



Marcos Rodrigo Momo, M.Sc.  
e-mail: marcos.momo@ifsc.edu.br

## Programação para Internet II



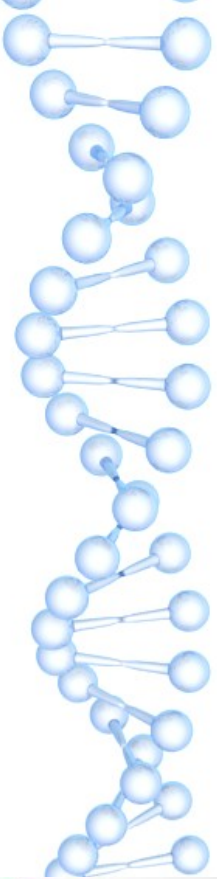
**INSTITUTO  
FEDERAL**  
Santa Catarina

Câmpus  
Gaspar

Gaspar, maio 2021.

# Roteiro

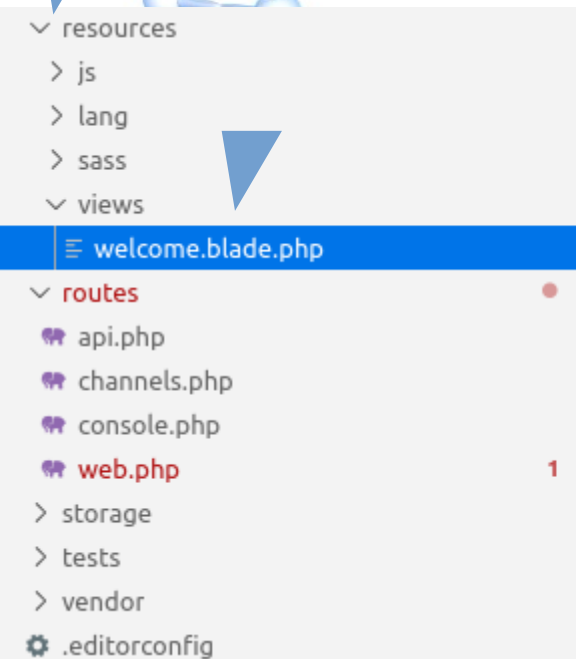
- Métodos do sistema de rotas
- Atividade práticas



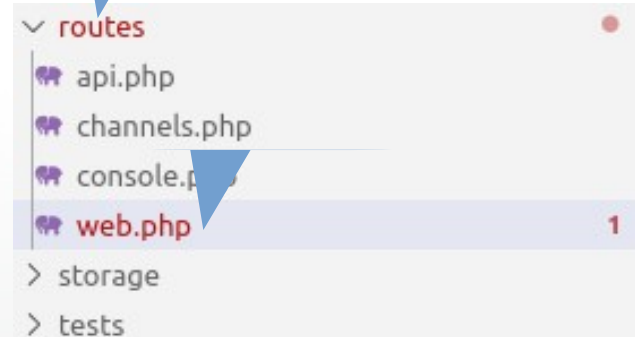
# Revisão aula 1

php artisan serve

php artisan route:list



```
83 <div class="title m-b-md">
84 |   Laravel
85 </div>
86
87 <div class="links">
88   <a href="https://laravel.com/docs">Docs</a>
89   <a href="https://laracasts.com">Laracasts</a>
90   <a href="https://laravel-news.com">News</a>
91   <a href="https://blog.laravel.com">Blog</a>
92   <a href="https://nova.laravel.com">Nova</a>
93   <a href="https://forge.laravel.com">Forge</a>
94   <a href="https://github.com/laravel/laravel">GitHub</a>
95 </div>
96 </div>
97 </div>
98 </body>
99 </html>
```



```
15
16
17
18
19 Route::get('/', function () {
20     return view('welcome');
21 });
22
```

# Rotas no Laravel

- As rotas no Laravel é uma das camadas mais importantes do Framework
- É um sistema de roteamento que se responsabiliza de manejar o fluxo das requisições e respostas, vindo do cliente
- As rotas definem a direção da URL (recurso) e o método no se pode aplicar a esta rota

# Métodos do sistema de rotas

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

# Atividades

- Com base no exemplo anterior, cria um agrupamento de três rotas retornando, respectivamente, três Views distintas.
- Em cada view criada, deverá conter o menu de navegação entre elas.
- No final, liste as rotas criadas.
- Lembre-se:
  - As views devem ser criadas na pasta resources/views
  - Os arquivos de views devem, necessariamente ter o nome com a extensão nome.blade.php
  - 
  - As views são HTMLs normais com a sintaxe blade, que é um template do Laravel, no qual permite fazermos algumas operações.

# Agrupamento de rotas

- Podemos agrupar rotas utilizando o método da classe route prefix.
- Vamos utilizar a função group para tratar todas as requisições que chegarem ao agrupamento criado.

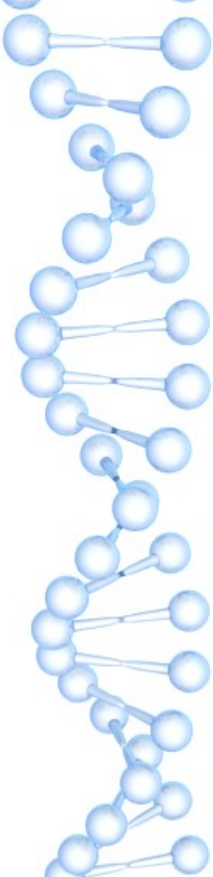
# Agrupamento de rotas

- Por exemplo:

```
//Agrupamento rotas e retornando strings
Route::prefix('/app')->Group (function(){
    Route::get('/', function(){
        return "Meu App";
    });

    Route::get("/user", function(){
        return "User";
    });

    Route::get("/profile", function(){
        return "Profile";
    });
});
```





```
//Agrupamento retornando Views
Route::prefix('/aplicacao')->Group (Function(){

    Route::get('/', function(){
        return view("app");
    });

    Route::get('/user', function(){
        return view("user");
    });

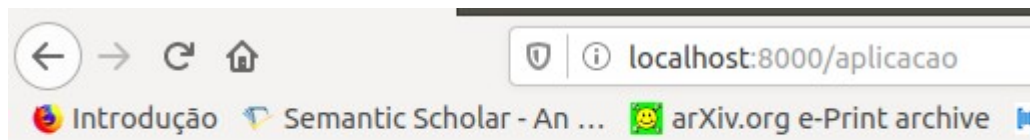
    Route::get('/profile', function(){
        return view("profile");
    });
});
```

```
<h1>Aqui a view do profile</h1>
<div>
    <a href="http://localhost:8000/aplicacao/user">User</a><br>
    <a href="http://localhost:8000/aplicacao/profile">profile</a><br>
    <a href="http://localhost:8000/aplicacao/">App</a><br>
</div>
```

Follow link (ctrl + click)

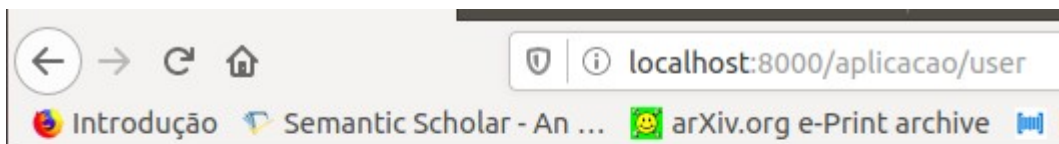
```
<h1>Aqui a view do user</h1>
<div>
    <a href="http://localhost:8000/aplicacao/user">User</a><br>
    <a href="http://localhost:8000/aplicacao/profile">profile</a><br>
    <a href="http://localhost:8000/aplicacao/">App</a><br>
</div>
```

# Exemplo



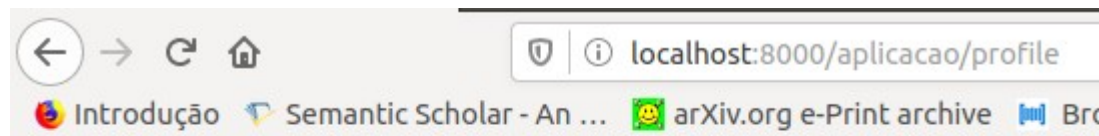
## Aqui a view do app

[User](#)  
[Profile](#)  
[App](#)



## Aqui a view do user

[User](#)  
[Profile](#)  
[App](#)

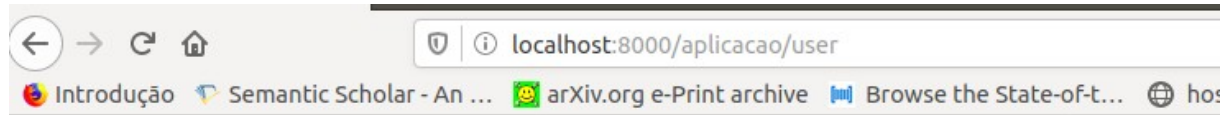


## Aqui a view do profile

[User](#)  
[Profile](#)  
[App](#)

# Nomeando rotas

- Caso mudamos o nome do agrupamento de uma rota o que vai acontecer
- Por exemplo mudar de aplicacao para sistema



Fazer o teste

# Nomeando rotas em web.php

- Nomeação de rotas é uma solução e uma boa prática para criar rotas.
- Por exemplo:

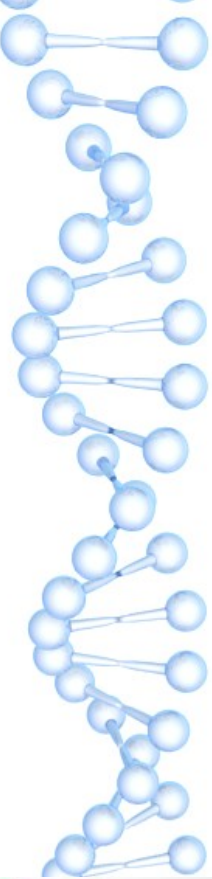
```
//Agrupamento retornando Views
//renomeando as rotas com função ->name
Route::prefix('/aplicacao')->Group (function(){

    Route::get('/', function(){
        return view("app2");
    })->name('app');

    Route::get('/user', function(){
        return view("user2");
    })->name('app.user');

    Route::get('/profile', function(){
        return view("profile2");
    })->name('app.profile');

});
```



# Alterando nas views

## Menu

- Podemos utilizar a função do route, passando como parâmetro o nome da rota
- Por exemplo:

```
<div>  
  <a href="{{route('app.user')}}">User</a><br>  
  <a href="{{route('app.profile')}}">Profile</a><br>  
  <a href="{{route('app')}}">App</a><br>  
</div>
```

# Adicionando itens de menu

```
Route::get('/produtos', function(){  
    echo "<h3>Produtos</h3>";  
    echo "<ol>";  
    echo "<li>Notebook </li>";  
    echo "<li>Impressora </li>";  
    echo "<li>Mouse </li>";  
    echo "</ol>";  
}) ->name('meusprodutos');
```

```
<h1>Meu App</h1>  
<div>  
    <a href="{{route('app.user')}}">User</a><br>  
    <a href="{{route('app.profile')}}">Profile</a><br>  
    <a href="{{route('app')}}">Meu App</a><br>  
    <a href="{{route('meusprodutos')}}">Meus produtos</a><br>  
</div>
```

# Vamos testar o método Post

```
Route::post('/requisicoes', function(Request $request) {  
    return 'Hello POST';  
});
```

Importando do Request

```
use Illuminate\Http\Request;
```





# Testando o método Post com o plugin do Chrome ARC

Advanced REST client

ARC Request

HTTP request

Socket

History

Yesterday

POST http://localhost:8000/requisicoes

terça-feira, 4 de fevereiro de 2020

POST http://localhost:8000/requisicoes

DELETE http://localhost:8000/requisicoes

sexta-feira, 17 de janeiro de 2020

DELETE http://localhost:8000/requisicoes

Method POST Request URL http://localhost:8000/requisicoes

Parameters

Headers

Body

Variables

Content-Type application/json

ADD HEADER

Headers are valid

Headers size: 30 bytes

SEND

Toggle source mode

Insert headers set

Header name Header value

X Edit Help



# CSRF

- Dar permissão à rota 'requisicao' contra CSRF
- Em:
  - APP/HTTP/Middleware/VerifyCsrfToken

# Testar demais métodos

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

# Atividade

## Renomeando rotas

- Utilizando a função name, redefinir as rotas criadas no exercício anterior.
- Fazer o teste alterando o nome do agrupamento.
- Listar as rotas

# Referências

- Laravel. <https://laravel.com/docs/6.x>
- Estrutura de pastas.  
<https://laravel.com/docs/5.4/structure>

