

Introdução ao Laravel 5

Pesquisadores: Duarte Fernandes e Ivanilson Melo



Instalação

- ☐ **Vagrant + Homestead (Windows / Mac / Linux)**
- ☐ **Valet (Mac)**
- ☐ **Manual (Linux / Mac)**

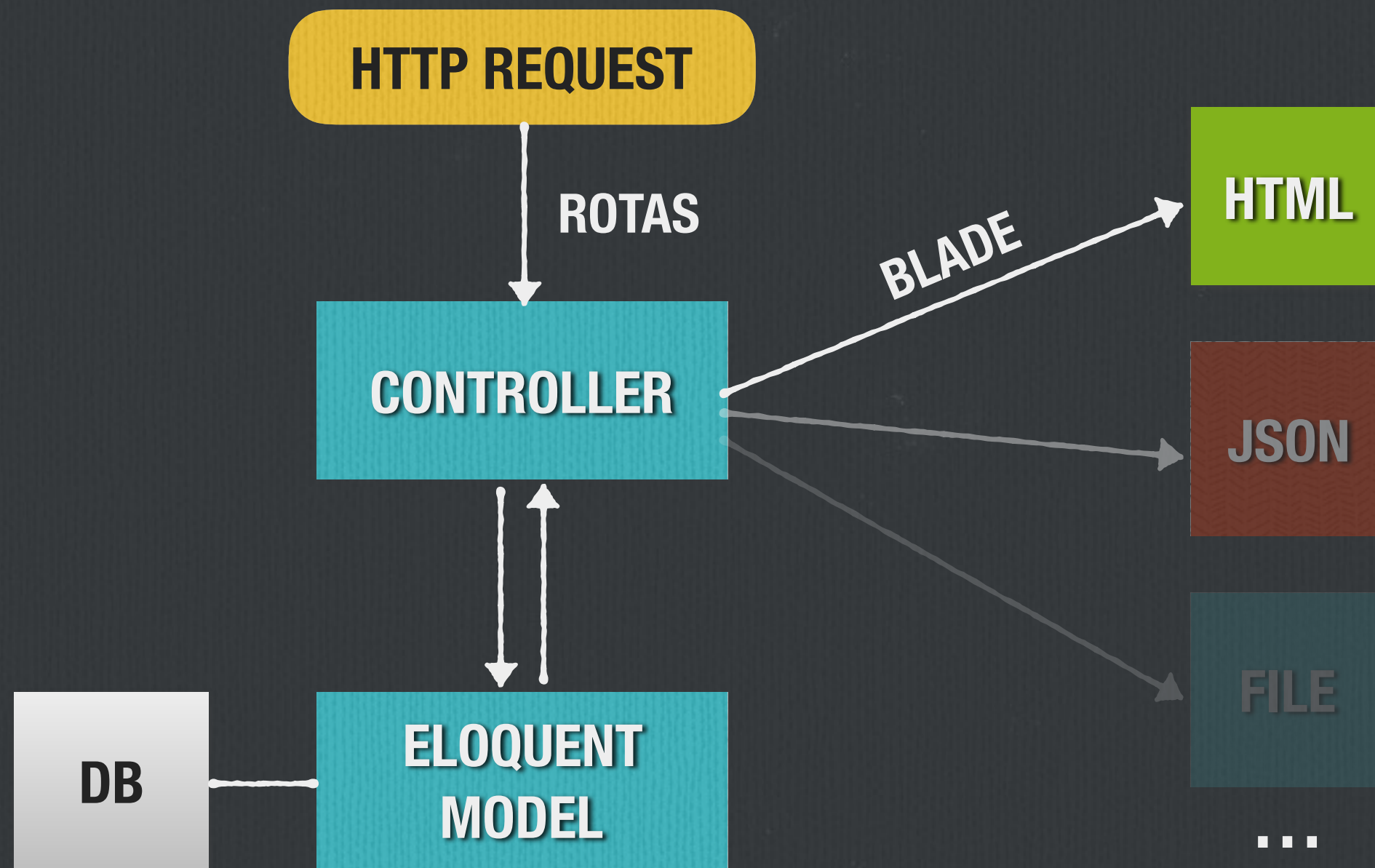
Características

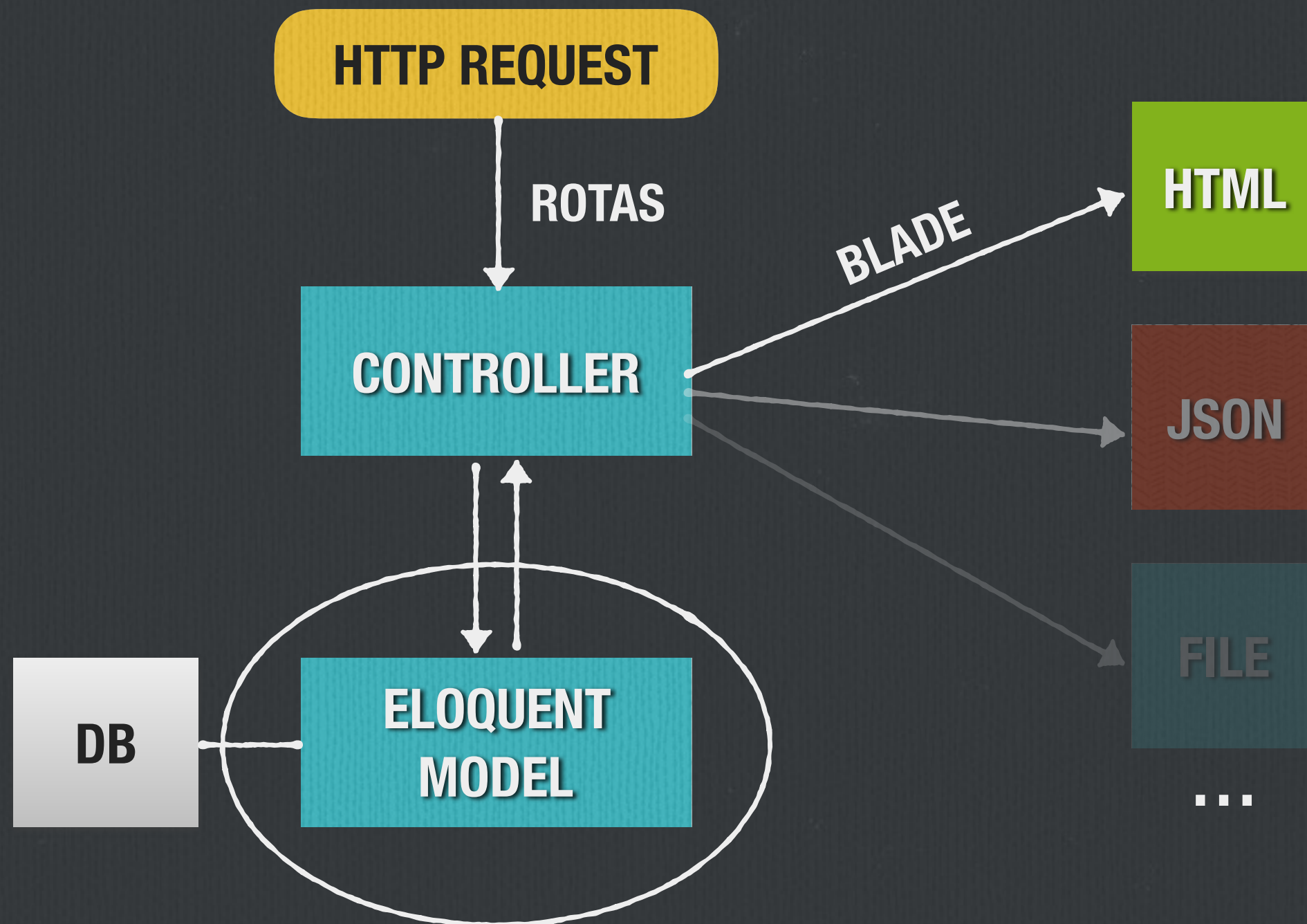
- ☐ **Arquitetura MVC bem definida**
 - ☐ **Estrutura pode ser alterada facilmente**
- ☐ **ORM poderoso**
- ☐ **Middlewares, event system, segurança**
- ☐ **Fácil de testar**
- ☐ **Template engine**

Workflow

- Todo comando executado com prefixo:

```
$ php artisan [comando]
```



Modelos

- ☐ Armazenados em `app/`

- ☐ Comando:

```
$ php artisan make:model [Model]
```

Modelos

Exemplo :

```
$ php artisan make:model Test
```

```
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Test extends Model
{
    //
}
```


Modelos

Relacionamentos

☐ 1 - n:

Post: `hasMany('App\Comment')`

Comment: `hasOne('App\Post')`

☐ 1 - 1

User: `hasOne('App\Phone')`

Phone: `belongsTo('App\User')`

☐ n - n

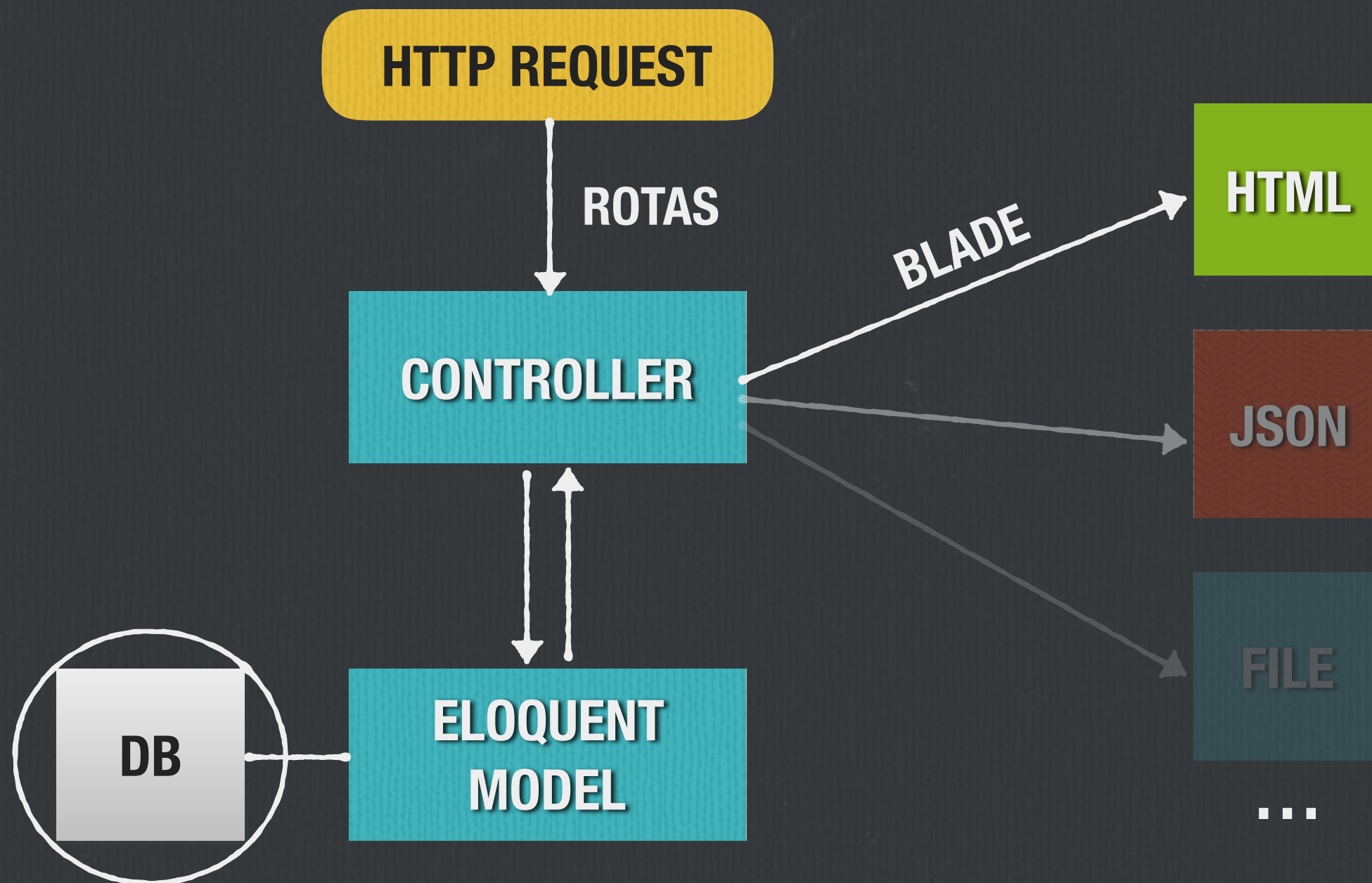
User: `belongsToMany('App\Role')`

Role: `belongsToMany('App\User')`

Modelos

Queries

- ❑ `User::all()`
- ❑ `User::find($id)`
- ❑ `User::where('active', 1)`
 `->orderBy('name', 'desc')`
 `->get()`



Migrations

- ☐ Armazenados em `database/migrations/`

- ☐ Comando:

```
$ php artisan make:migration [nome]
```

- ☐ Ou:

```
$ php artisan make:model -m [nome]
```


Migrations

- Exemplo

```
$ php artisan make:model -m Test
```

- Arquivo criado

```
2016_10_05_015152_create_tests_table
```

Migrations

```
<?php

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

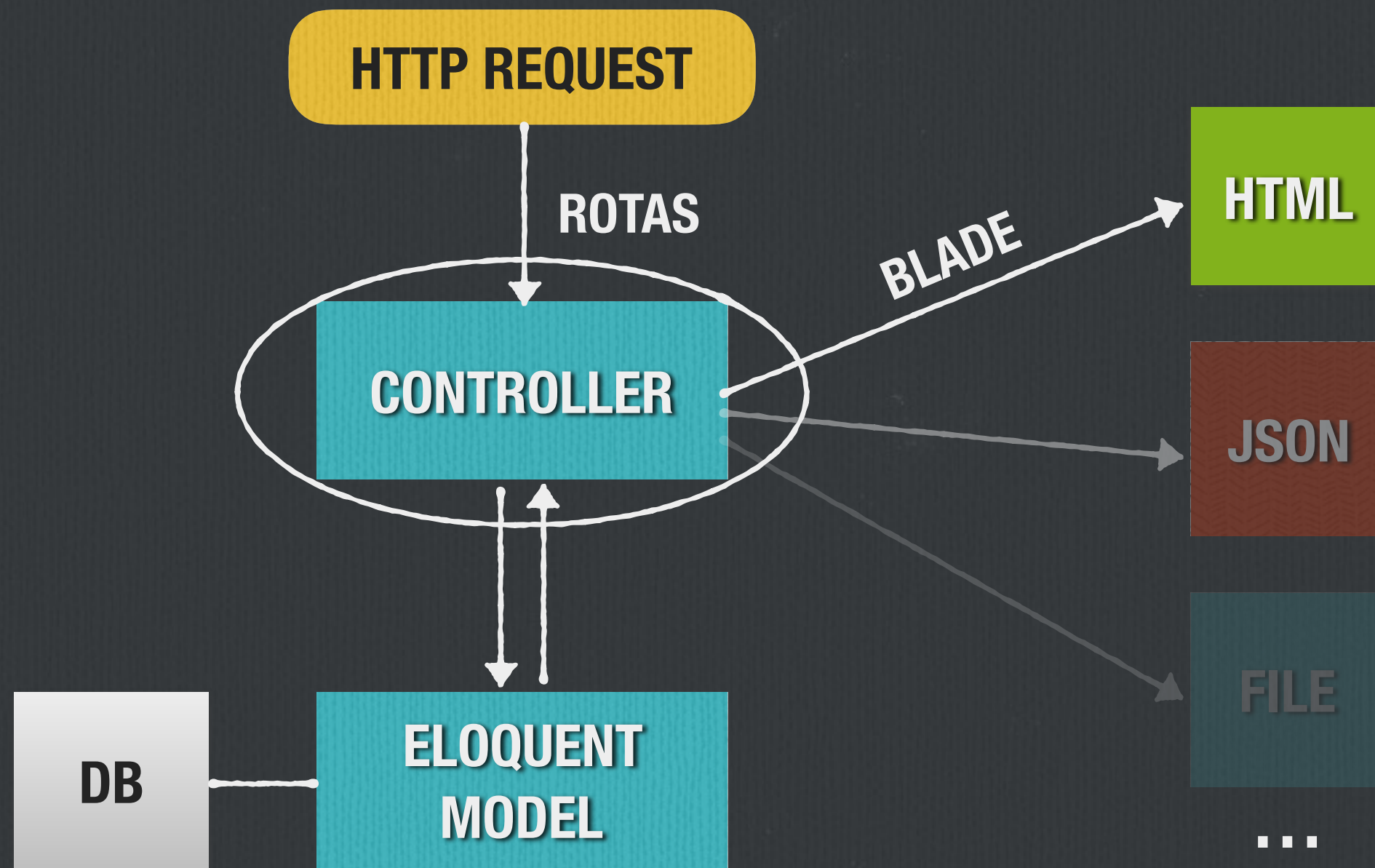
class CreateTestsTable extends Migration
{
    public function up()
    {
        Schema::create('tests', function (Blueprint $table) {
            $table->increments('id');
            $table->timestamps();
        });
    }

    public function down()
    {
        Schema::drop('tests');
    }
}
```


Migrations

Atributos

- ❑ `$table->string('nome')`
`varchar`
- ❑ `$table->integer('user_id')`
- ❑ `$table->foreign('user_id')`
`->references('id')->on('users')`
`->onDelete('cascade')`



Controllers

- ☐ Armazenados em `app/Http/Controllers/`
- ☐ Comando:

```
$ php artisan make:controller [nome]
```

Controllers

□ Exemplo

```
$ php artisan make:controller TestController
```

```
class TestController extends Controller
{
    public function index() {}
    public function create() {}
    public function store(Request $request) {}
    public function show($id) {}
    public function edit($id) {}
    public function update(Request $request, $id) {}
    public function destroy($id) {}
}
```


Controllers

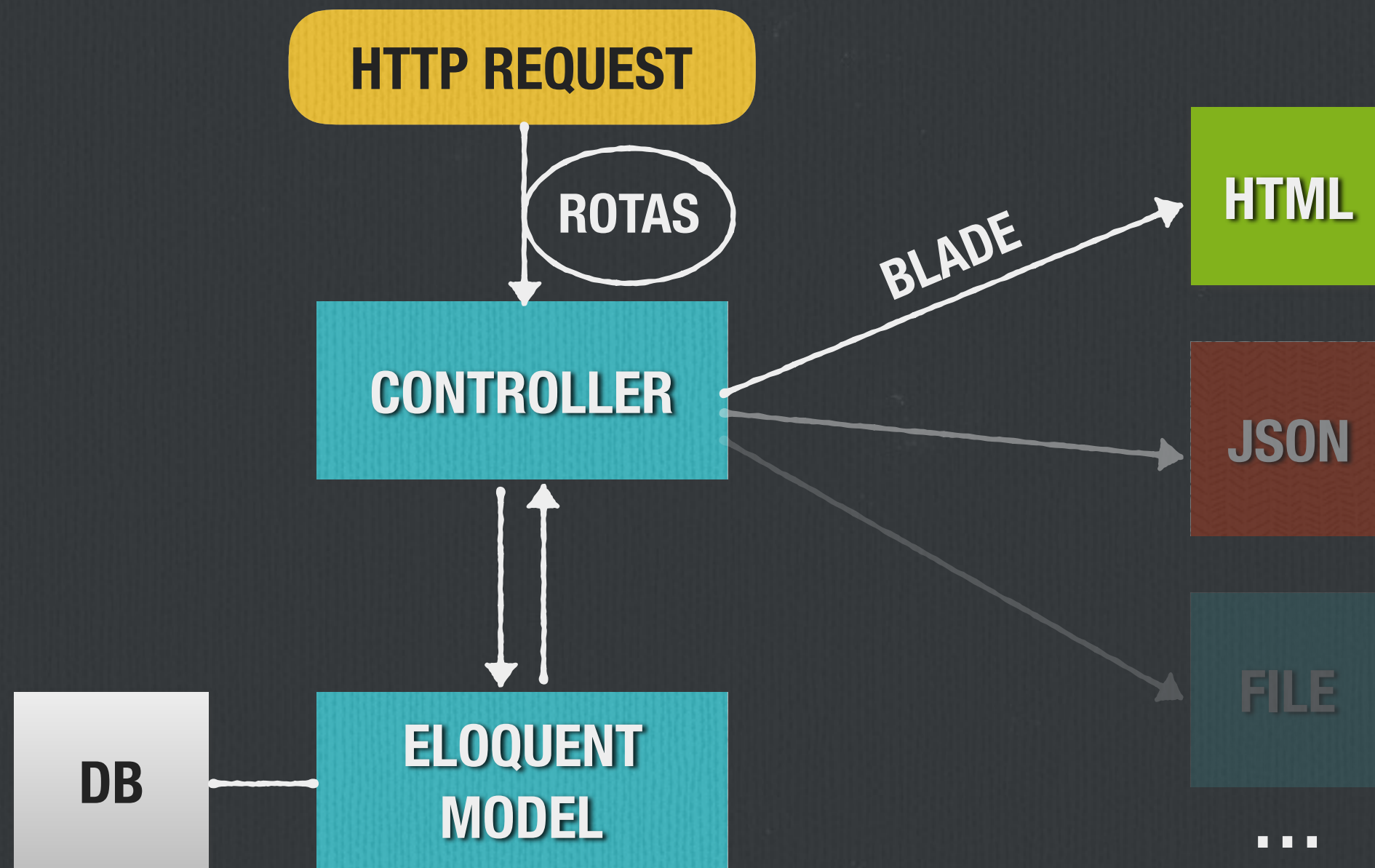
□ Exemplo de um método

```
public function showProfile($id)
{
    return view('user.profile', ['user' => User::findOrFail($id)]);
}
```

Controllers

□ Exemplo de responses

```
return response()->json([  
    'cidade' => 'Natal',  
    'estado' => 'RN'  
]);  
  
return response()->download($pathToFile);
```

Rotas

- Armazenados em
 - [5.2] `app/Http/routes.php`
 - [5.3] `routes/[web.php, api.php,...]`

```
Route::get($uri, $callback);  
Route::post($uri, $callback);  
Route::put($uri, $callback);  
Route::patch($uri, $callback);  
Route::delete($uri, $callback);  
Route::options($uri, $callback);
```


Rotas

□ Exemplos

```
Route::get('user/{id}', function ($id) {  
    return 'User '.$id;  
});
```

// opcional

```
Route::get('user/{name?}', function ($name = null) {  
    return $name;  
});
```

Rotas

□ Exemplos

```
Route::get('user/{id}', function ($id) {
    return 'User '.$id;
});

// opcional
Route::get('user/{name?}', function ($name = null) {
    return $name;
});

// regex
Route::get('user/{id}', function ($id) {
    //
})->where('id', '[0-9]+');
```


Rotas

□ Exemplos

```
// Named routes
Route::get('user/profile', 'UserController@showProfile')->name('profile');

/* chamando a "rota nomeada" */

// Gerando URL com base na rota nomeada...
$url = route('profile');

// Gerando um redirect com base na rota nomeada...
// utilizado em controllers
return redirect()->route('profile');
```

Rotas

☐ Grupo de Rotas

☐ Sintaxe:

```
Route::group([], function() { /* rotas */ })
```

☐ Exemplos

```
Route::group(['namespace' => 'Admin'], function()
{
    // Controllers dentro do namespace "App\Http\Controllers\Admin"

    Route::group(['namespace' => 'User'], function() {
        // Controllers dentro do namespace "App\Http\Controllers\Admin\User"
    });
});
```


Rotas

□ Mais exemplos de grupo de rotas

```
// Autorizando ou restringindo acesso de usuários
Route::group(['middleware' => 'auth'], function () {
    Route::get('user/list', function () {
        // Usa Auth Middleware
    });

    Route::get('user/profile', function () {
        // Usa Auth Middleware
    });
});

Route::group(['middleware' => 'guest'], function () {
    Route::get('/', function () {
        // Usa Auth Middleware
    });
});
```

Rotas

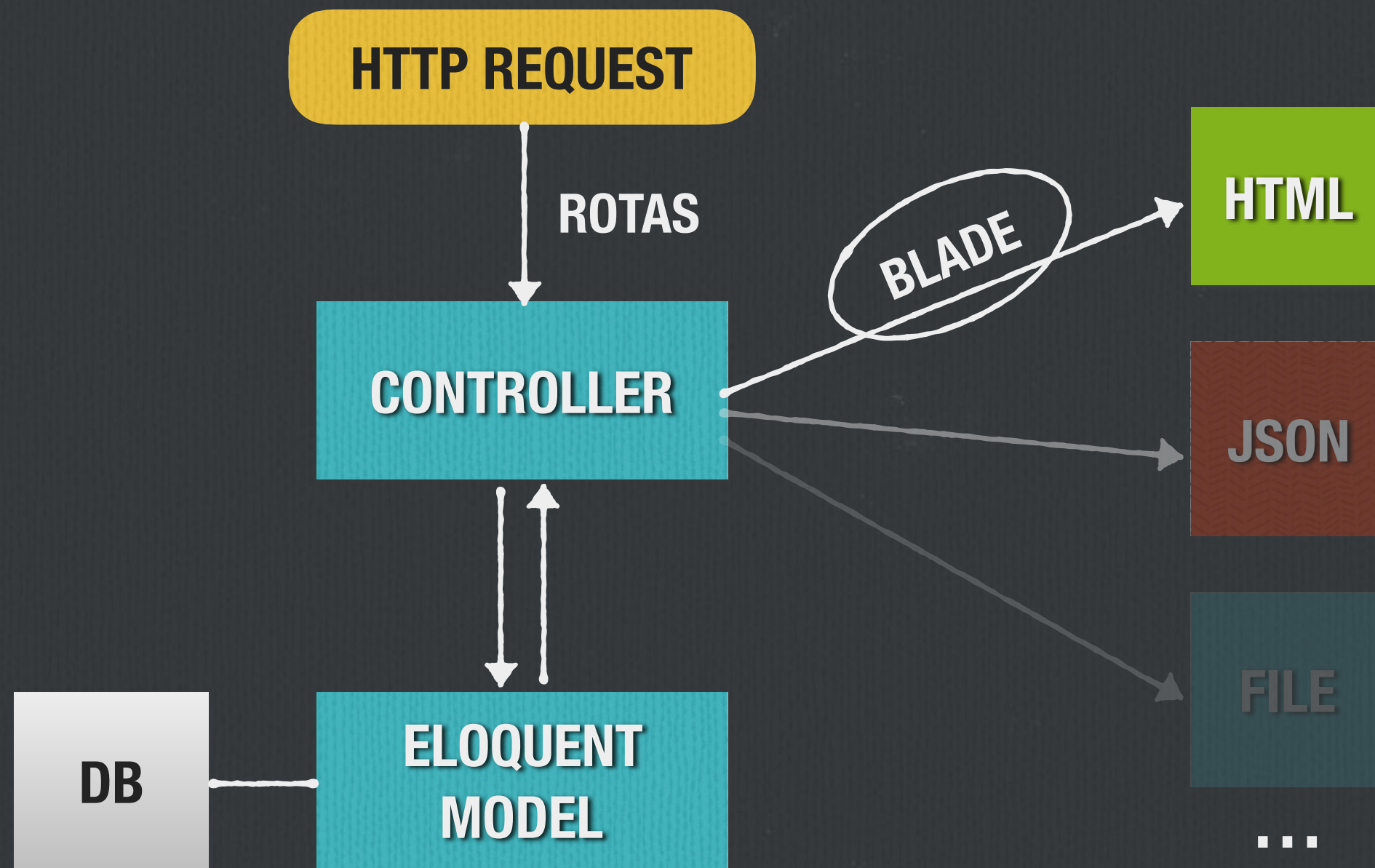
□ Mais exemplos de grupo de rotas

```
// Múltiplos groups

Route::group(['middleware' => 'auth'], function () {

    Route::group(['prefix' => 'user'], function() {
        Route::get('list', function ()    {
            // Usa Auth Middleware e prefixo de rota "user"
        });

        Route::get('profile', function () {
            // Usa Auth Middleware e prefixo de rota "user"
        });
    })
});
```

Blade Templates

- ☐ Armazenados em `resources/views/`
- ☐ Herança de templates
- ☐ Reuso (includes)
- ☐ Estruturas de controles, loops

Blade Templates

□ Exemplos

Template base: `resources/views/layouts/master.blade.php`

```
<html>
  <head>
    <title>App Name - @yield('title')</title>
  </head>
  <body>
    @section('sidebar')
      This is the master sidebar.
    @show

    <div class="container">
      @yield('content')
    </div>
  </body>
</html>
```

Blade Templates

□ Exemplos

Template filho: `resources/views/layouts/child.blade.php`

```
@extends('layouts.master')

@section('title', 'Titulo da pagina')

@section('sidebar')
    @parent

    <p>Isto será adicionado à section sidebar do master.</p>
@endsection

@section('content')
    <p>Conteúdo (content) do body.</p>
@endsection
```


Blade Templates

□ Exemplos

Rota: `Route::get('hello', 'Test/TestController@hello');`

Controller:

```
namespace App\Http\Controllers\Test;

public function hello()
{
    return view('welcome', ['nome' => 'Sorriso']);
}
```

□ Template:

```
@extends('layouts.master')
@section('title', 'Welcome')
@section('content')
    <p>Olá, {{ $nome }}.</p>
@endsection
```

Blade Templates

- Herança

- `@yield('area')`

- `@section('area')`

- `@extends('layouts.master')`

- `@include('partials.subview',
['optional' => 'data'])`

Blade Templates

- **Exibindo dados**

- `{{ $var }}`

- `@{{ $var }}`

- `{{ $var or 'Default' }}`

- `{{ isset($var) ? $var : 'Default' }}`

Blade Templates

□ Estrutura de Controle

```
@if (count($records) === 1)
|   I have one record!
@elseif (count($records) > 1)
|   I have multiple records!
@else
|   I don't have any records!
@endif

@unless (Auth::check())
|   You are not signed in.
@endunless
```


Blade Templates

□ Loops

```
@for ($i = 0; $i < 10; $i++)  
    The current value is {{ $i }}  
@endfor  
  
@foreach ($users as $user)  
    <p>This is user {{ $user->id }}</p>  
@endforeach
```

Lab:

Primeiro app Laravel



laratwitter

Avante

- ☐ **Documentação**
laravel.com/docs/
laravel.artesaos.org/docs/
- ☐ **Laracasts**
laracasts.com
- ☐ **Laravel News:**
laravel-news.com
- ☐ **Slack BR**
slack.laravel.com.br

Obrigado!