Voluptatem dolores quaerat totam ipsam. Molestiae dolorem excepturi dolores repellendus nihil cupiditate. Omnis corporis ea enim vel sunt minima commodi. Nesciunt a ipsam amet aut nobis itaque consequatur et. Nemo corrupti asperiores officia. Accusamus tenetur ab facere voluptatem.",
    "created_at": "2019-02-22 15:34:46",
    "updated_at": "2019-02-22 15:34:46",
    "author_id": 6
  },
  {
    "id": 3,
    "title": "Explicabo qui quibusdam.",
    "content": "Enim voluptatem earum corporis saepe aperiam. Animi voluptatem eos fugit. Dignissimos quia omnis et. Possimus aliquid eius repellendus.",
    "created_at": "2019-02-22 15:34:46",
    "updated_at": "2019-02-22 15:34:46",
    "author_id": 3
  },
```


```
ID:         5
Title:      Voluptas id ipsum nam id fugit.
Content:    Dolor est et autem blanditiis. Molestiae quibusdam impedit aut est in. Asperiores dolorum molestias mollitia modi. Et tempora culpa iure.
Author:     Keyon Block
```

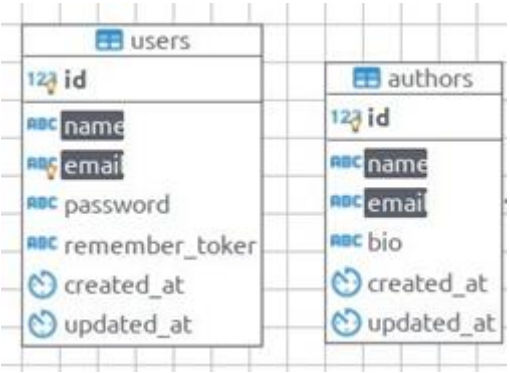
Aula 06 - Login

Fazer login dos autores



Alterar o model Author para relacionar com User

- Tabelas users e authors
- Tabelas users e authors com relacionamento 1:1



Alterar a tabela authors

Criar migration para adicionar user_id foreign e remover as colunas name e email

php artisan make:migration add_user_id_remove_name_email_from_authors --table=authors

```
C:\laragon\www\blog
λ php artisan make:migration add_user_id_remove_name_email_from_authors --table=authors
Created Migration: 2019_02_22_211703_add_user_id_remove_name_email_from_authors

C:\laragon\www\blog
λ
```

Código para a migration

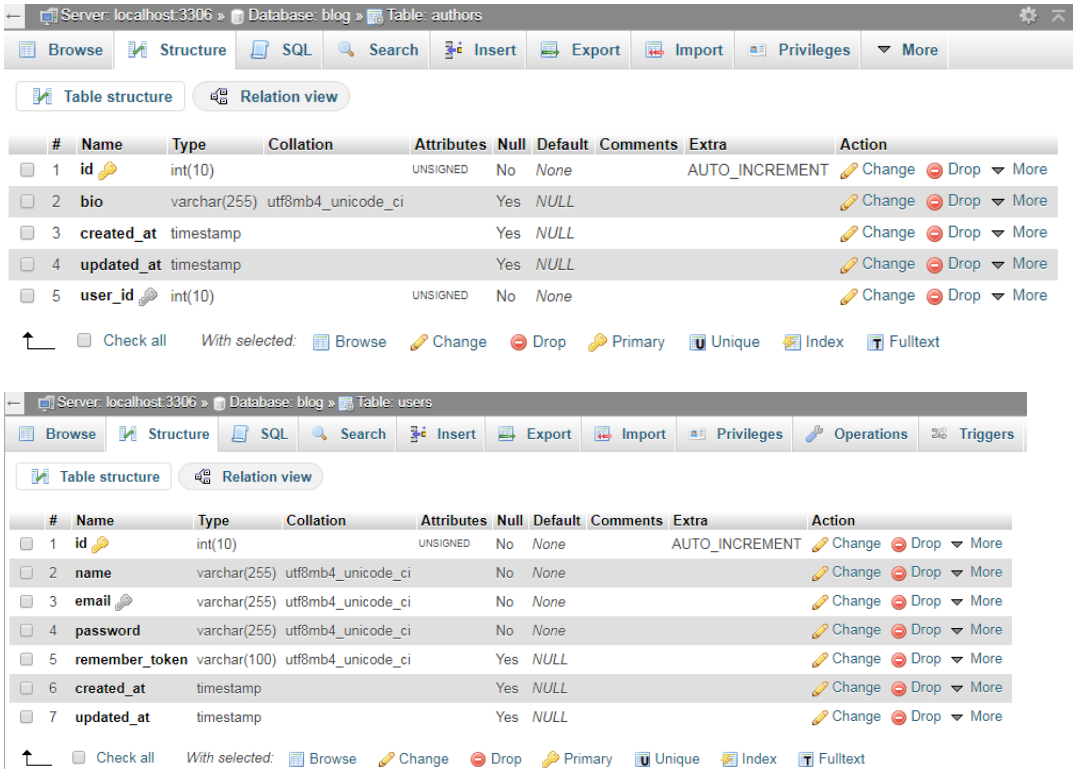
database/migrations/add_user_id_remove_name_email_from_authors.php

```
<?php

use Illuminate\Support\Facades\Schema;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Database\Migrations\Migration;

class AddUserIdRemoveNameEmailFromAuthors extends Migration
{
    public function up()
    {
        Schema::table('authors', function (Blueprint $table) {
            $table->dropColumn('name');
            $table->dropColumn('email');
            $table->integer('user_id')->unsigned();

            $table->foreign('user_id')
                ->references('id')
                ->on('users')
                ->onDelete('cascade');
        });
    }

    public function down()
    {
        Schema::table('authors', function (Blueprint $table) {
            $table->string('name');
            $table->string('email');
            $table->dropForeign('authors_user_id_foreign');
            $table->dropColumn('user_id');
        });
    }
}
```

php artisan migrate

php artisan migrate:refresh

```
C:\laragon\www\blog
λ php artisan migrate:refresh
Rolling back: 2019_02_22_044234_add_author_id_to_posts
Rolled back:  2019_02_22_044234_add_author_id_to_posts
Rolling back: 2019_02_22_044010_create_authors_table
Rolled back:  2019_02_22_044010_create_authors_table
Rolling back: 2019_02_22_043021_create_comments_table
Rolled back:  2019_02_22_043021_create_comments_table
Rolling back: 2019_02_22_042912_create_posts_table
Rolled back:  2019_02_22_042912_create_posts_table
Rolling back: 2014_10_12_100000_create_password_resets_table
Rolled back:  2014_10_12_100000_create_password_resets_table
Rolling back: 2014_10_12_000000_create_users_table
Rolled back:  2014_10_12_000000_create_users_table
Migrating: 2014_10_12_000000_create_users_table
Migrated:  2014_10_12_000000_create_users_table
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated:  2014_10_12_100000_create_password_resets_table
Migrating: 2019_02_22_042912_create_posts_table
Migrated:  2019_02_22_042912_create_posts_table
Migrating: 2019_02_22_043021_create_comments_table
Migrated:  2019_02_22_043021_create_comments_table
Migrating: 2019_02_22_044010_create_authors_table
Migrated:  2019_02_22_044010_create_authors_table
Migrating: 2019_02_22_044234_add_author_id_to_posts
Migrated:  2019_02_22_044234_add_author_id_to_posts
Migrating: 2019_02_22_211703_add_user_id_remove_name_email_from_authors
Migrated:  2019_02_22_211703_add_user_id_remove_name_email_from_authors
```



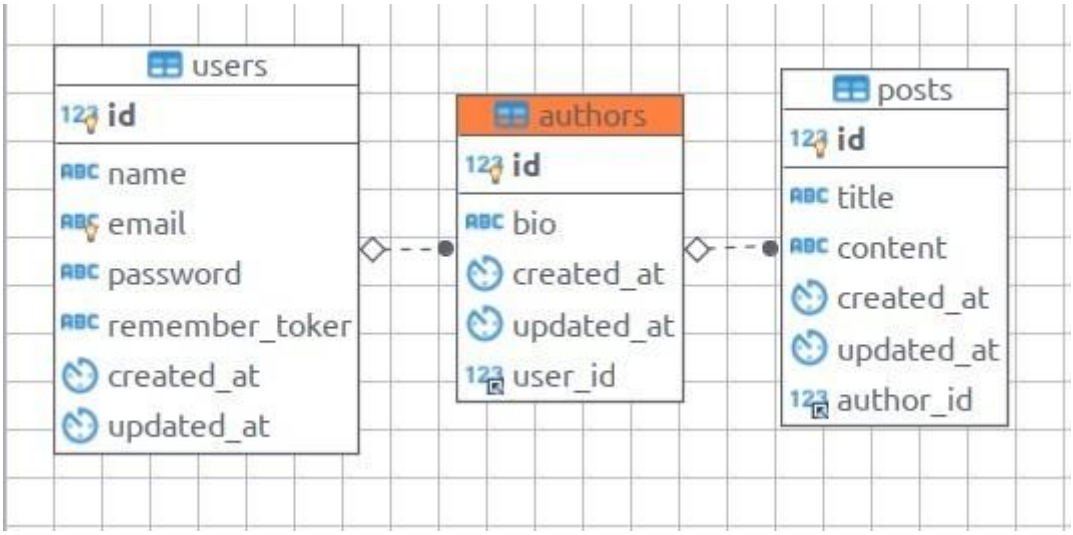
Ajustar os models User e Author

Setar o relacionamento: adicionar em app\User.php

```
public function author()
{
   return $this->hasOne('App\Author');
}
```

Setar o relacionamento: adicionar em app\Author.php

```
public function user()
{
   return $this->belongsTo('App\User');
}
```



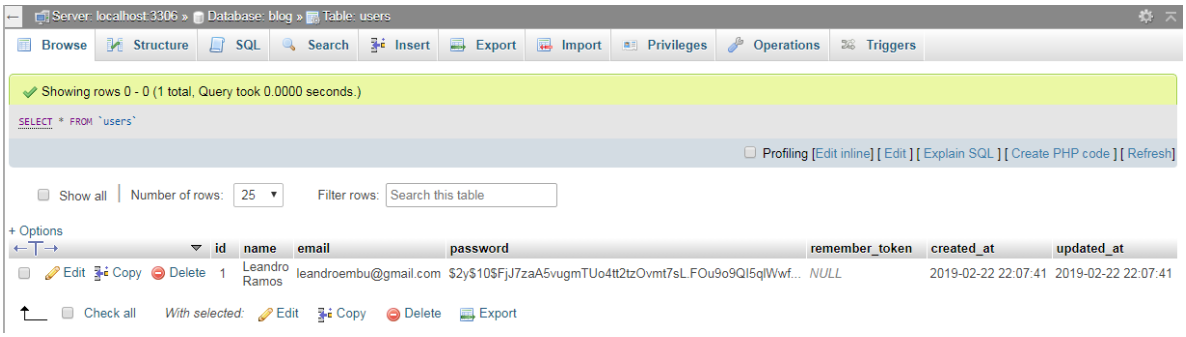
Inserir user e author no Banco de Dados

php artisan tinker

No tinker:

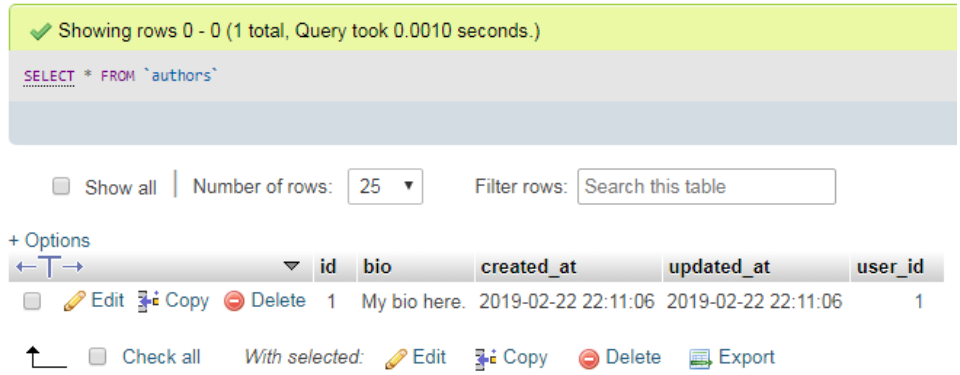
```
>>> $user = new User
>>> $user->name = "Leandro Ramos"
>>> $user->email = "leandroembu@gmail.com"
>>> $user->password = Hash::make('123456');
>>> $user->save()
```

```
C:\laragon\www\blog
λ php artisan tinker
Psy Shell v0.9.6 (PHP 7.2.11 — cli) by Justin Hileman
>>> $user = new User
[!] Aliasing 'User' to 'App\User' for this Tinker session.
=> App\User {#2879}
>>> $user->name = "Leandro Ramos"
=> "Leandro Ramos"
>>> $user->email = "leandroembu@gmail.com"
=> "leandroembu@gmail.com"
>>> $user->password = Hash::make('123456');
=> "$2y$10$FjJ7zaA5vugmTUo4tt2tzOvmt7sL.FOu9o9QI5qlWwflBek1vlXlO"
>>> $user->save()
=> true
>>>
```



>>> $author = new Author
>>> $author->bio = "My bio here."
>>> $user->author()->save($author)

```
>>> $author = new Author
[!] Aliasing 'Author' to 'App\Author' for this Tinker session.
=> App\Author {#2876}
>>> $author->bio = "My bio here."
=> "My bio here."
>>> $user->author()->save($author)
=> App\Author {#2876
     bio: "My bio here.",
     user_id: 1,
     updated_at: "2019-02-22 22:11:06",
     created_at: "2019-02-22 22:11:06",
     id: 1,
   }
>>>
```



>>> $u- = User::first();

```
>>> $u = User::first();
=> App\User {#2888
     id: 1,
     name: "Leandro Ramos",
     email: "leandroembu@gmail.com",
     created_at: "2019-02-22 22:07:41",
     updated_at: "2019-02-22 22:07:41",
   }
>>>
```

>>> $u- -> author -> bio;

```
>>> $u -> author -> bio;
=> "My bio here."
>>>
```

Criar a autenticação

php artisan make:auth

```
C:\laragon\www\blog
λ php artisan make:auth
Authentication scaffolding generated successfully.

C:\laragon\www\blog
λ
```

Limpar o cache

php artisan config:cache

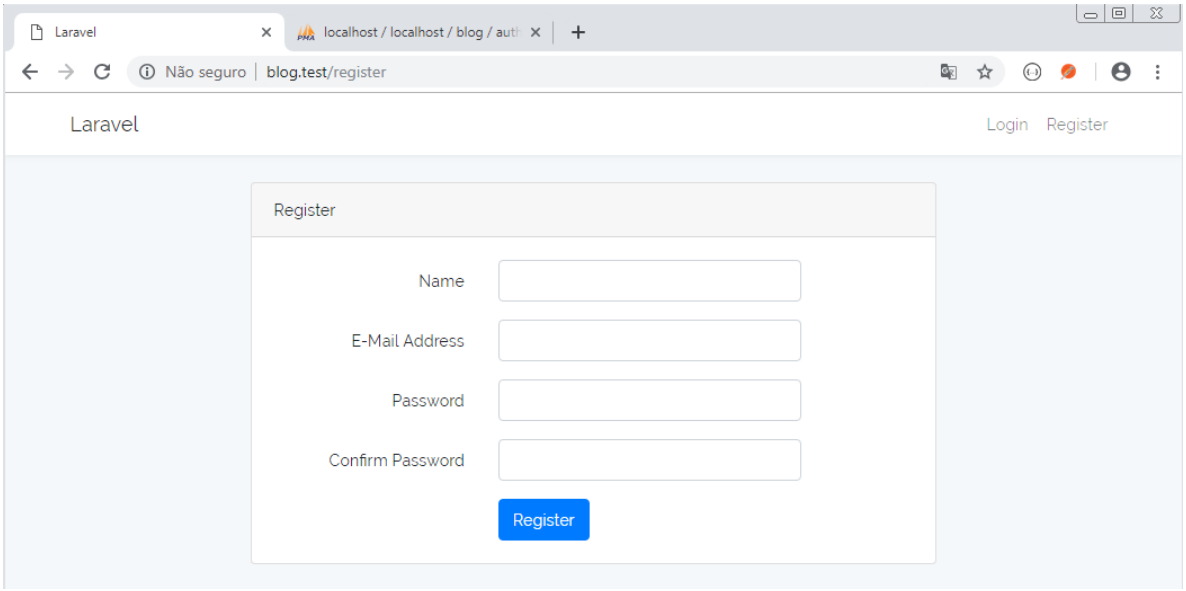
```
C:\laragon\www\blog
λ php artisan config:cache
Configuration cache cleared!
Configuration cached successfully!

C:\laragon\www\blog
λ
```

routes/web.php

```
Route::resources([
    'authors' => 'AuthorController',
    'posts' => 'PostController'
]);

// Criado automaticamente
Auth::routes();
Route::get('/home', 'HomeController@index')->name('home');
```

php artisan route:list

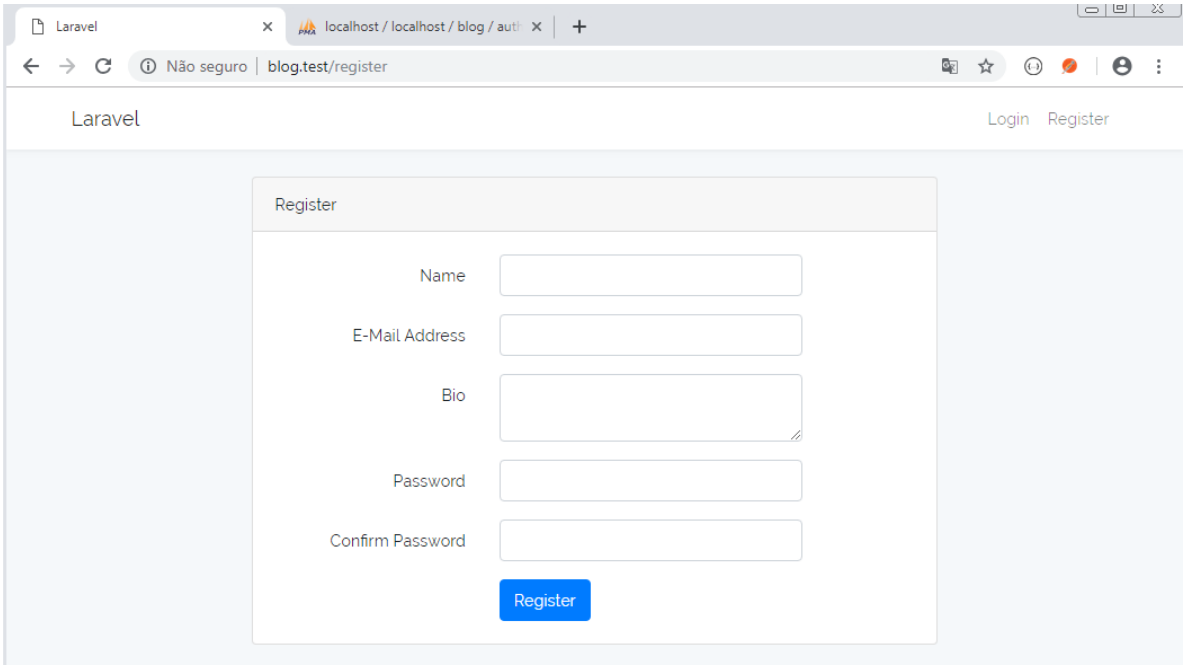
```
C:\laragon\www\blog
λ php artisan route:list
+--------+-----------+-----------------------+------------------+-------------------------------------------------------------------+-------------+
| Domain | Method    | URI                   | Name             | Action                                                            | Middleware  |
+--------+-----------+-----------------------+------------------+-------------------------------------------------------------------+-------------+
|        | GET|HEAD  | /                     |                  | Closure                                                           | web         |
|        | GET|HEAD  | api/user              |                  | Closure                                                           | api,auth:api |
|        | GET|HEAD  | authors               | authors.index    | App\Http\Controllers\AuthorController@index                       | web         |
|        | POST      | authors               | authors.store    | App\Http\Controllers\AuthorController@store                       | web         |
|        | GET|HEAD  | authors/create        | authors.create   | App\Http\Controllers\AuthorController@create                      | web         |
|        | GET|HEAD  | authors/{author}      | authors.show     | App\Http\Controllers\AuthorController@show                        | web         |
|        | PUT|PATCH | authors/{author}      | authors.update   | App\Http\Controllers\AuthorController@update                      | web         |
|        | DELETE    | authors/{author}      | authors.destroy  | App\Http\Controllers\AuthorController@destroy                     | web         |
|        | GET|HEAD  | authors/{author}/edit | authors.edit     | App\Http\Controllers\AuthorController@edit                        | web         |
|        | GET|HEAD  | home                  | home             | App\Http\Controllers\HomeController@index                         | web,auth    |
|        | GET|HEAD  | login                 | login            | App\Http\Controllers\Auth\LoginController@showLoginForm           | web,guest   |
|        | POST      | login                 |                  | App\Http\Controllers\Auth\LoginController@login                   | web,guest   |
|        | POST      | logout                | logout           | App\Http\Controllers\Auth\LoginController@logout                  | web         |
|        | POST      | password/email        | password.email   | App\Http\Controllers\Auth\ForgotPasswordController@sendResetLinkEmail | web,guest   |
|        | GET|HEAD  | password/reset        | password.request | App\Http\Controllers\Auth\ForgotPasswordController@showLinkRequestForm | web,guest   |
|        | POST      | password/reset        |                  | App\Http\Controllers\Auth\ResetPasswordController@reset           | web,guest   |
|        | GET|HEAD  | password/reset/{token} | password.reset   | App\Http\Controllers\Auth\ResetPasswordController@showResetForm   | web,guest   |
|        | POST      | posts                 | posts.store      | App\Http\Controllers\PostController@store                         | web         |
|        | GET|HEAD  | posts                 | posts.index      | App\Http\Controllers\PostController@index                         | web         |
|        | GET|HEAD  | posts/create          | posts.create     | App\Http\Controllers\PostController@create                        | web         |
|        | GET|HEAD  | posts/{post}          | posts.show       | App\Http\Controllers\PostController@show                          | web         |
|        | PUT|PATCH | posts/{post}          | posts.update     | App\Http\Controllers\PostController@update                        | web         |
|        | DELETE    | posts/{post}          | posts.destroy    | App\Http\Controllers\PostController@destroy                       | web         |
|        | GET|HEAD  | posts/{post}/edit     | posts.edit       | App\Http\Controllers\PostController@edit                          | web         |
|        | GET|HEAD  | register              | register         | App\Http\Controllers\Auth\RegisterController@showRegistrationForm | web,guest   |
|        | POST      | register              |                  | App\Http\Controllers\Auth\RegisterController@register             | web,guest   |
+--------+-----------+-----------------------+------------------+-------------------------------------------------------------------+-------------+

C:\laragon\www\blog
λ
```



- Adicionar o campo "bio" ao form de registro
- Adicionar ao formulário (antes do campo password)

resources/views/auth/register.blade.php:

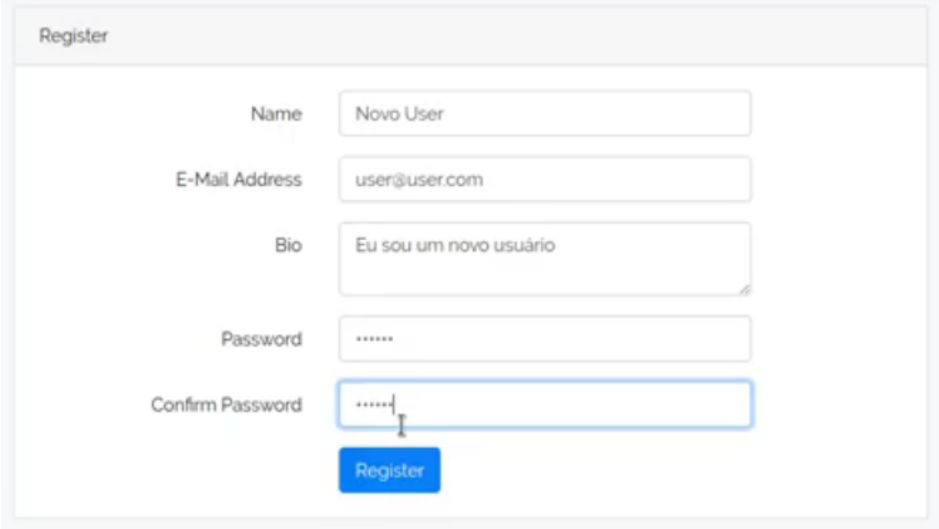
```
<div class="form-group row">
   <label for="bio" class="col-md-4 col-form-label text-md-right">Bio</label>
   <div class="col-md-6">
      <textarea id="bio" class="form-control" name="bio" value="{{ old('bio')  }}"></textarea>
   </div>
</div>
```



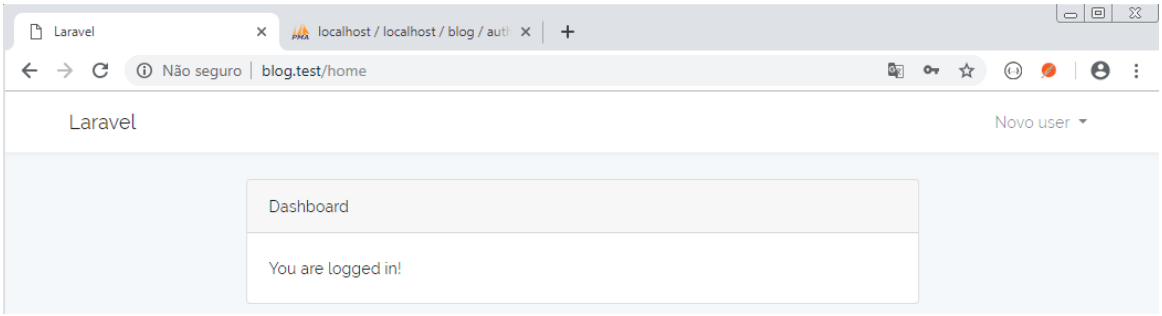
Ajustar o controller

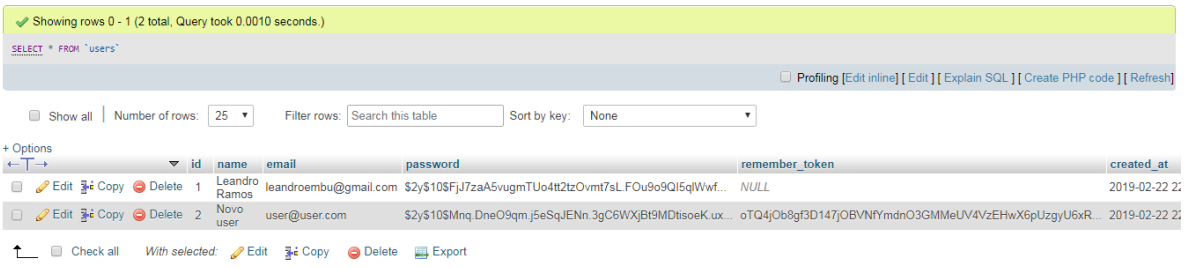
app/Http/Controllers/Auth/RegisterController.php

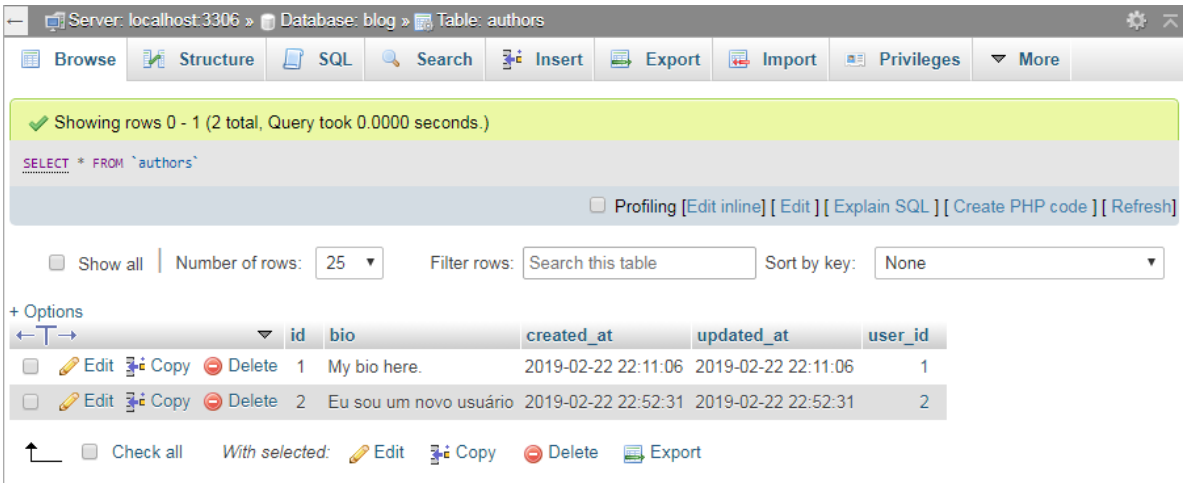
```
// Coloque abaixo do namespace
use App\User;

// ... aqui tem um monte de código
// troque o código do método create por isso:
protected function create(array $data)
{
   $user = User::create([
      'name' => $data['name'],
      'email' => $data['email'],
      'password' => bcrypt($data['password']),
   ]);

   $author = new Author;
   $author->bio = $data['bio'];

   $user->author()->save($author);

   return $user;
}
```

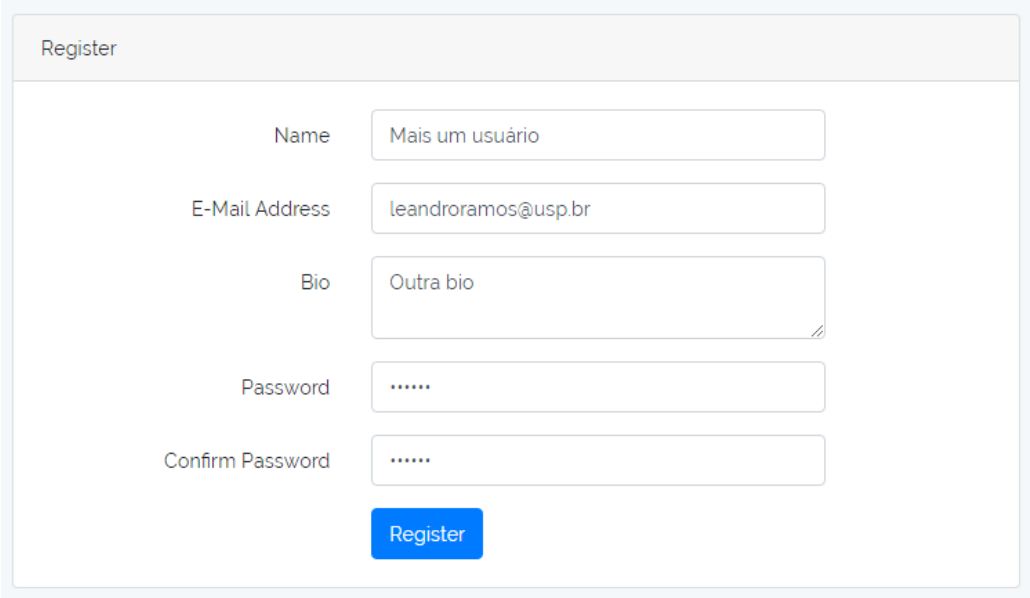


Senha: 123456

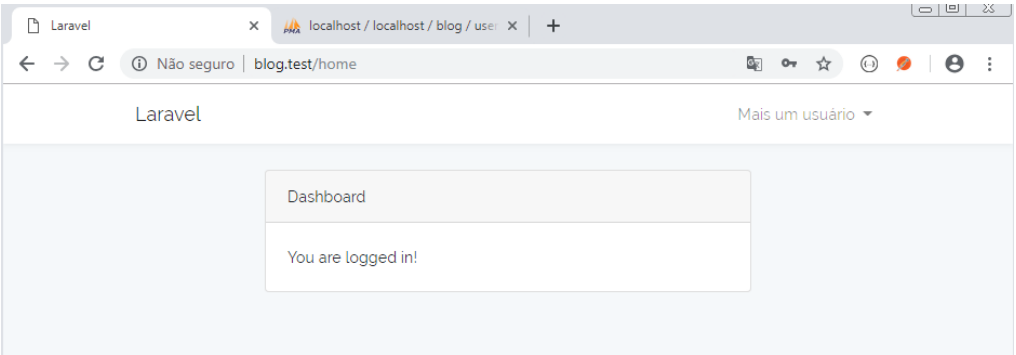


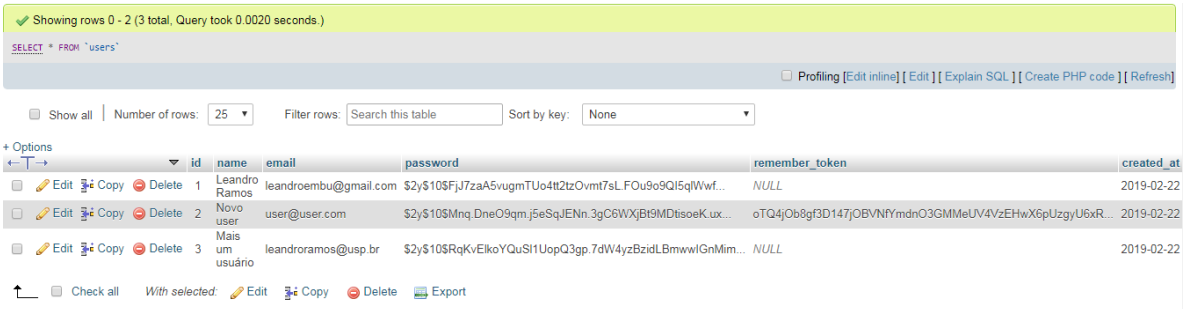


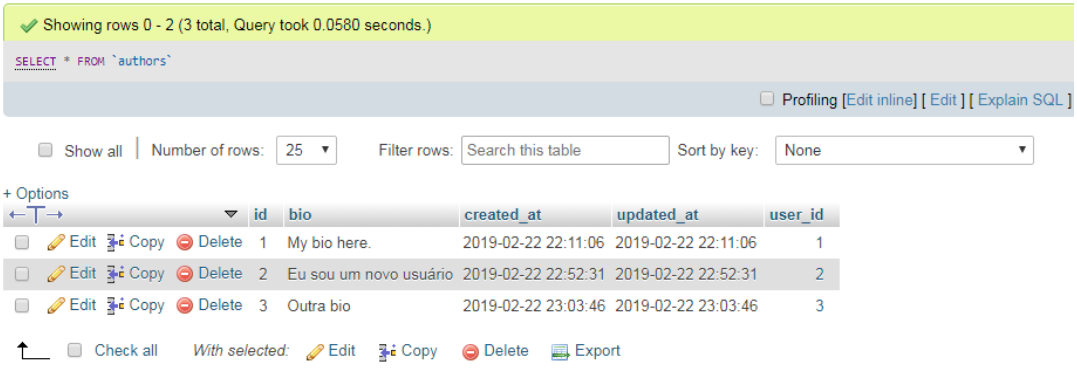




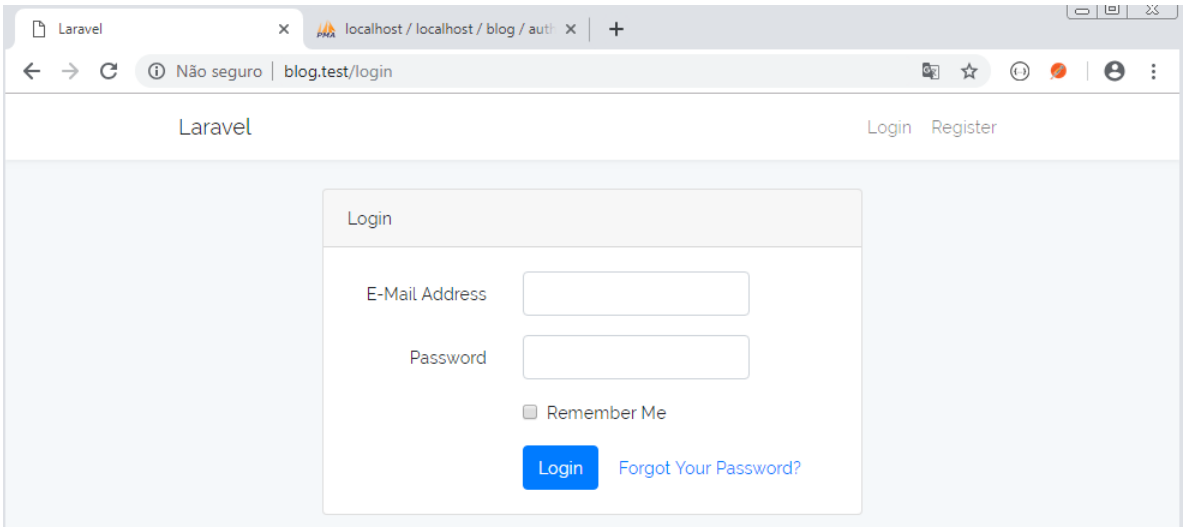
Senha: 654321

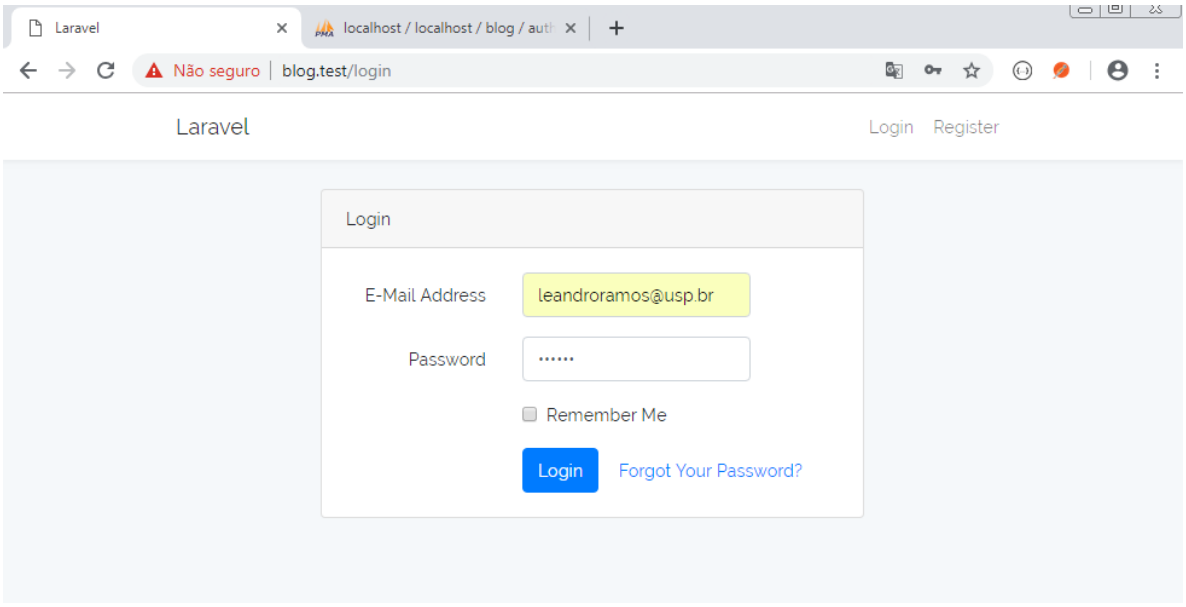


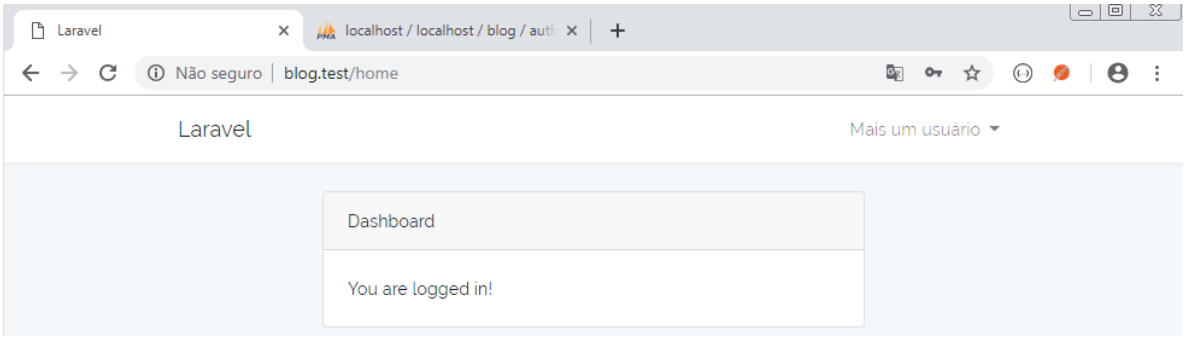




Fazer login:


Laravel
localhost / localhost / blog / auth
Não seguro | blog.test/login
Laravel
Login Register
Login
E-Mail Address
Password
Remember Me
Login
Forgot Your Password?


Laravel
localhost / localhost / blog / auth
Não seguro | blog.test/login
Laravel
Login Register
Login
E-Mail Address
leandroramos@usp.br
Password
Remember Me
Login
Forgot Your Password?


Laravel
localhost / localhost / blog / auth
Não seguro | blog.test/home
Laravel
Mais um usuário
Dashboard
You are logged in!

Aula 07 - Views

Populando o banco de dados

- Apague o banco de dados blog e em seguida o recrie.

- Recrie as tabelas:

php artisan migrate

```
C:\laragon\www\blog
λ php artisan migrate
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated:  2014_10_12_000000_create_users_table
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated:  2014_10_12_100000_create_password_resets_table
Migrating: 2019_02_22_042912_create_posts_table
Migrated:  2019_02_22_042912_create_posts_table
Migrating: 2019_02_22_043021_create_comments_table
Migrated:  2019_02_22_043021_create_comments_table
Migrating: 2019_02_22_044010_create_authors_table
Migrated:  2019_02_22_044010_create_authors_table
Migrating: 2019_02_22_044234_add_author_id_to_posts
Migrated:  2019_02_22_044234_add_author_id_to_posts
Migrating: 2019_02_22_211703_add_user_id_remove_name_email_from_authors
Migrated:  2019_02_22_211703_add_user_id_remove_name_email_from_authors

C:\laragon\www\blog
λ
```

- Altere os arquivos a seguir:

database/factories/AuthorFactory.php

```
<?php

use Faker\Generator as Faker;

$factory->define(App\Author::class, function (Faker $faker) {
    return [
        // 'name'  => $faker->name,
        //'email' => $faker->unique()->safeEmail,
        'bio'   => $faker->paragraph(1),
        'user_id' => $faker->unique()->numberBetween(1,10)
    ];
});
```

database/factories/UserFactory.php

```
<?php

use Faker\Generator as Faker;

$factory->define(App\User::class, function (Faker $faker) {
    return [
        'name' => $faker->name,
        'email' => $faker->unique()->safeEmail,
        'password' => '$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFlYg7B77UdFm', // secret
        'remember_token' => str_random(10),
    ];
});
```

database/seeds/DatabaseSeeder.php

```
<?php

use Illuminate\Database\Seeder;

class DatabaseSeeder extends Seeder
{
    public function run()
    {
        // $this->call(UsersTableSeeder::class);

        echo "Creating 10 users...\n";
        factory(App\User::class, 10)->create();

        echo "Creating 10 authors...\n";
        factory(App\Author::class, 10)->create();

        echo "Creating 36 posts related to random authors...\n";
        factory(App\Post::class, 36)->create();

        echo "Creating 67 comments related to random posts...\n";
        factory(App\Comment::class, 67)->create();
    }
}
```

- Popule o banco de dados.

```
php artisan db:seed
```

Listando todos os posts

app/Controllers/PostController.php

```
public function index()
{
   $posts = Post::all();
   return view('posts.index', compact('posts'));
}
```

resources/views/posts/index.blade.php

```
@extends('layouts.app')
@section('content')
  <div class="row justify-content-center">
    <div class="col-md-8">
      <div class="card card-default">
        @foreach ($posts as $post)
          <div class="card-header">
            <h3>{{ $post->title }} :: por {{ $post->author->user->name }}</h3>
          </div>
          <div class="card-body">
            <p>
              {{ substr($post->content, 0, 60) }}...
              <a href="{{ action('PostController@show', $post->id) }}" title="Ler o post">Ler o post</a>
            </p>
          </div>
        @endforeach
      </div>
    </div>
  </div>
@endsection
```

resources/views/posts/show.blade.php

```
@extends('layouts.app')
@section('content')
  <div class="row justify-content-center">
    <div class="col-md-8">
      <div class="card card-default">
        <div class="card-header">
          <h3>{{ $post->title }} :: por {{ $post->author->user->name }}</h3>
        </div>
        <div class="card-body">
          <p>
            {{ $post->content }}
          </p>
          <h4>Comentários:</h4>
          <!-- Comentários do post -->
          @foreach($post->comments as $comment)
            <div class="card card-default">
              <div class="card-header">
                {{ $comment->author_email }}
              </div>
              <div class="card-body">
                {{ $comment->content }}
              </div>
            </div>
            <br>
          @endforeach
        </div>
      </div>
    </div>
  </div>
@endsection
```

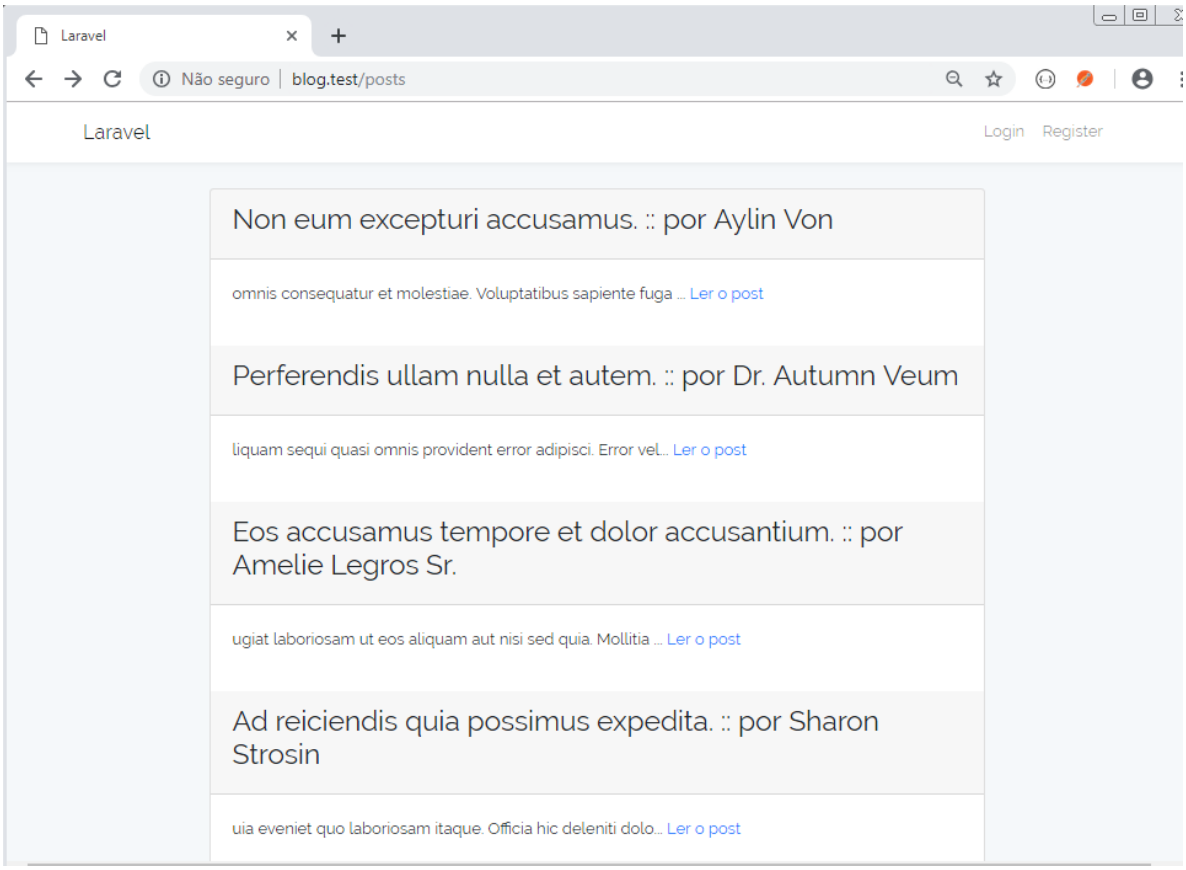
resources/views/layouts/app.blade.php

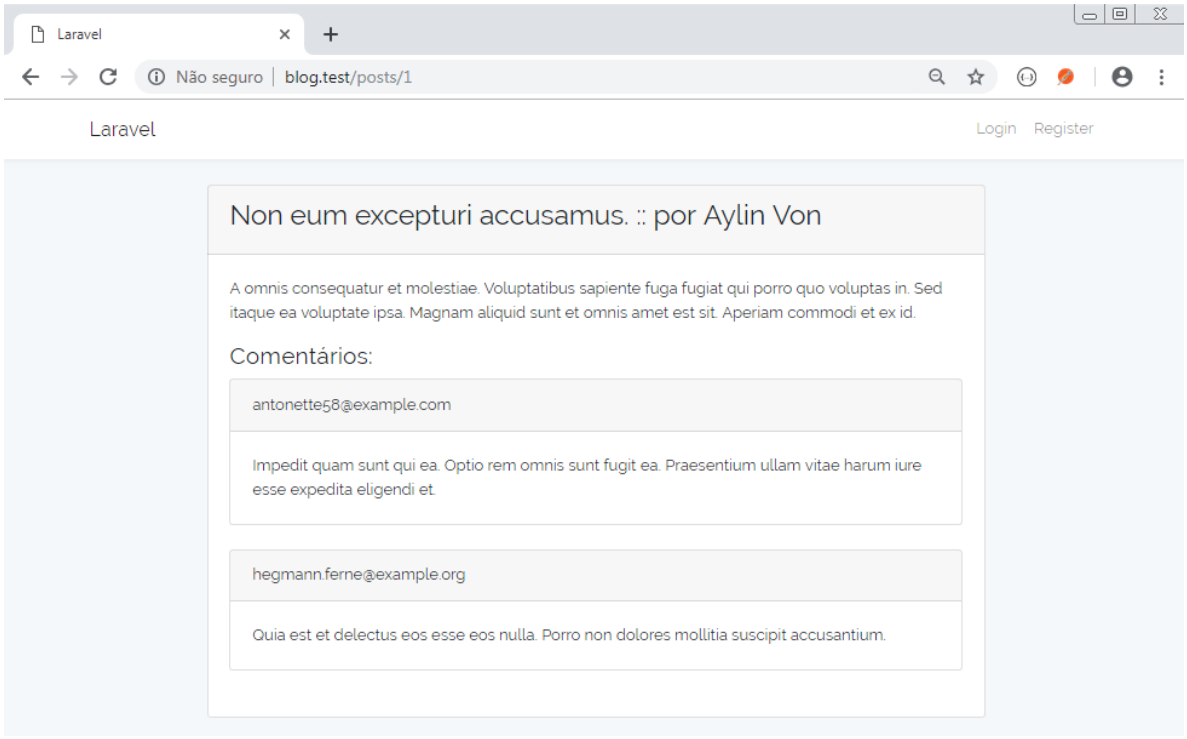
```
<!DOCTYPE html>
<html lang="{{ str_replace('_', '-', app()->getLocale()) }}">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <!-- CSRF Token -->
    <meta name="csrf-token" content="{{ csrf_token() }}">

    <title>{{ config('app.name', 'Laravel') }}</title>

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

    <!-- Fonts -->
    <link rel="dns-prefetch" href="https://fonts.gstatic.com">
    <link href="https://fonts.googleapis.com/css?family=Nunito" rel="stylesheet" type="text/css">

    <!-- Styles -->
    <link href="{{ asset('css/app.css') }}" rel="stylesheet">
</head>
<body>
    <div id="app">
        <nav class="navbar navbar-expand-md navbar-light navbar-laravel">
            <div class="container">
                <a class="navbar-brand" href="{{ url('/') }}">
                    {{ config('app.name', 'Laravel') }}
                </a>
                <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-
label="{{ __('Toggle navigation') }}">
                    <span class="navbar-toggler-icon"></span>
                </button>

                <div class="collapse navbar-collapse" id="navbarSupportedContent">
                    <!-- Left Side Of Navbar -->
                    <ul class="navbar-nav mr-auto">

                    </ul>

                    <!-- Right Side Of Navbar -->
                    <ul class="navbar-nav ml-auto">
                        <!-- Authentication Links -->
                        @guest
                            <li class="nav-item">
                                <a class="nav-link" href="{{ route('login') }}">{{ __('Login') }}</a>
                            </li>

                            <li class="nav-item">
                                <a class="nav-link" href="{{ route('register') }}">{{ __('Register') }}</a>
                            </li>
```

```
                @else
                    <li class="nav-item dropdown">
                        <a id="navbarDropdown" class="nav-link dropdown-toggle" href="#" role="button"
data-toggle="dropdown" aria-haspopup="true" aria-expanded="false" v-pre>
                            {{ Auth::user()->name }} <span class="caret"></span>
                        </a>

                        <div class="dropdown-menu dropdown-menu-right" aria-
labelledby="navbarDropdown">
                            <a class="dropdown-item" href="{{ route('logout') }}"
                               onclick="event.preventDefault();
                                        document.getElementById('logout-form').submit();">
                                {{ __('Logout') }}
                            </a>

                            <form id="logout-form" action="{{ route('logout') }}" method="POST" style="display:
none;">
                                @csrf
                            </form>
                        </div>
                    </li>
                @endguest
            </ul>
        </div>
    </div>
</nav>

<main class="py-4">
    @yield('content')
</main>
</div>
</body>
</html>
```

app/Controllers/PostController.php

```
<?php

namespace App\Http\Controllers;

use App\Post;
use Illuminate\Support\Facades\DB;
use Illuminate\Http\Request;

class PostController extends Controller
{
   public function index()
   {
      $posts = Post::all();
      return view('posts.index', compact('posts'));
   }
   public function show(Post $post)
   {
      return view('posts.show', compact('post'));
   }
}
```




Laravel
Não seguro | blog.test/posts/1
Laravel
Login Register
Non eum excepturi accusamus. :: por Aylin Von
A omnis consequatur et molestiae. Voluptatibus sapiente fuga fugiat qui porro quo voluptas in. Sed itaque ea voluptate ipsa. Magnam aliquid sunt et omnis amet est sit. Aperiam commodi et ex id.
Comentários:
antonette58@example.com
Impedit quam sunt qui ea. Optio rem omnis sunt fugit ea. Praesentium ullam vitae harum iure esse expedita eligendi et.
hegmann.ferne@example.org
Quia est et delectus eos esse eos nulla. Porro non dolores mollitia suscipit accusantium.

Aula 08 - Views - Criando Posts

Logar o usuário

O user (que também é um author) só poderá criar e editar posts se estiver logado.

Ajustar o PostController para autenticar o usuário

Criar método __construct em:

app/Http/Controllers/PostController.php

```
public function __construct()
{
    $this->middleware('auth')->except(['index', 'show']);
}
```

Criando o formulário

resources/views/forms/post.blade.php

```
@extends('layouts.app')
@section('content')
  <div class="row justify-content-center">
    <div class="col-md-8">
      <div class="card card-default">
        <div class="card-header">
          <h3>Post</h3>
        </div>
        <div class="card-body">
          <form method="post" action="{{ $action or url('posts') }}">
            {{csrf_field()}}
            @isset($post) {{method_field('patch')}} @endisset
            <input name="author_id" type="hidden" value="{{ encrypt(Auth::User()->author->id) }}">
            <div class="form-group row">
              <label for="title" class="col-md-4 col-form-label text-md-right">Title</label>
              <div class="col-md-6">
                <input id="title" class="form-control" name="title" type="text" value="{{ $post->title or ''}}">
              </div>
            </div>
            <div class="form-group row">
              <label for="content" class="col-md-4 col-form-label text-md-right">Content</label>
              <div class="col-md-6">
                <textarea id="content" class="form-control" name="content">{{ $post->content or ''
}}</textarea>
              </div>
            </div>
            <div class="form-group row mb-0">
              <div class="col-md-6 offset-md-4">
                <button type="submit" class="btn btn-primary">
                  Save
                </button>
              </div>
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>
@endsection
```

resources/views/posts/create.blade.php

```
@include('forms.post')
```

resources/views/posts/edit.blade.php

```
@include('forms.post')
```

Ajustando o app/Http/Controllers/PostController.php

Ajustar o controller para efetuar as operações corretamente

Método index

```
public function index()
{
    $posts = Post::orderBy('created_at', 'desc')->get();
    return view('posts.index', compact('posts'));
}
```

Método create

```
public function create()
{
    return view('posts.create');
}
```

Método store

```
public function store(Request $request)
{
    $post = new Post;

    $post->title  = $request->title;
    $post->content  = $request->content;
    $post->author_id  = decrypt($request->author_id);

    $post->save();
    $request->session()->flash('alert-success', 'Post criado com sucesso!');
    return redirect()->route('posts.index');
}
```

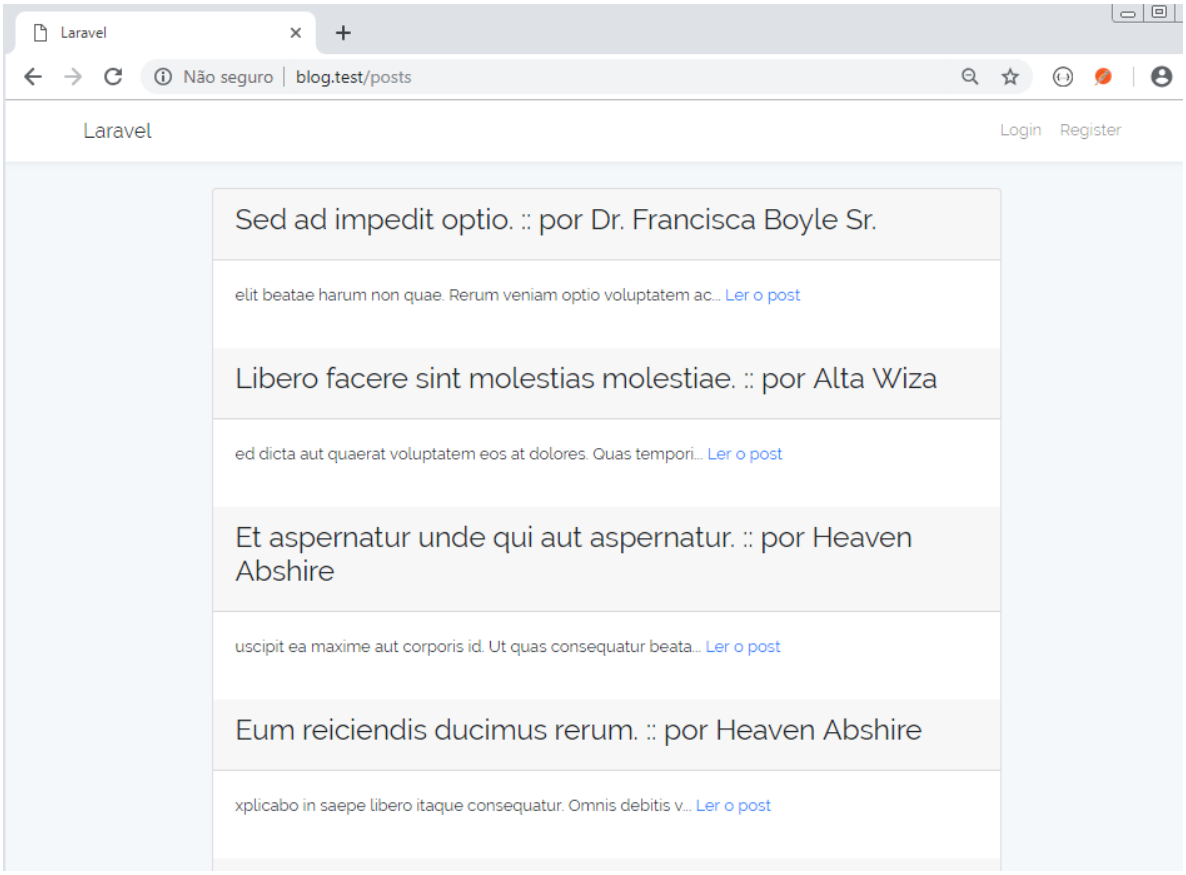
Método edit

```
public function edit(Post $post)
{
    $action = action('PostController@update', $post->id);

    return view('posts.edit', compact('post', "action"));
}
```

View para edição do Post

resources/views/posts/edit.blade.php

```
@include('forms.post')
```



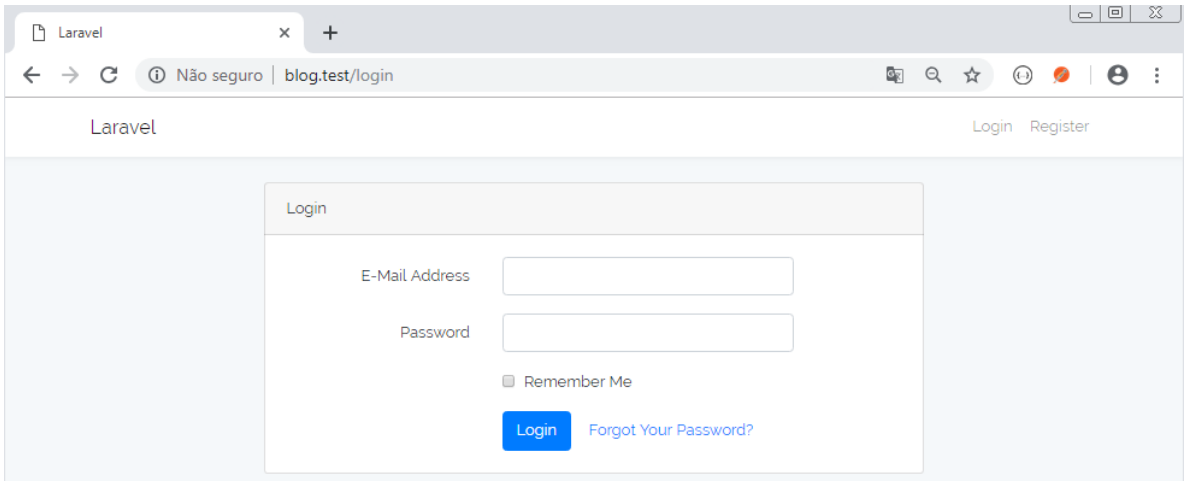
Entrando com:

http://blog.test/posts/create

ou

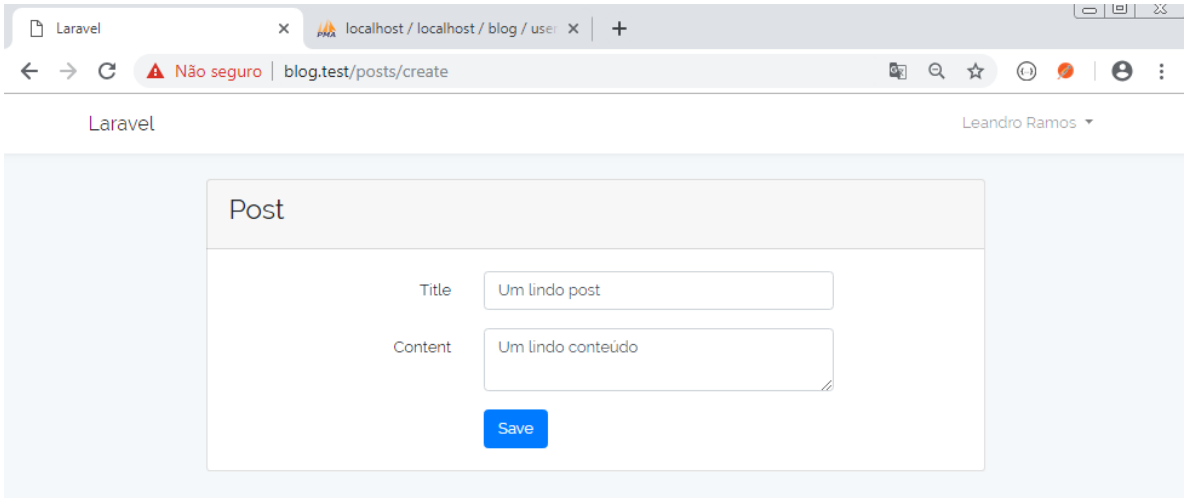
http://blog.test/posts/5/edit

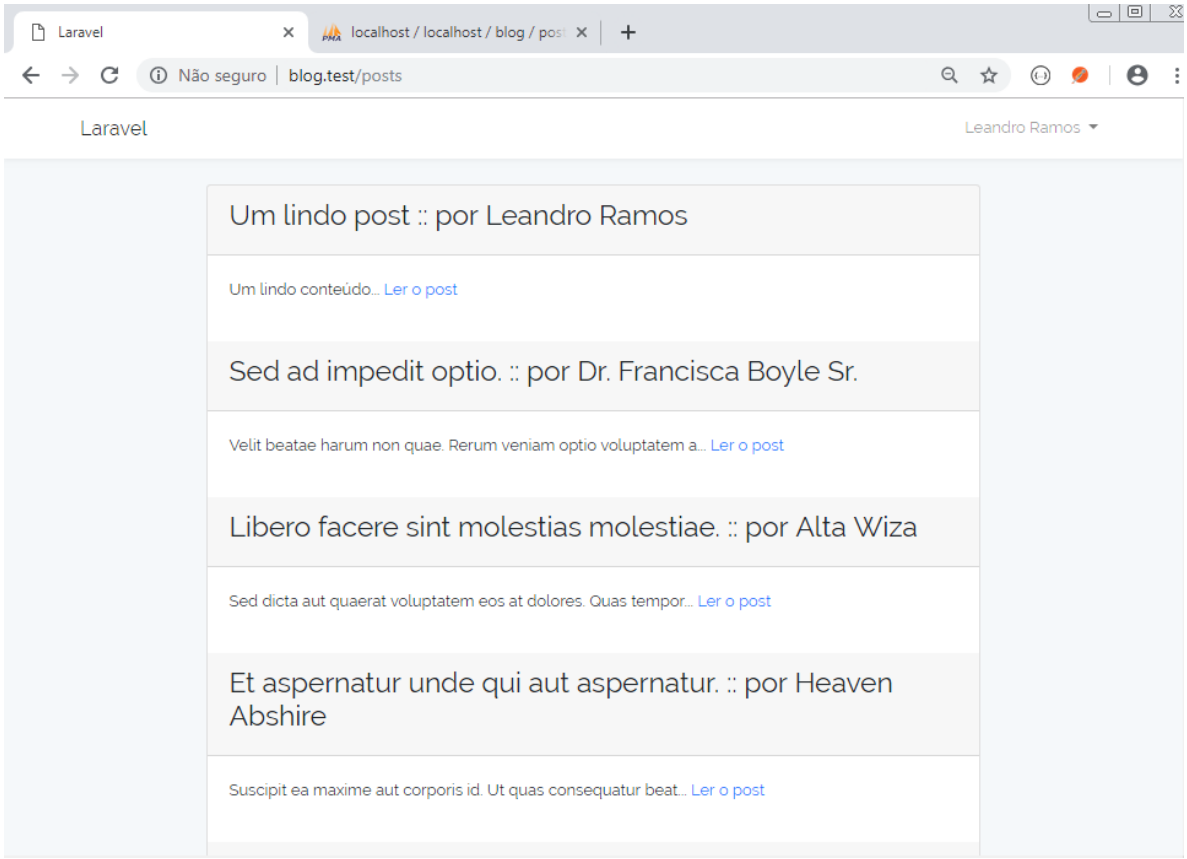
Vai para a tela de login

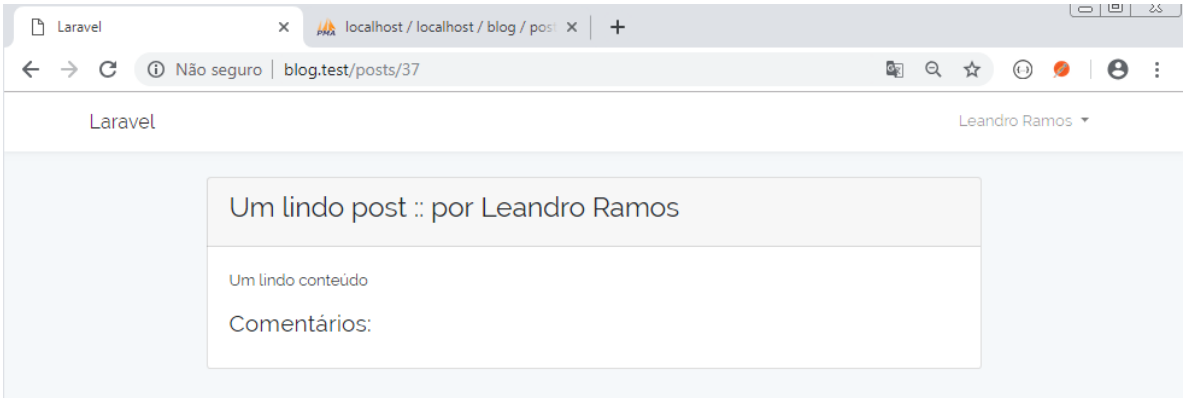


Criando um novo post

blog.test/posts/create

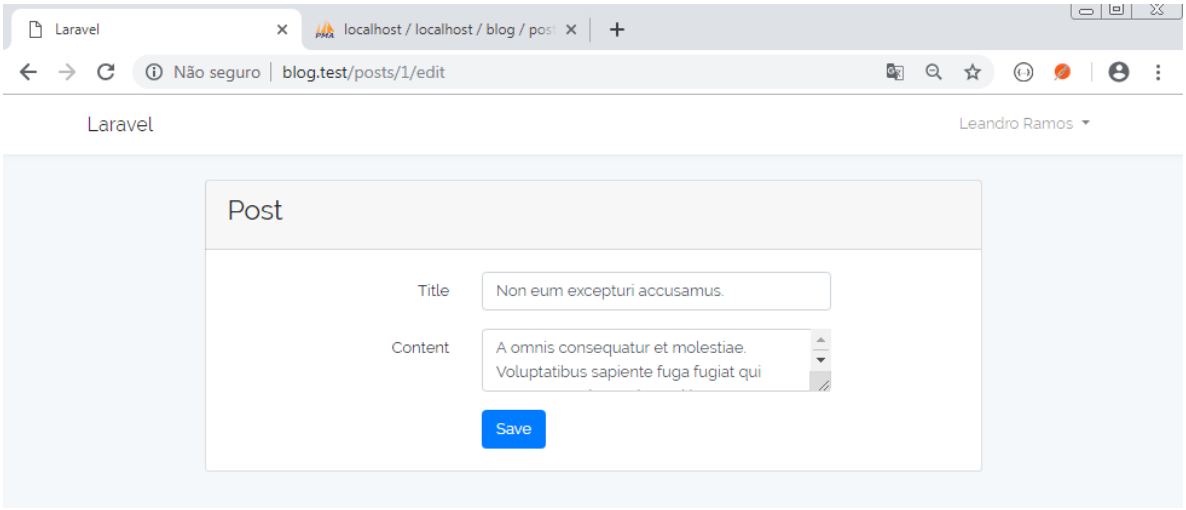






Editando um post

Editando o post 1



Aula 09 - Listando posts do autor

Alterar os seeders do banco de dados

database/seeds/DatabaseSeeder.php

```
<?php

use Illuminate\Database\Seeder;

class DatabaseSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        echo "Creating 10 users...\n";
        factory(App\User::class, 10)->create();
        echo "Creating 10 authors...\n";
        factory(App\Author::class, 10)->create();
        echo "Creating 36 posts related to random authors...\n";
        factory(App\Post::class, 36)->create();
        echo "Creating 67 comments related to random posts...\n";
        factory(App\Comment::class, 67)->create();
    }
}
```

database/factories/AuthorFactory.php

```
<?php

use Faker\Generator as Faker;

$factory->define(App\Author::class, function (Faker $faker) {
    return [
        'bio'   => $faker->paragraph(1),
        'user_id' => $faker->unique()->numberBetween(1, 10)
    ];
});
```

"Zerar"o banco de dados e popular com o novo seeder

php artisan migrate:fresh

```
C:\laragon\www\blog
λ php artisan migrate:fresh
Dropped all tables successfully.
Migration table created successfully.
Migrating: 2014_10_12_000000_create_users_table
Migrated:  2014_10_12_000000_create_users_table
Migrating: 2014_10_12_100000_create_password_resets_table
Migrated:  2014_10_12_100000_create_password_resets_table
Migrating: 2019_02_22_042912_create_posts_table
Migrated:  2019_02_22_042912_create_posts_table
Migrating: 2019_02_22_043021_create_comments_table
Migrated:  2019_02_22_043021_create_comments_table
Migrating: 2019_02_22_044010_create_authors_table
Migrated:  2019_02_22_044010_create_authors_table
Migrating: 2019_02_22_044234_add_author_id_to_posts
Migrated:  2019_02_22_044234_add_author_id_to_posts
Migrating: 2019_02_22_211703_add_user_id_remove_name_email_from_authors
Migrated:  2019_02_22_211703_add_user_id_remove_name_email_from_authors

C:\laragon\www\blog
λ
```

php artisan db:seed

```
C:\laragon\www\blog
λ php artisan db:seed
Creating 10 users...
Creating 10 authors...
Creating 36 posts related to random authors...
Creating 67 comments related to random posts...

C:\laragon\www\blog
λ
```

Criar a rota para a view dos posts do autor

routes/web.php

```
Route::resources([
   'authors' => 'AuthorController',
   'posts' => 'PostController'
]);


Auth::routes();

Route::get('authors/{author}/posts', 'AuthorController@posts');

Route::get('/home', 'HomeController@index')->name('home');
```

Criar o método que lista os posts do autor

app/Http/Controllers/AuthorController.php

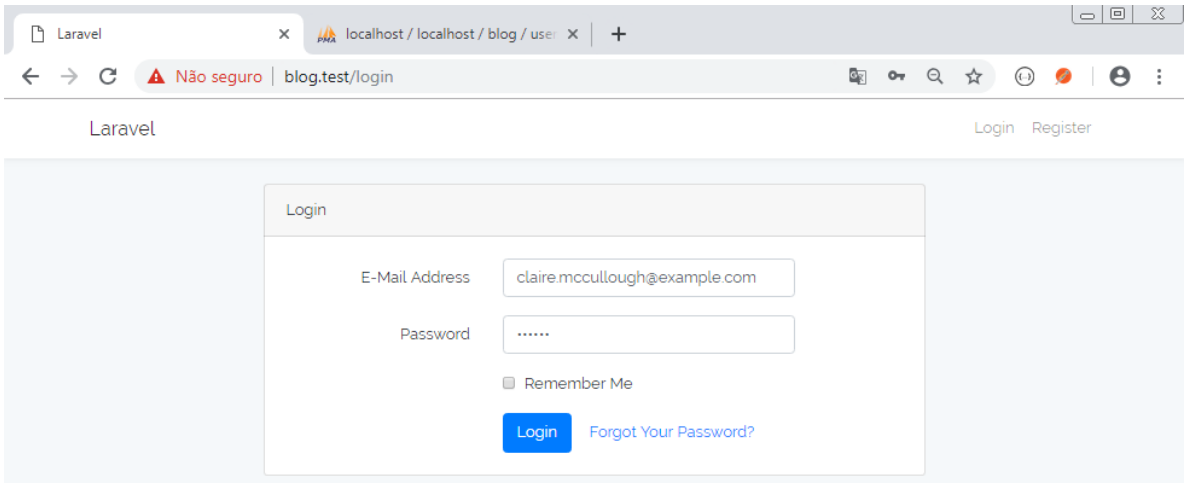
```
public function posts(Author $author)
{
   $posts = $author->posts;
   return view('authors.posts', compact('posts'));
}
```

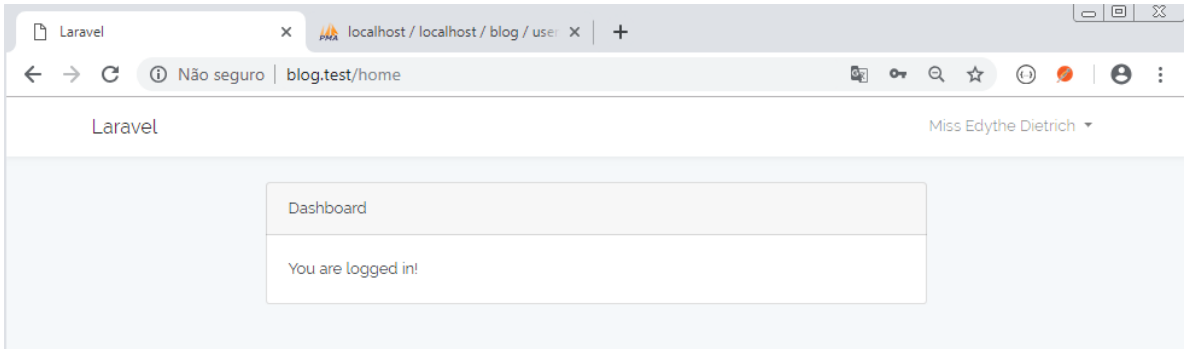
Criar a view que lista os posts do autor

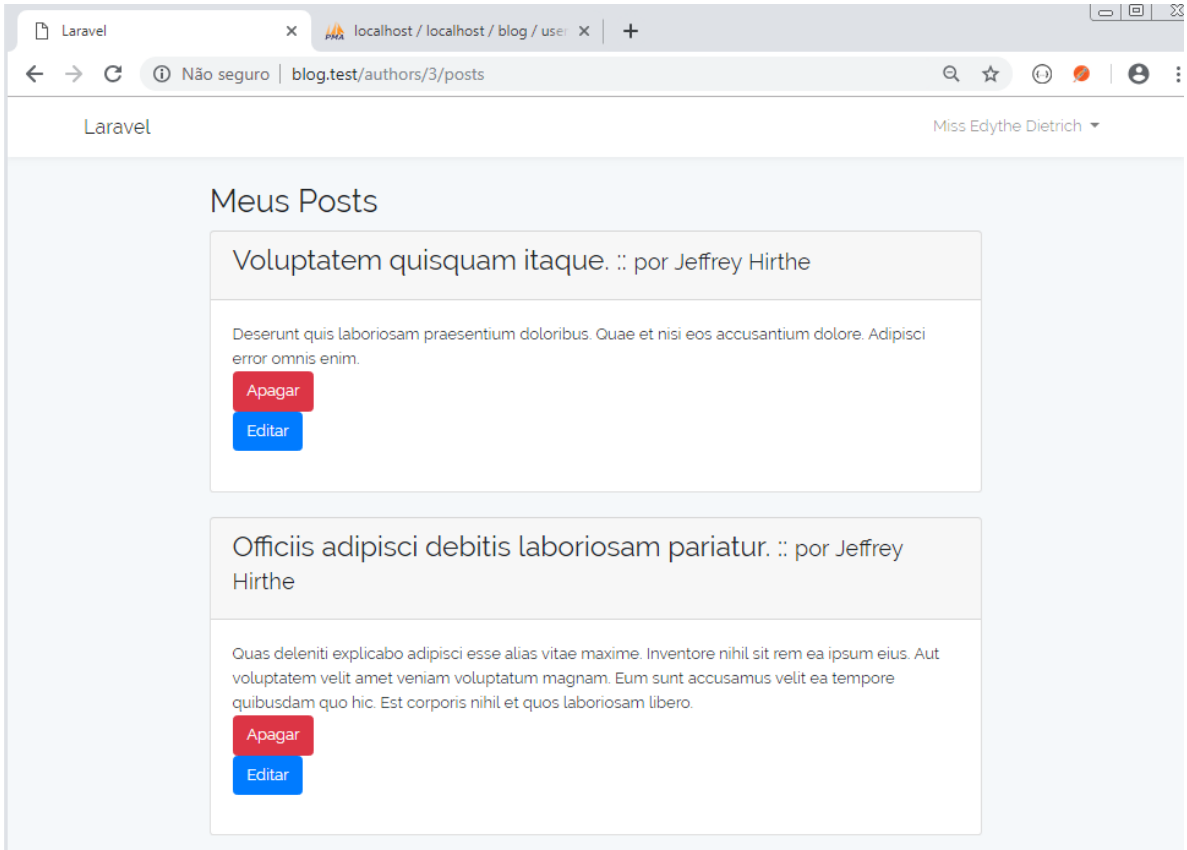
resources/views/authors/posts.blade.php

```
@extends('layouts.app')
@section('content')
  <div class="row justify-content-center">
    <div class="col-md-8">
      <h2>Meus Posts</h2>
      <div class="flash-message">
        @foreach (['danger', 'warning', 'success', 'info'] as $msg)
          @if(Session::has('alert-' . $msg))
          <p class="alert alert-{{ $msg }}">{{ Session::get('alert-' . $msg) }} <a href="#" class="close" data-dismiss="alert" aria-label="fechar">&times;</a></p>
          @endif
        @endforeach
      </div>
      @foreach ($posts as $post)
        <div class="card card-default">
          <div class="card-header">
            <h3>{{ $post->title }} :: <small>por {{ $post->author->user->name }}</small></h3>
          </div>
          <div class="card-body">
            <p>
              {{ $post->content }} <br>
              <a class="btn btn-danger" href="{{ action('PostController@destroy', $post->id) }}" title="Apagar o post">Apagar</a><br>
              <a class="btn btn-primary" href="{{ action('PostController@edit', $post->id) }}" title="Editar o post">Editar</a><br>
            </p>
          </div>
        </div>
        <br>
      @endforeach
    </div>
  </div>
@endsection
```

A senha de todos os usuários do banco é: secret

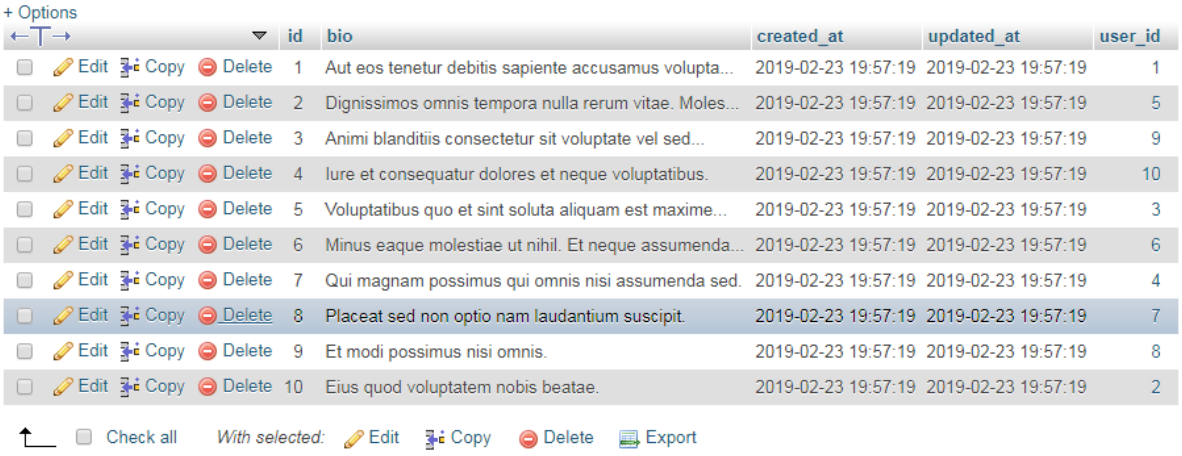






Perceba que o autor 3 é o usuário de id 9

Tabela authors

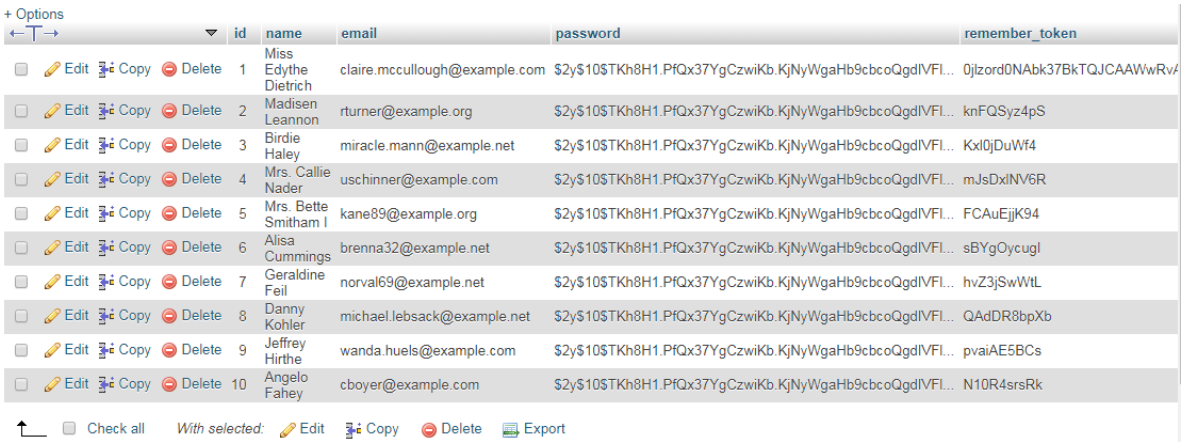


+ Options

	id	bio	created_at	updated_at	user_id
Edit Copy Delete	1	Aut eos tenetur debitis sapiente accusamus volupta...	2019-02-23 19:57:19	2019-02-23 19:57:19	1
Edit Copy Delete	2	Dignissimos omnis tempora nulla rerum vitae. Moles...	2019-02-23 19:57:19	2019-02-23 19:57:19	5
Edit Copy Delete	3	Animi blanditiis consectetur sit voluptate vel sed...	2019-02-23 19:57:19	2019-02-23 19:57:19	9
Edit Copy Delete	4	Iure et consequatur dolores et neque voluptatibus.	2019-02-23 19:57:19	2019-02-23 19:57:19	10
Edit Copy Delete	5	Voluptatibus quo et sint soluta aliquam est maxime...	2019-02-23 19:57:19	2019-02-23 19:57:19	3
Edit Copy Delete	6	Minus eaque molestiae ut nihil. Et neque assumenda...	2019-02-23 19:57:19	2019-02-23 19:57:19	6
Edit Copy Delete	7	Qui magnam possimus qui omnis nisi assumenda sed.	2019-02-23 19:57:19	2019-02-23 19:57:19	4
Edit Copy Delete	8	Placeat sed non optio nam laudantium suscipit.	2019-02-23 19:57:19	2019-02-23 19:57:19	7
Edit Copy Delete	9	Et modi possimus nisi omnis.	2019-02-23 19:57:19	2019-02-23 19:57:19	8
Edit Copy Delete	10	Eius quod voluptatem nobis beatae.	2019-02-23 19:57:19	2019-02-23 19:57:19	2

Check all With selected: Edit Copy Delete Export

Tabela users



+ Options

	id	name	email	password	remember_token
Edit Copy Delete	1	Miss Edythe Dietrich	claire.mccullough@example.com	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	0jlzord0NAbk37BkTQJCAAWwRv
Edit Copy Delete	2	Madisen Leannon	rturner@example.org	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	knFQSyz4pS
Edit Copy Delete	3	Birdie Haley	miracle.mann@example.net	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	Kxl0jDuWf4
Edit Copy Delete	4	Mrs. Callie Nader	uschinner@example.com	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	mJsDxlNV6R
Edit Copy Delete	5	Mrs. Bette Smitham I	kane89@example.org	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	FCAuEjjK94
Edit Copy Delete	6	Alisa Cummings	brenna32@example.net	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	sBYgOycugI
Edit Copy Delete	7	Geraldine Feil	norval69@example.net	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	hvZ3jSwWtL
Edit Copy Delete	8	Danny Kohler	michael.lebsack@example.net	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	QAdDR8bpXb
Edit Copy Delete	9	Jeffrey Hirthe	wanda.huels@example.com	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	pvaiAE5BCs
Edit Copy Delete	10	Angelo Fahey	cboyer@example.com	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	N10R4srsRk

Check all With selected: Edit Copy Delete Export

OBS.: As rotinas para apagar e editar os posts ainda não estão implementadas.

Aula 10 - Apagando posts do autor

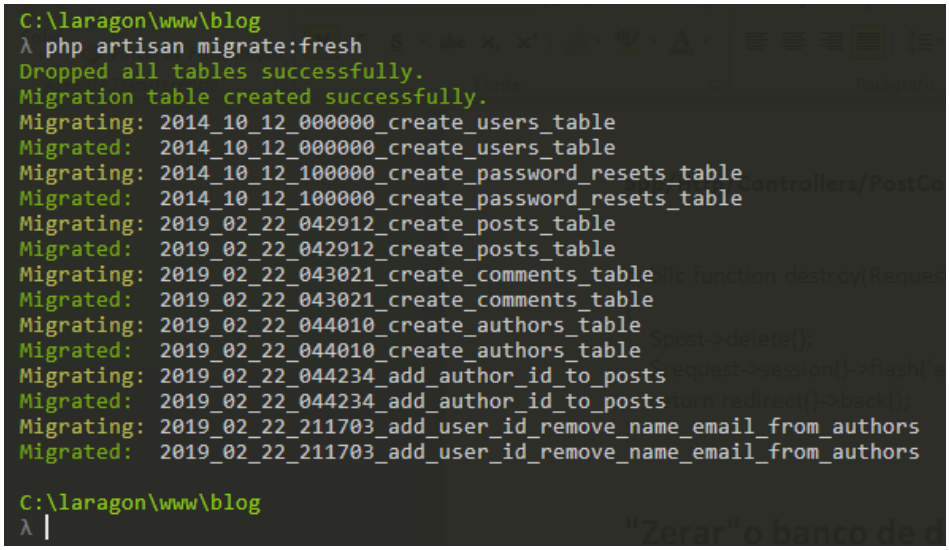
Implementar método destroy no PostController

app/Http/Controllers/PostController.php

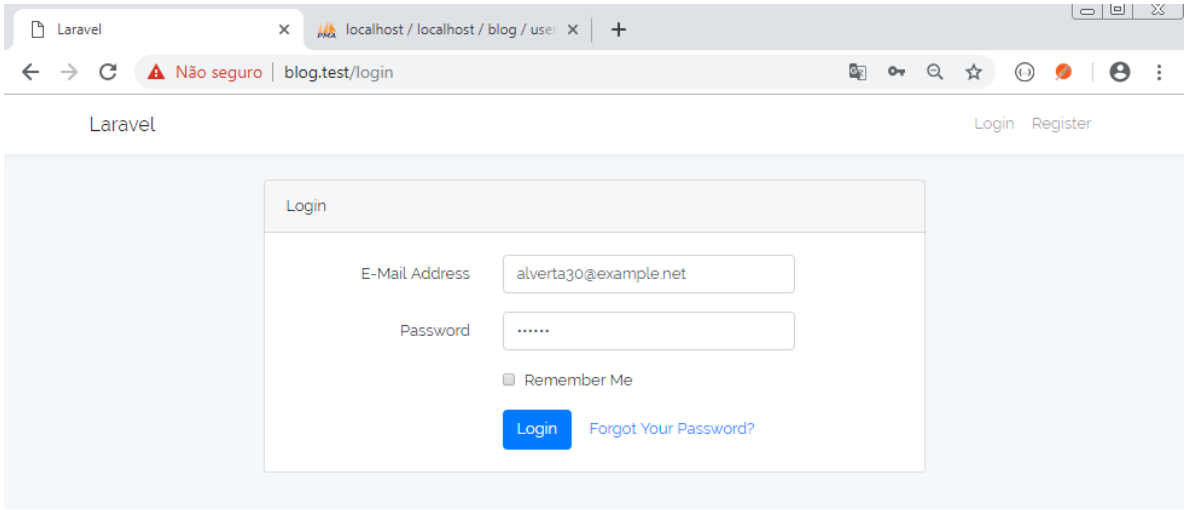
```
public function destroy(Request $request, Post $post)
{
   $post->delete();
   $request->session()->flash('alert-success', 'Post apagado com sucesso!');
   return redirect()->back();
}
```

"Zerar"o banco de dados e popular com o novo seeder

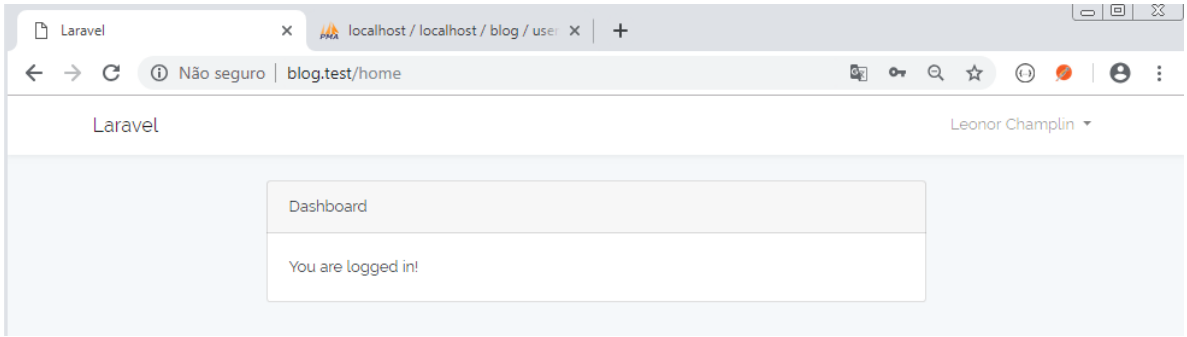
php artisan migrate:fresh

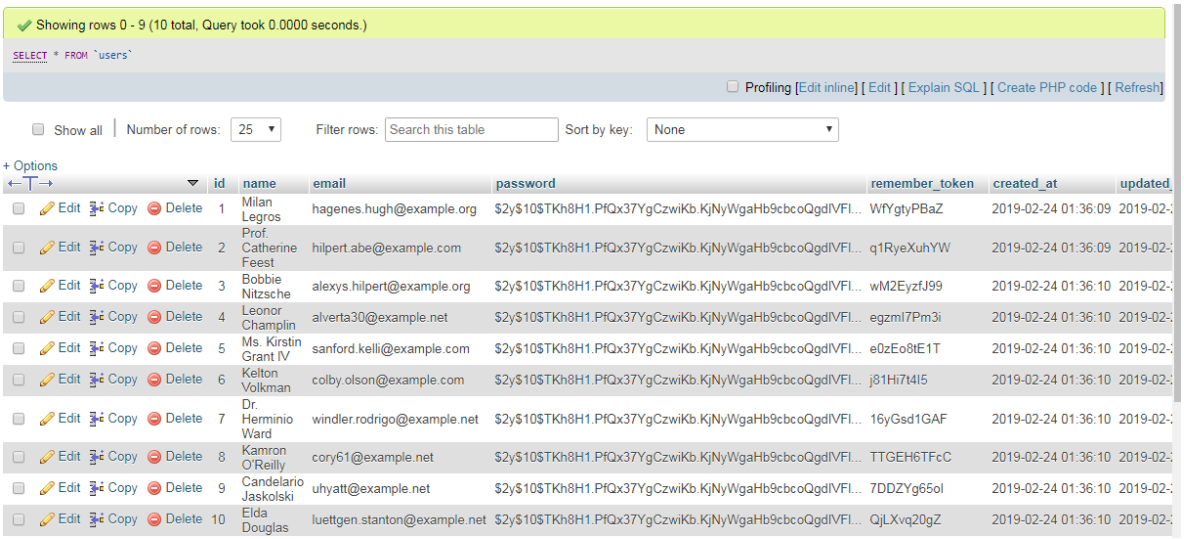


php artisan db:seed

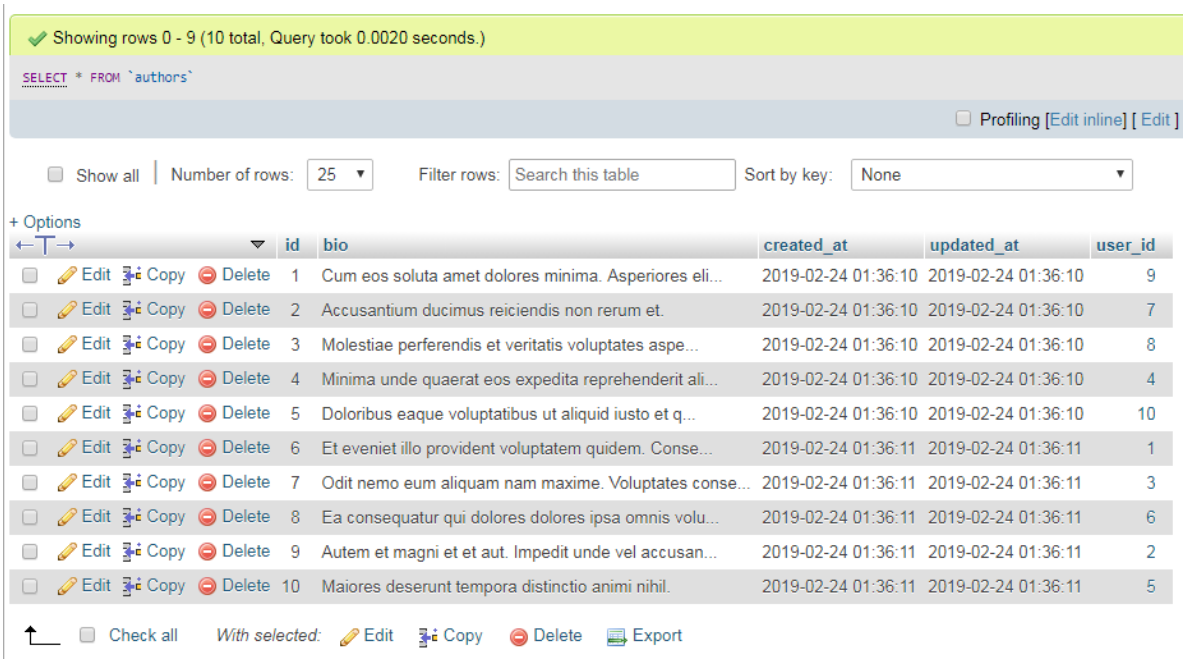


A senha é: secret





	id	name	email	password	remember_token	created_at	updated
Edit Copy Delete	1	Milan Legros	hagenes.hugh@example.org	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	WfYgtyPBaZ	2019-02-24 01:36:09	2019-02-...
Edit Copy Delete	2	Prof. Catherine Feest	hilpert.abe@example.com	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	q1RyeXuhYW	2019-02-24 01:36:09	2019-02-...
Edit Copy Delete	3	Bobbie Nitzsche	alexys.hilpert@example.org	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	wM2EyzfJ99	2019-02-24 01:36:10	2019-02-...
Edit Copy Delete	4	Leonor Champlin	alverta30@example.net	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	egzml7Pm3i	2019-02-24 01:36:10	2019-02-...
Edit Copy Delete	5	Ms. Kirstin Grant IV	sanford.kelli@example.com	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	e0zEo8tE1T	2019-02-24 01:36:10	2019-02-...
Edit Copy Delete	6	Kelton Volkman	colby.olson@example.com	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	j81Hi7t4l5	2019-02-24 01:36:10	2019-02-...
Edit Copy Delete	7	Dr. Herminio Ward	windler.rodrigo@example.net	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	16yGsd1GAF	2019-02-24 01:36:10	2019-02-...
Edit Copy Delete	8	Kamron O'Reilly	cory61@example.net	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	TTGEH6TFcC	2019-02-24 01:36:10	2019-02-...
Edit Copy Delete	9	Candelario Jaskolski	uhyatt@example.net	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	7DDZYg65ol	2019-02-24 01:36:10	2019-02-...
Edit Copy Delete	10	Elda Douglas	luettgen.stanton@example.net	$2y$10$TKh8H1.PfQx37YgCzwiKb.KjNyWgaHb9cbcoQgdIVFl...	QjLXvq20gZ	2019-02-24 01:36:10	2019-02-...



Showing rows 0 - 9 (10 total, Query took 0.0020 seconds.)

```
SELECT * FROM `authors`
```

Profiling [Edit inline] [Edit]

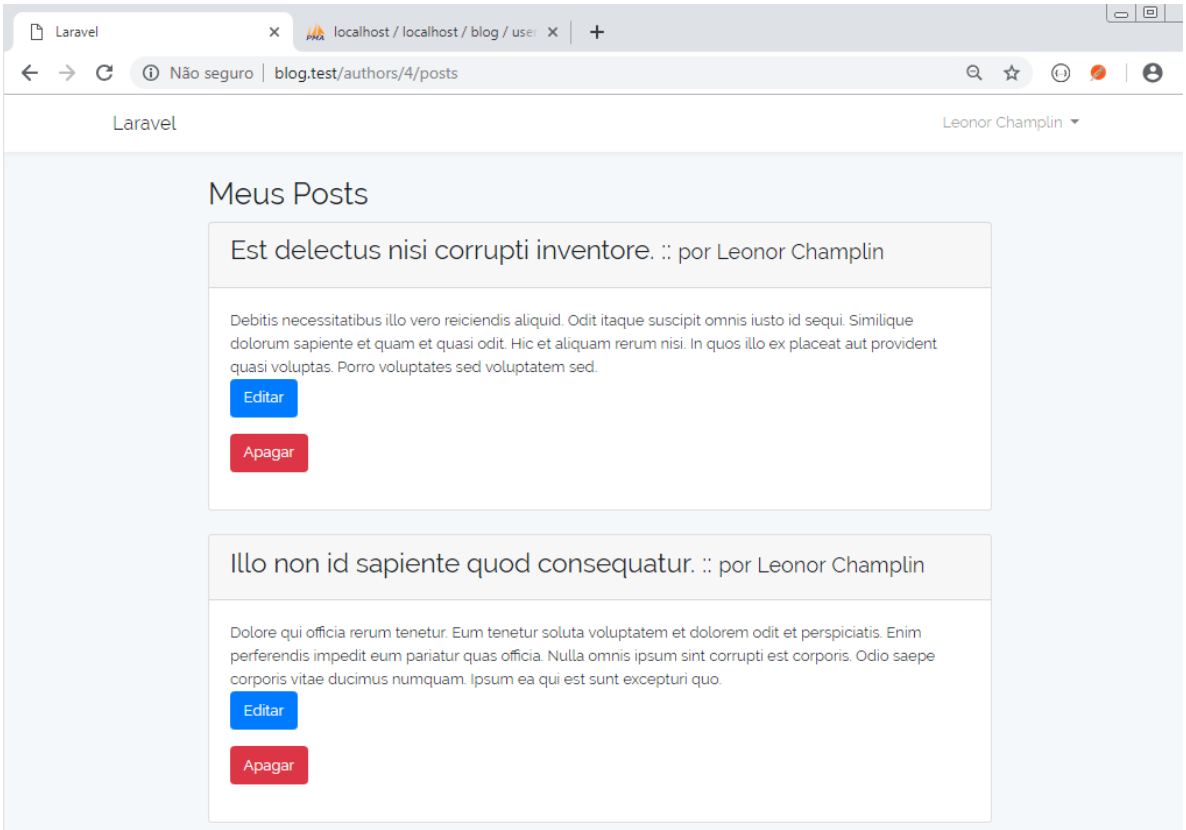
Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

+ Options

	id	bio	created_at	updated_at	user_id
Edit Copy Delete	1	Cum eos soluta amet dolores minima. Asperiores eli...	2019-02-24 01:36:10	2019-02-24 01:36:10	9
Edit Copy Delete	2	Accusantium ducimus reiciendis non rerum et.	2019-02-24 01:36:10	2019-02-24 01:36:10	7
Edit Copy Delete	3	Molestiae perferendis et veritatis voluptates aspe...	2019-02-24 01:36:10	2019-02-24 01:36:10	8
Edit Copy Delete	4	Minima unde quaerat eos expedita reprehenderit ali...	2019-02-24 01:36:10	2019-02-24 01:36:10	4
Edit Copy Delete	5	Doloribus eaque voluptatibus ut aliquid iusto et q...	2019-02-24 01:36:10	2019-02-24 01:36:10	10
Edit Copy Delete	6	Et eveniet illo provident voluptatem quidem. Conse...	2019-02-24 01:36:11	2019-02-24 01:36:11	1
Edit Copy Delete	7	Odit nemo eum aliquam nam maxime. Voluptates conse...	2019-02-24 01:36:11	2019-02-24 01:36:11	3
Edit Copy Delete	8	Ea consequatur qui dolores dolores ipsa omnis volu...	2019-02-24 01:36:11	2019-02-24 01:36:11	6
Edit Copy Delete	9	Autem et magni et et aut. Impedit unde vel accusan...	2019-02-24 01:36:11	2019-02-24 01:36:11	2
Edit Copy Delete	10	Maiores deserunt tempora distinctio animi nihil.	2019-02-24 01:36:11	2019-02-24 01:36:11	5

Check all *With selected:* Edit Copy Delete Export

O id do autor é 4



Ajustar a migration para que possamos apagar o post e seus comentários

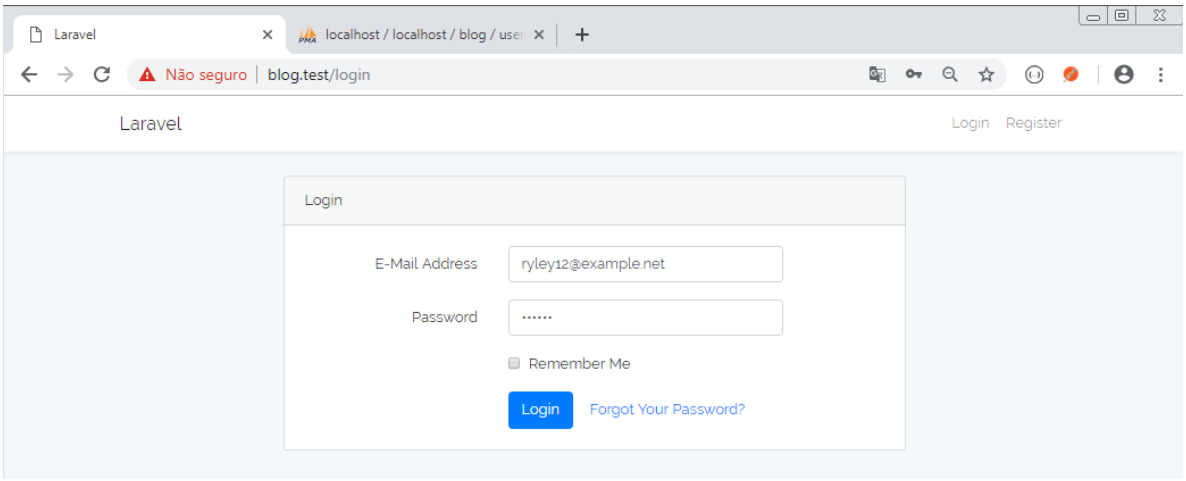
Temos que incluir o "onDelete('cascade')", para que os comentários sejam apagados junto com o post.

database/migrations/2019_02_22_043021_create_comments_table.php

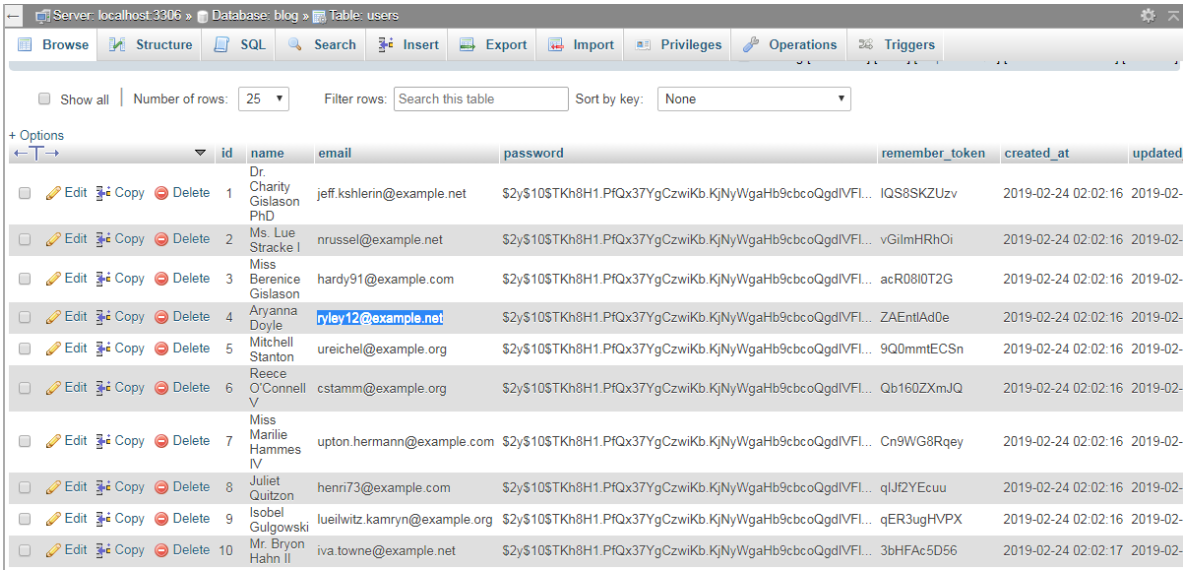
```
<?php

use Illuminate\Support\Facades\Schema;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Database\Migrations\Migration;

class CreateCommentsTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('comments', function (Blueprint $table) {
            $table->increments('id');
            $table->string('author_email');
            $table->text('content');
            $table->integer('post_id')->unsigned();
            $table->timestamps();

            $table->foreign('post_id')->references('id')->on('posts')->onDelete('cascade');
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('comments');
    }
}
```

Zerar novamente o banco de dados e em seguida popular novamente.

```
php artisan migrate:fresh
php artisan db:seed
```



A senha é: secret

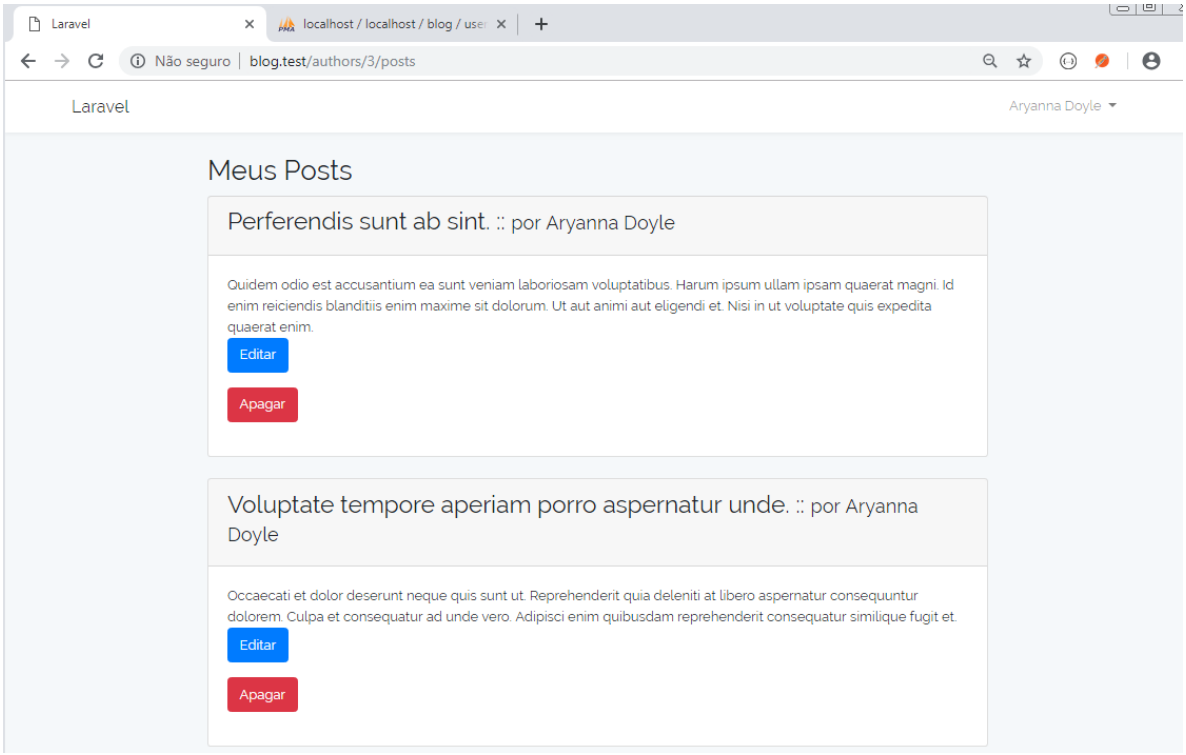


+ Options

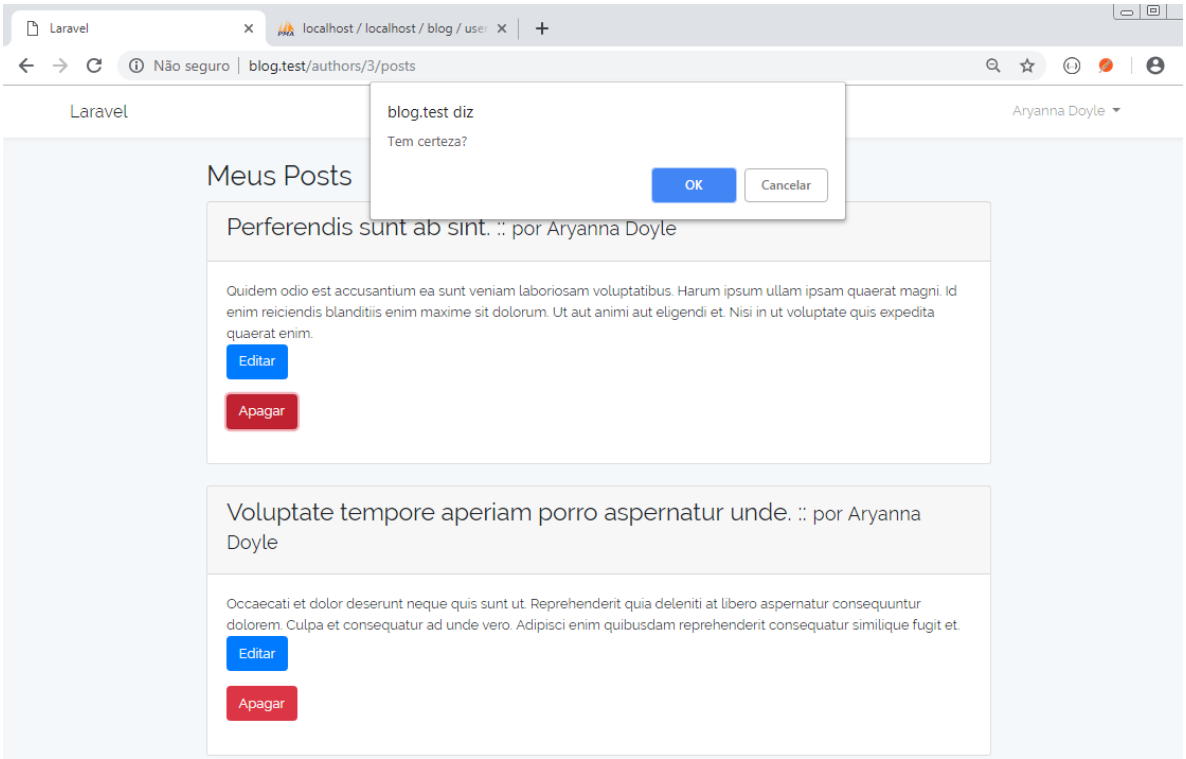
	id	bio	created_at	updated_at	user_id
Edit Copy Delete	1	Quasi soluta id eos numquam odit ut. Repellat ex a...	2019-02-24 02:02:17	2019-02-24 02:02:17	1
Edit Copy Delete	2	Maxime quo aliquid dolores ratione. Perferendis ra...	2019-02-24 02:02:17	2019-02-24 02:02:17	3
Edit Copy Delete	3	Libero nihil nisi cupiditate dolores.	2019-02-24 02:02:17	2019-02-24 02:02:17	4
Edit Copy Delete	4	Nulla iste ipsa exercitationem et ab non.	2019-02-24 02:02:17	2019-02-24 02:02:17	5
Edit Copy Delete	5	Nulla sunt amet velit impedit a.	2019-02-24 02:02:17	2019-02-24 02:02:17	9
Edit Copy Delete	6	Recusandae quia sit laborum consequuntur vel. Illo...	2019-02-24 02:02:17	2019-02-24 02:02:17	2
Edit Copy Delete	7	Reprehenderit est in quisquam autem a molestiae na...	2019-02-24 02:02:17	2019-02-24 02:02:17	8
Edit Copy Delete	8	Assumenda autem nostrum saepe quia eos. In non qui...	2019-02-24 02:02:17	2019-02-24 02:02:17	7
Edit Copy Delete	9	Reiciendis error eum est quia fuga sed optio.	2019-02-24 02:02:17	2019-02-24 02:02:17	10
Edit Copy Delete	10	Eos qui dolores sed impedit sunt.	2019-02-24 02:02:17	2019-02-24 02:02:17	6

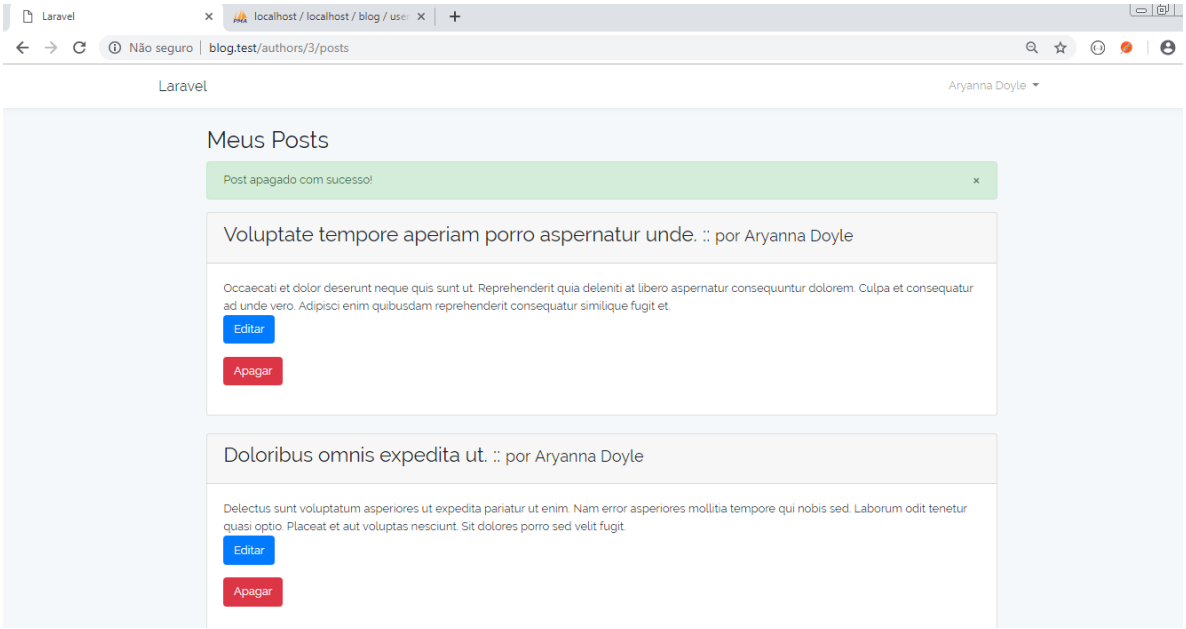
O id do autor é 3

http://blog.test/authors/3/posts



Apagando o primeiro post

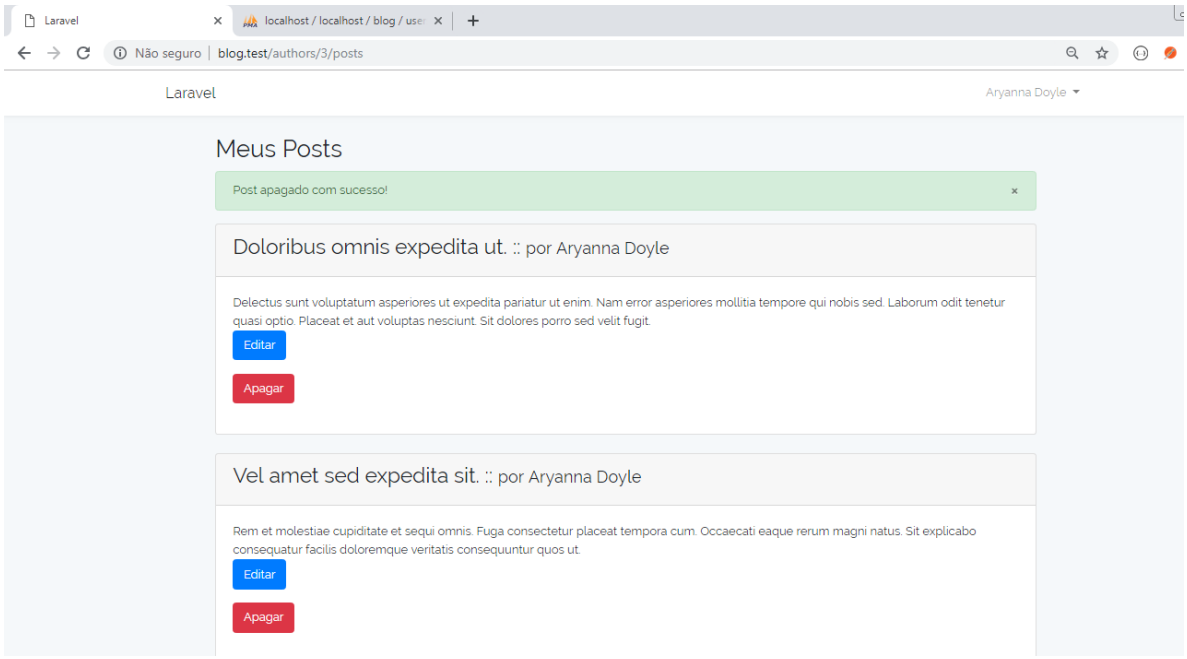



Laravel
localhost / localhost / blog / use
Não seguro | blog.test/authors/3/posts
Laravel
Aryanna Doyle
Meus Posts
Post apagado com sucesso!
Voluptate tempore aperiam porro aspernatur unde. :: por Aryanna Doyle
Occaecati et dolor deserunt neque quis sunt ut. Reprehenderit quia deleniti at libero aspernatur consequuntur dolorem. Culpa et consequatur ad unde vero. Adipisci enim quibusdam reprehenderit consequatur similique fugit et.
Editar
Apagar
Doloribus omnis expedita ut. :: por Aryanna Doyle
Delectus sunt voluptatum asperiores ut expedita pariatur ut enim. Nam error asperiores mollitia tempore qui nobis sed. Laborum odit tenetur quasi optio. Placeat et aut voluptas nesciunt. Sit dolores porro sed velit fugit.
Editar
Apagar

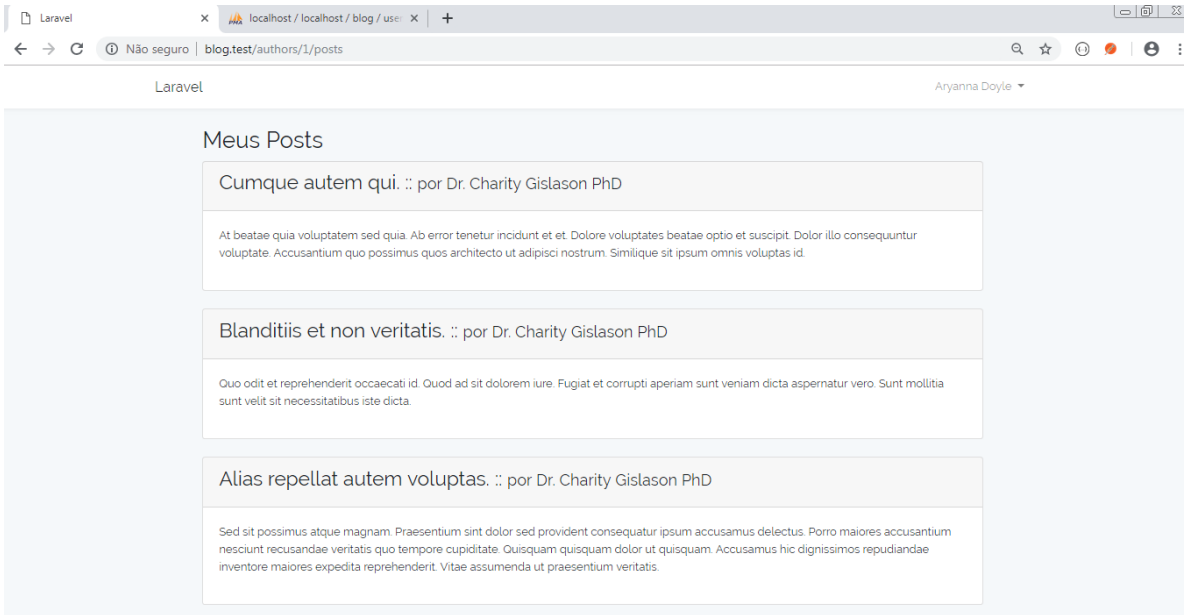
resources/views/authors/post.blade.php

```
@extends('layouts.app')
@section('content')
  <div class="row justify-content-center">
    <div class="col-md-8">
      <h2>Meus Posts</h2>
      <div class="flash-message">
        @foreach (['danger', 'warning', 'success', 'info'] as $msg)
          @if(Session::has('alert-' . $msg))

          <p class="alert alert-{{ $msg }}">{{ Session::get('alert-' . $msg) }} <a href="#" class="close" data-
dismiss="alert" aria-label="fechar">&times;</a></p>
          @endif
        @endforeach
      </div>
      @foreach ($posts as $post)
        <div class="card card-default">
          <div class="card-header">
            <h3>{{ $post->title }} :: <small>por {{ $post->author->user->name }}</small></h3>
          </div>
          <div class="card-body">
            <p>
              {{ $post->content }} <br>
              @if(Auth::check() && Auth::user()->author->id==$post->author_id)
                <a class="btn btn-primary" href="{{ action('PostController@edit', $post->id) }}" title="Editar o
post">Editar</a><br>
                <form method="post" action="{{ action('PostController@destroy', $post->id) }}">
                  {{ csrf_field() }}
                  {{ method_field('delete') }}
                  <button type="submit" class="btn btn-danger delete-button" onclick="return confirm('Tem
certeza?');")>Apagar</button>
                </form>
              @endif
            </p>
          </div>
        </div>
        <br>
      @endforeach
    </div>
  </div>
@endsection
```

Apague o primeiro post exibido



Entre em um outro id (exemplo: 1) para apagar e os botões não devem ser exibidos.



Alterar a view dos posts do autor

resources/views/authors/posts

```
@extends('layouts.app')
@section('content')
  <div class="row justify-content-center">
    <div class="col-md-8">
      <h2>Meus Posts</h2>
      <div class="flash-message">
        @foreach (['danger', 'warning', 'success', 'info'] as $msg)
          @if(Session::has('alert-' . $msg))

          <p class="alert alert-{{ $msg }}">{{ Session::get('alert-' . $msg) }} <a href="#" class="close" data-dismiss="alert" aria-label="fechar">&times;</a></p>
          @endif
        @endforeach
      </div>
      @foreach ($posts as $post)
        <div class="card card-default">
          <div class="card-header">
            <h3>{{ $post->title }} :: <small>por {{ $post->author->user->name }}</small></h3>
          </div>
          <div class="card-body">
            <p>
              {{ $post->content }} <br>
              <a class="btn btn-primary" href="{{ action('PostController@edit', $post->id) }}" title="Editar o post">Editar</a><br>
              <form method="post" action="{{ action('PostController@destroy', $post->id) }}">
                {{ csrf_field() }}
                {{ method_field('delete') }}
                <button type="submit" class="btn btn-danger delete-button" onclick="return confirm('Tem certeza?');")>Apagar</button>
              </form>
            </p>
          </div>
        </div>
        <br>
      @endforeach
    </div>
  </div>
@endsection
```

Aula 11 - Editando posts do autor

Implementar método update no PostController

app/Http/Controllers/PostController.php

```
public function update(Request $request, Post $post)
{
   $post->title  = $request->title;
   $post->content = $request->content;

   $post->save();
   $request->session()->flash('alert-success', 'Post alterado com sucesso!');
   return redirect("authors/$post->author_id/posts");
}
```

Alterar a rota para a home do autor

Adicionar o método login ao AuthorController

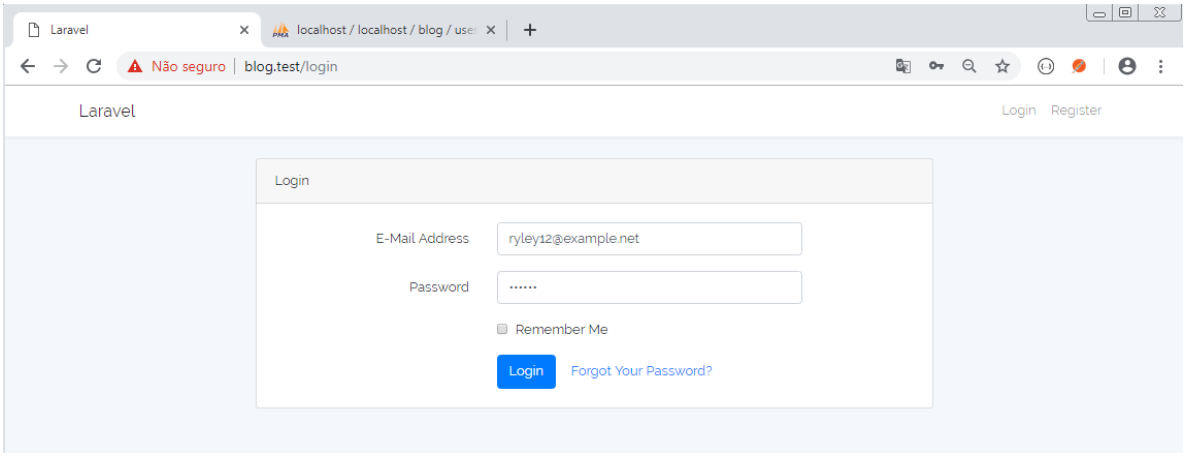
```
/**
 * Redirect logged author to posts
 *
 * @param \App\Author  $author
 * @return \Illuminate\Http\Response
 */
public function login()
{
   $author = \Auth::user()->author;
   $posts = $author->posts;
   return view('authors.posts', compact('posts'));
}
```

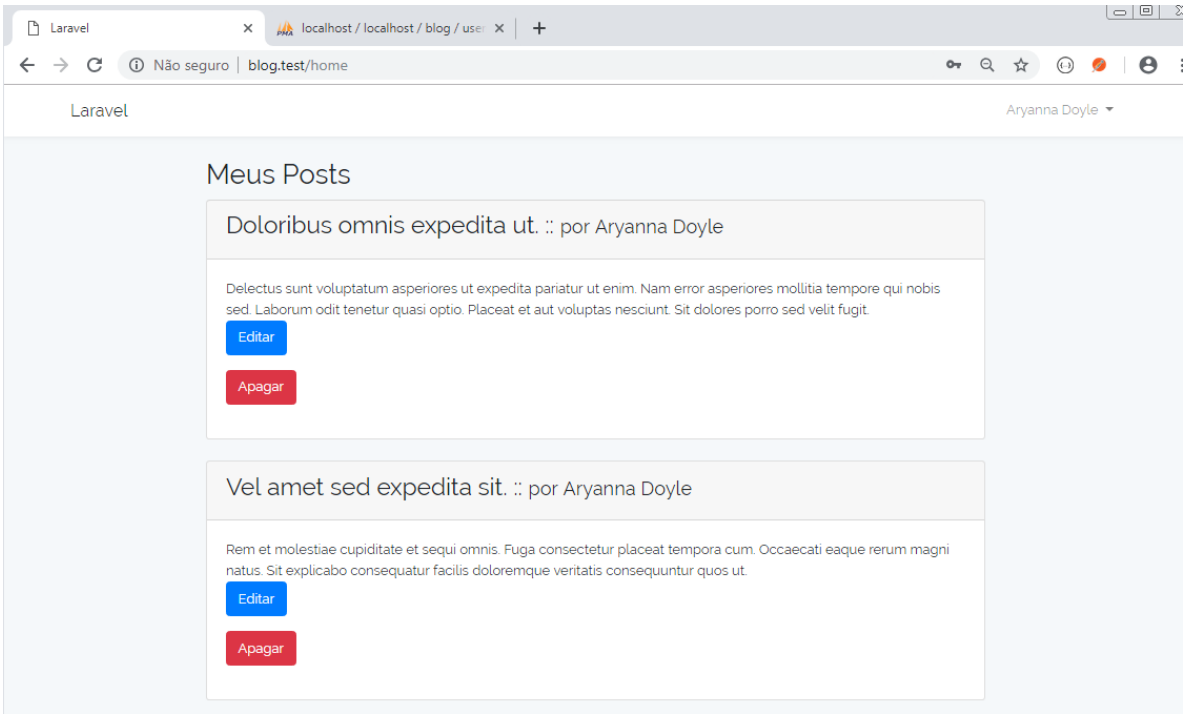
Alterar a rota /home para o autor logado

```
Route::get('/home', 'AuthorController@login');
```

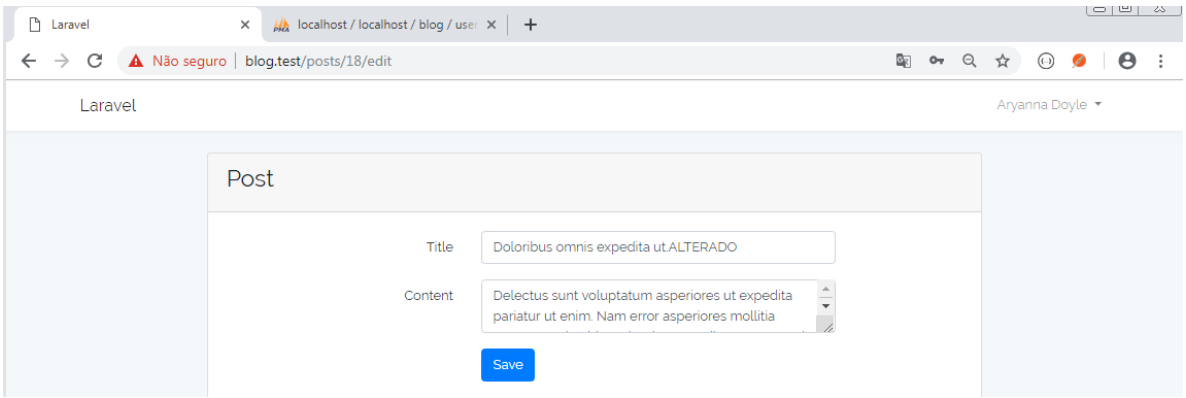
Alterar a rota / para a lista de todos os posts (sem login)

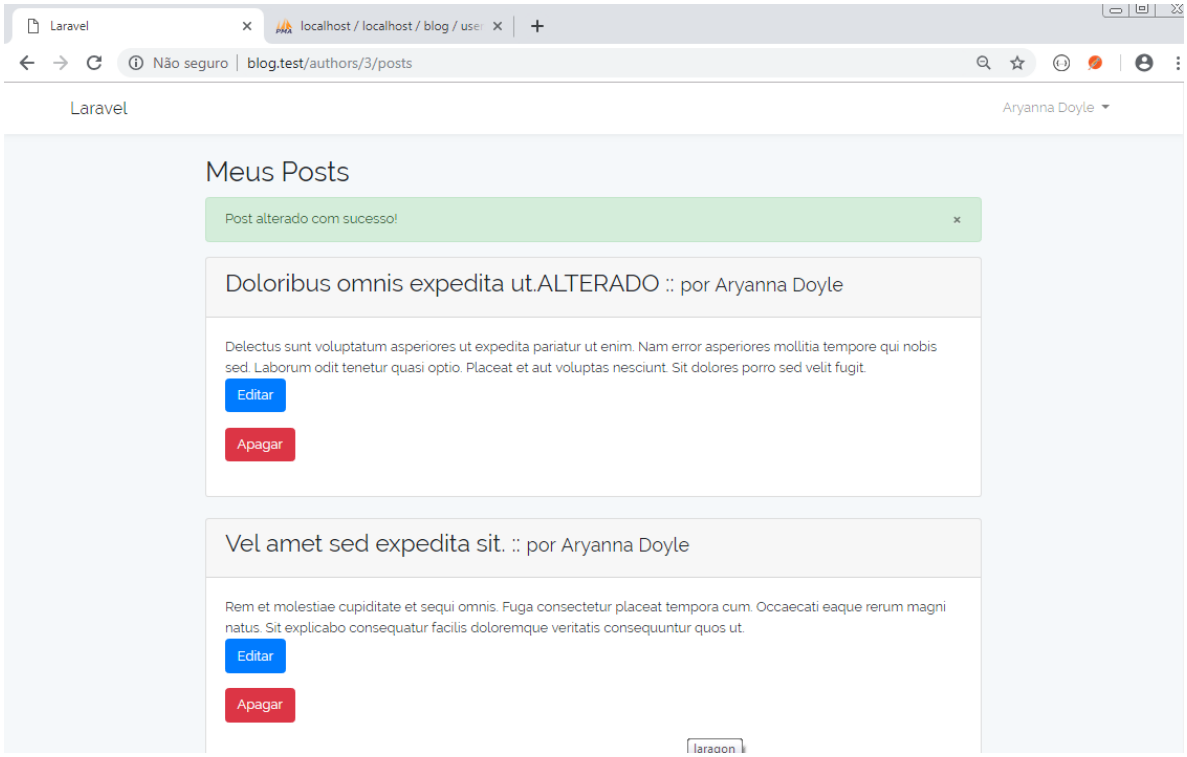
```
Route::get('/', 'PostController@index');
```



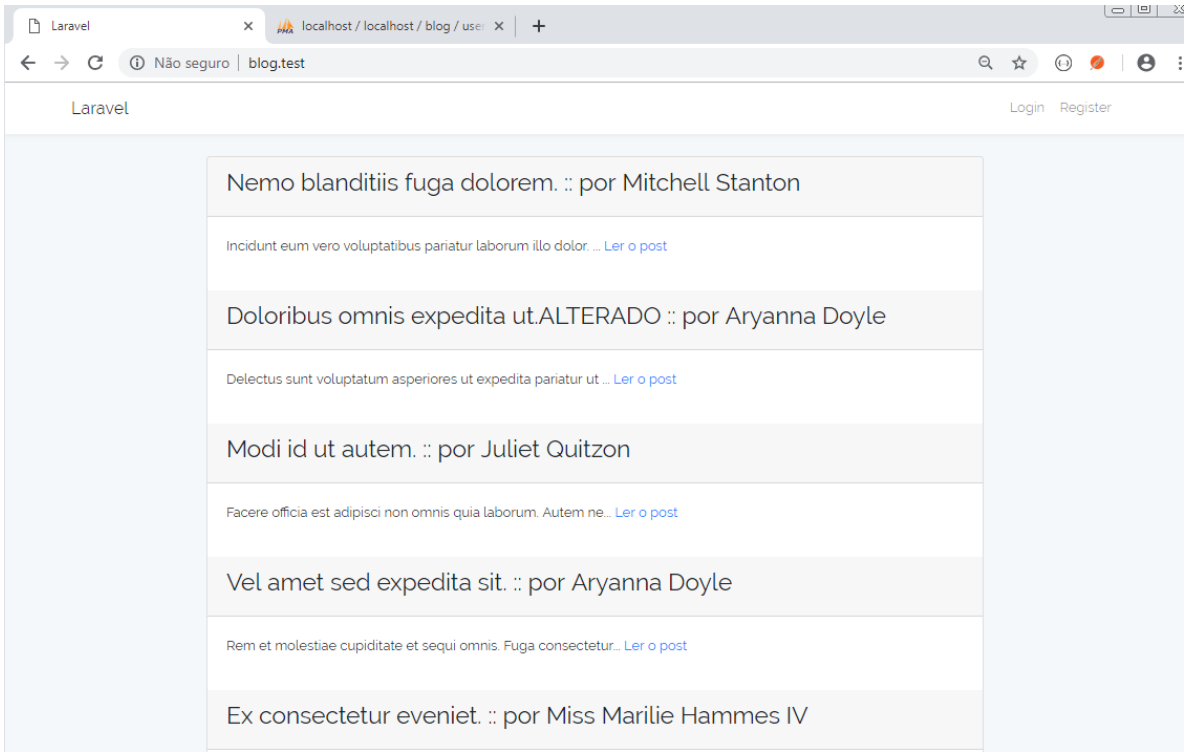


Editando o primeiro post:





Ao fazer logout:



Aula 12 - Deletando comentários

Implementar método destroy no CommentController

app/Http/Controllers/CommentController.php

```
public function destroy(Request $request, Comment $comment)
{
   $comment->delete();
   $request->session()->flash('alert-success', 'Comentário apagado com sucesso!');
   return redirect()->back();
}
```

Adicionar a rota para o método destroy do CommentController

routes/web.php

```
Route::delete('comments/{comment}', 'CommentController@destroy');
```

Adicionar link para mostrar o post e seus comentários na view de posts do autor

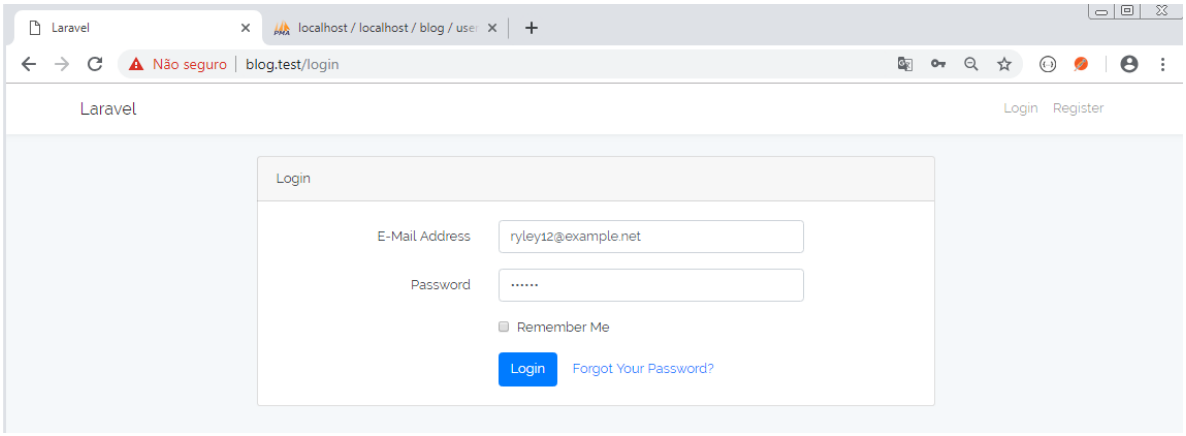
resources/views/authors/posts.blade.php

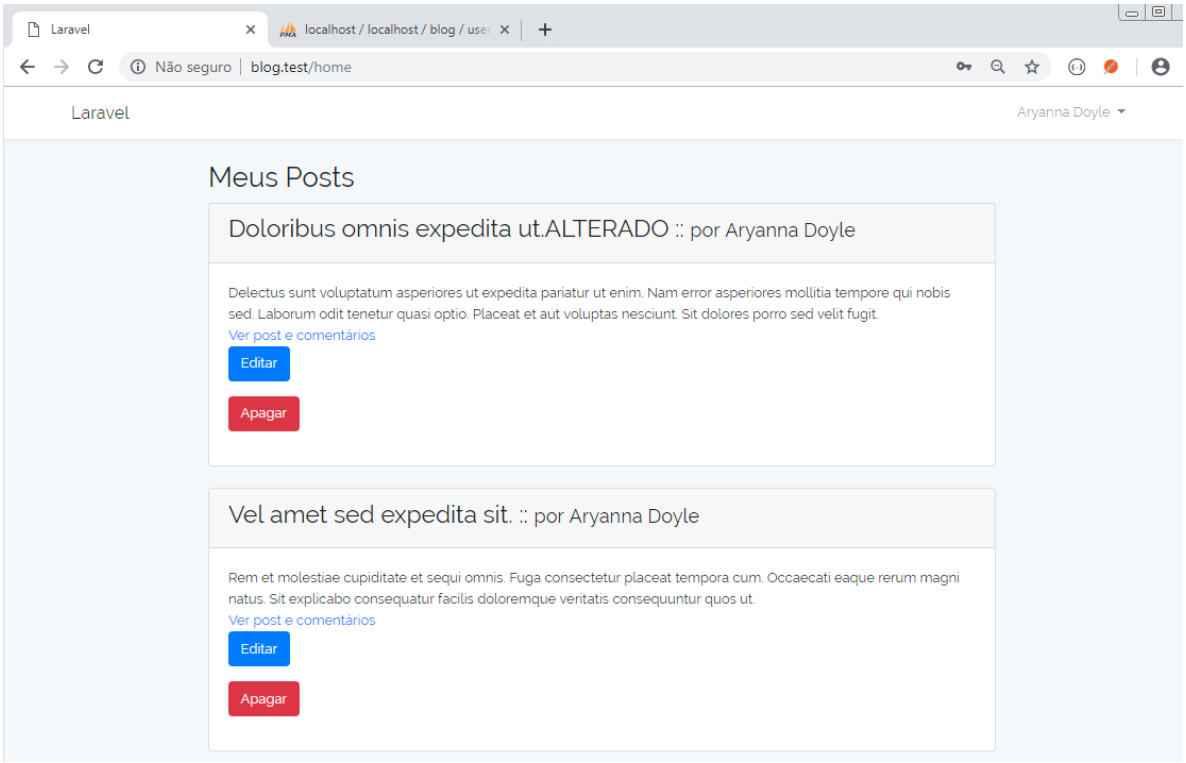
```
@extends('layouts.app')
@section('content')
  <div class="row justify-content-center">
    <div class="col-md-8">
      <h2>Meus Posts</h2>
      <div class="flash-message">
        @foreach (['danger', 'warning', 'success', 'info'] as $msg)
          @if(Session::has('alert-' . $msg))
          <p class="alert alert-{{ $msg }}">{{ Session::get('alert-' . $msg) }} <a href="#" class="close" data-dismiss="alert" aria-label="fechar">&times;</a></p>
          @endif
        @endforeach
      </div>
      @foreach ($posts as $post)
        <div class="card card-default">
          <div class="card-header">
            <h3>{{ $post->title }} :: <small>por {{ $post->author->user->name }}</small></h3>
          </div>
          <div class="card-body">
            <p>
              {{ $post->content }} <br>
              <a href="{{ action('PostController@show', $post->id) }}" title="Ver post e comentários">Ver post e comentários</a><br>
              <a class="btn btn-primary" href="{{ action('PostController@edit', $post->id) }}" title="Editar o post">Editar</a><br>
              <form method="post" action="{{ action('PostController@destroy', $post->id) }}">
                {{ csrf_field() }}
                {{ method_field('delete') }}
                <button type="submit" class="btn btn-danger delete-button" onclick="return confirm('Tem certeza?');")>Apagar</button>
              </form>
            </p>
          </div>
        </div>
        <br>
      @endforeach
    </div>
  </div>
@endsection
```

Alterar a view show do post para mostrar o botão deletar só para o autor do post

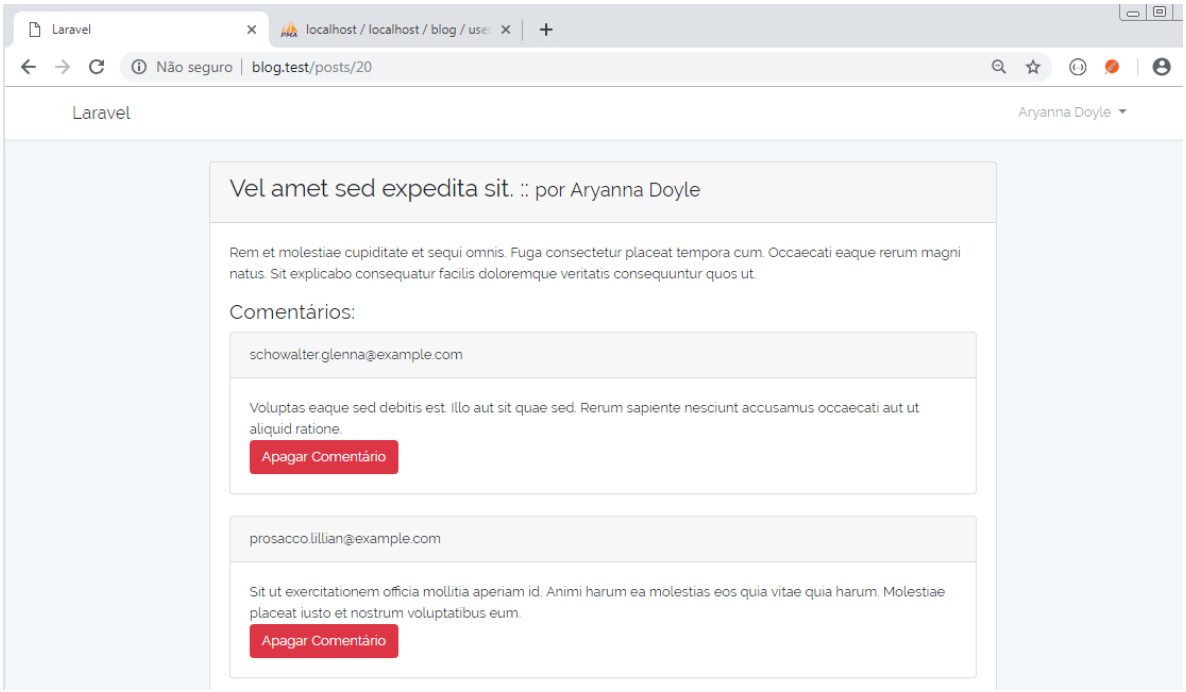
resources/views/posts/show.blade.php

```
@extends('layouts.app')
@section('content')
  <div class="row justify-content-center">
    <div class="col-md-8">
      <div class="card card-default">

        <div class="flash-message">
          @foreach (['danger', 'warning', 'success', 'info'] as $msg)
            @if(Session::has('alert-' . $msg))

            <p class="alert alert-{{ $msg }}">{{ Session::get('alert-' . $msg) }} <a href="#" class="close" data-dismiss="alert" aria-label="fechar">&times;</a></p>
            @endif
          @endforeach
        </div>

        <div class="card-header">
          <h3>{{ $post->title }} :: <small>por {{ $post->author->user->name }}</small></h3>
        </div>
        <div class="card-body">
          <p>
            {{ $post->content }}
          </p>
          <h4>Comentários:</h4>
          <!-- Comentários do post -->
          @foreach($post->comments as $comment)
            <div class="card card-default">
              <div class="card-header">
                {{ $comment->author_email }}
              </div>
              <div class="card-body">
                {{ $comment->content }}
                @if (Auth::check() && Auth::user()->author->id == $post->author_id)
                  <form method="post" action="{{ action('CommentController@destroy', $comment->id) }}">
                    {{ csrf_field() }}
                    {{ method_field('delete') }}
                    <button type="submit" class="btn btn-danger delete-button" onclick="return confirm('Tem certeza?');")>Apagar Comentário</button>
                  </form>
                @endif
              </div>
            </div>
            <br>
          @endforeach
        </div>
      </div>
    </div>
  </div>
@endsection
```

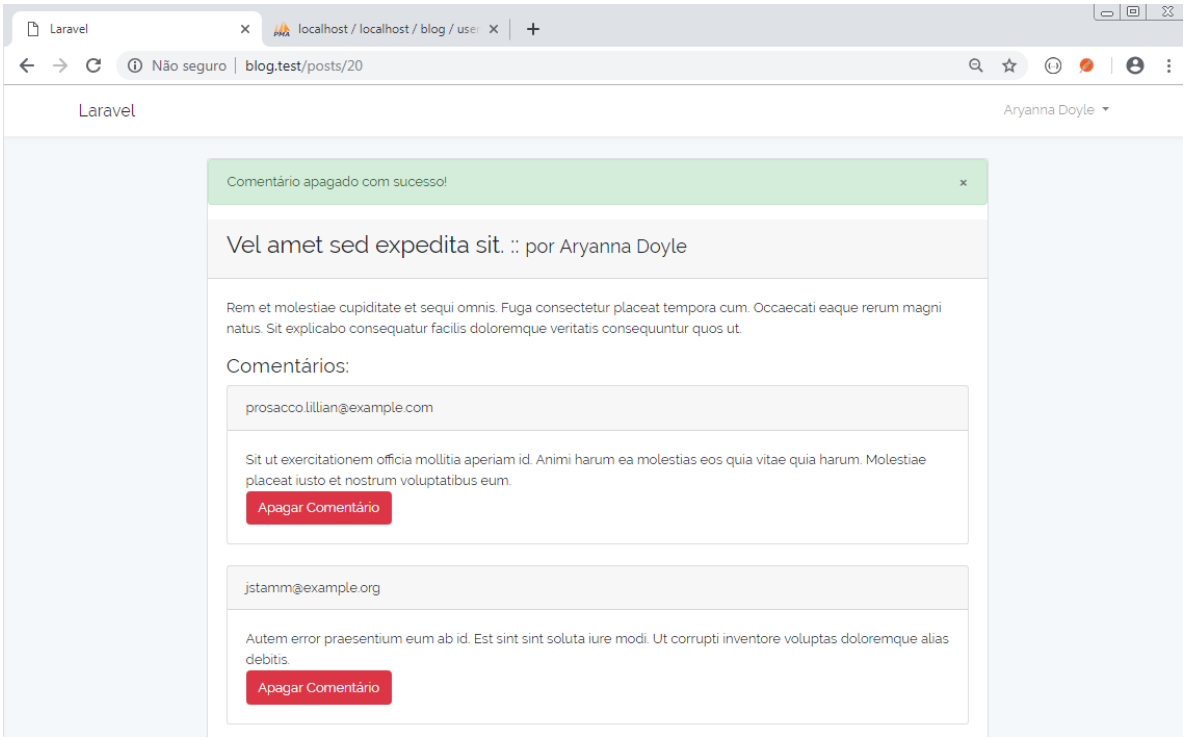

Laravel
localhost / localhost / blog / use
Não seguro | blog.test/login
Laravel
Login Register
Login
E-Mail Address
ryley12@example.net
Password
Remember Me
Login
Forgot Your Password?


Laravel
localhost / localhost / blog / use
Não seguro | blog.test/home
Laravel
Aryanna Doyle
Meus Posts
Doloribus omnis expedita ut.ALTERADO :: por Aryanna Doyle
Delectus sunt voluptatum asperiores ut expedita pariatur ut enim. Nam error asperiores mollitia tempore qui nobis sed. Laborum odit tenetur quasi optio. Placeat et aut voluptas nesciunt. Sit dolores porro sed velit fugit.
Ver post e comentários
Editar
Apagar
Vel amet sed expedita sit. :: por Aryanna Doyle
Rem et molestiae cupiditate et sequi omnis. Fuga consectetur placeat tempora cum. Occaecati eaque rerum magni natus. Sit explicabo consequatur facilis doloremque veritatis consequuntur quos ut.
Ver post e comentários
Editar
Apagar

Clique no link "Ver post e comentário" do segundo post:



Apague o primeiro comentário:



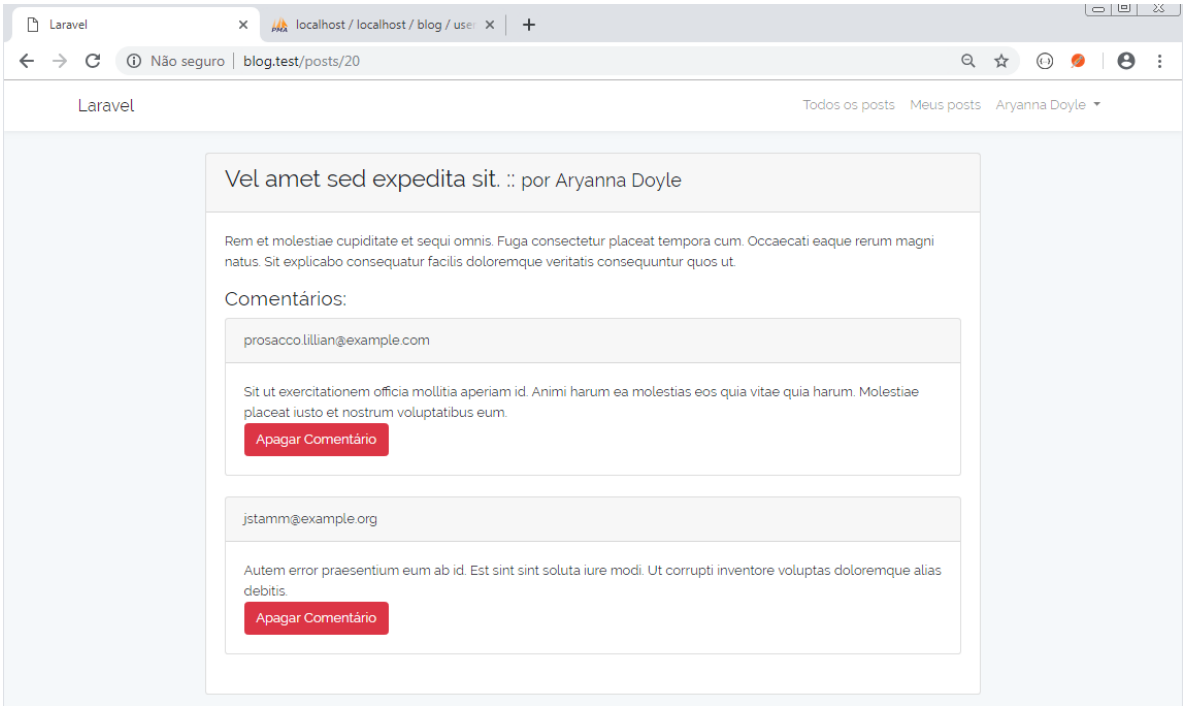
Aula 13 - Menu da aplicação

Colocar os links no menu da aplicação

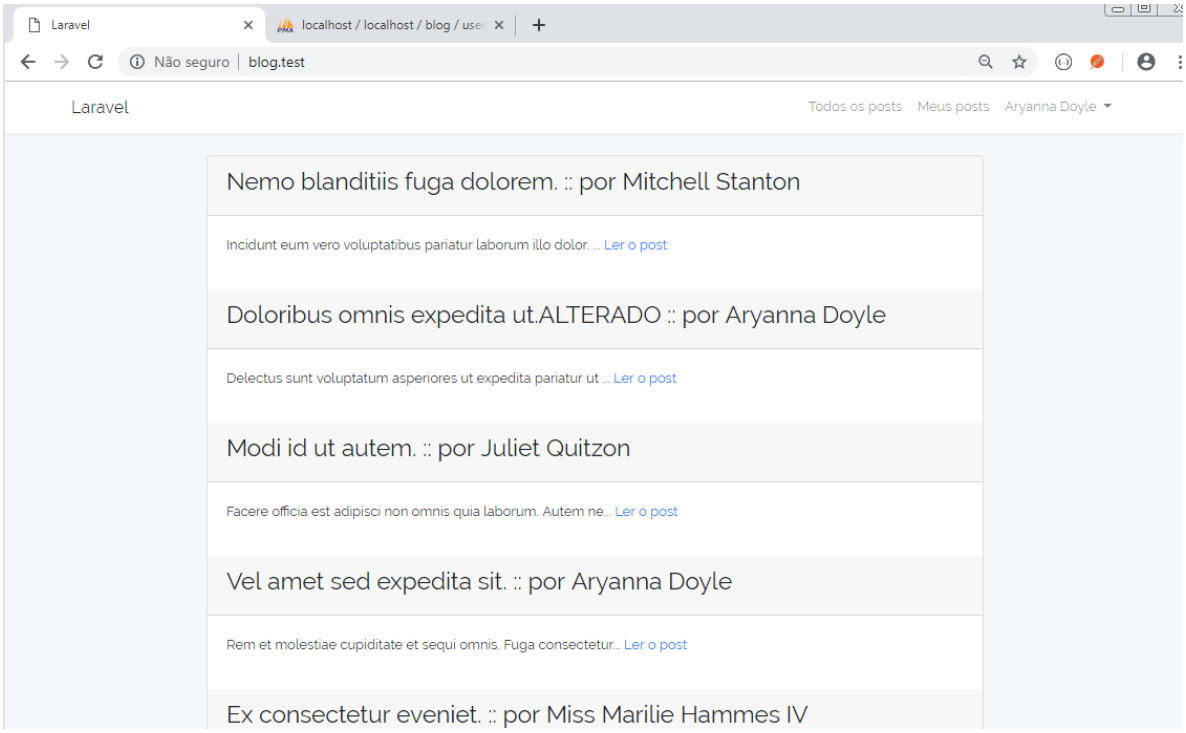
resources/views/layouts/app.blade.php

```
<!DOCTYPE html>
<html lang="{{ app()->getLocale() }}">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <!-- CSRF Token -->
    <meta name="csrf-token" content="{{ csrf_token() }}">

    <title>{{ config('app.name', 'Laravel') }}</title>

    <!-- Styles -->
    <link href="{{ asset('css/app.css') }}" rel="stylesheet">
</head>
<body>
    <div id="app">
        <nav class="navbar navbar-expand-md navbar-light navbar-laravel">
            <div class="container">
                <a class="navbar-brand" href="{{ url('/') }}">
                    {{ config('app.name', 'Laravel') }}
                </a>
                <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-
label="Toggle navigation">
                    <span class="navbar-toggler-icon"></span>
                </button>

                <div class="collapse navbar-collapse" id="navbarSupportedContent">
                    <!-- Left Side Of Navbar -->
                    <ul class="navbar-nav mr-auto">

                    </ul>

                    <!-- Right Side Of Navbar -->
                    <ul class="navbar-nav ml-auto">
                        <li>
                            <a class="nav-link" href="/">Todos os posts</a>
                        </li>
                        @auth
                            <li>
                                <a class="nav-link"href="/home">Meus posts</a>
                            </li>
                        @endauth
```

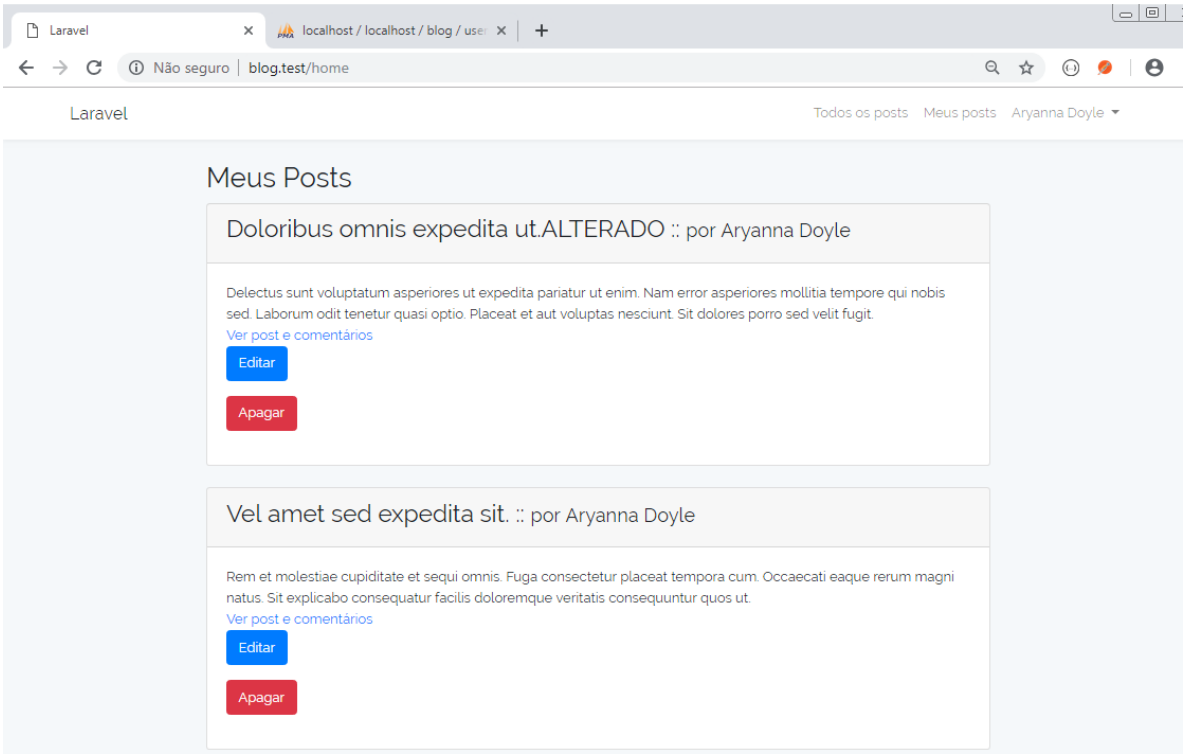
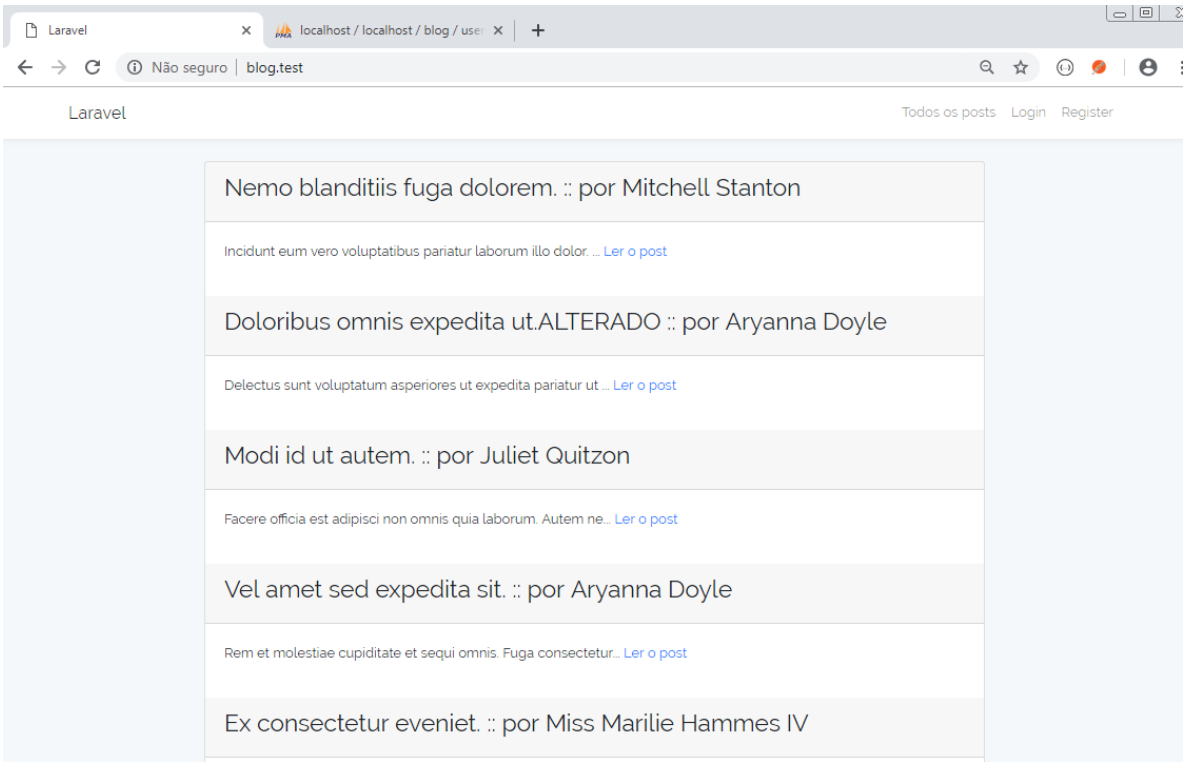
```
                <!-- Authentication Links -->
                @guest
                    <li><a class="nav-link" href="{{ route('login') }}">Login</a></li>
                    <li><a class="nav-link" href="{{ route('register') }}">Register</a></li>
                @else
                    <li class="nav-item dropdown">
                        <a class="nav-link dropdown-toggle" href="#" role="button" data-toggle="dropdown" aria-
haspopup="true" aria-expanded="false">
                            {{ Auth::user()->name }} <span class="caret"></span>
                        </a>
                        <div class="dropdown-menu" aria-labelledby="navbarDropdown">
                            <a class="dropdown-item" href="{{ route('logout') }}"
                              onclick="event.preventDefault();
                                            document.getElementById('logout-form').submit();">
                                Logout
                            </a>

                            <form id="logout-form" action="{{ route('logout') }}" method="POST" style="display:
none;">
                                @csrf
                            </form>
                        </div>
                    </li>
                @endguest
            </ul>
        </div>
    </div>
</nav>

<main class="py-4">
    @yield('content')
</main>
</div>

<!-- Scripts -->
<script src="{{ asset('js/app.js') }}"></script>
</body>
</html>
```



Clicando no link "Todos os posts":



Clicando no link "Meus posts":



Clicando em "Logout":