

# LARAVEL

ROTTER – VIEWS - CONTROLLERS

# ARQUIVOS GERADOS

- Arquivos Gerados (Rotas)
    - `/routes/auth.php` - contém todas as rotas necessárias para executar o processo de autenticação dentro da aplicação em questão.

## • NO TERMINAL:

## • **php artisan Route:list**

```
routes > auth.php
1  <?php
2
3  use App\Http\Controllers\Auth\AuthenticatedSessionController;
4  use App\Http\Controllers\Auth\ConfirmablePasswordController;
5  use App\Http\Controllers\Auth\EmailVerificationNotificationController;
6  use App\Http\Controllers\Auth\EmailVerificationPromptController;
7  use App\Http\Controllers\Auth\NewPasswordController;
8  use App\Http\Controllers\Auth>PasswordResetLinkController;
9  use App\Http\Controllers\Auth\RegisteredUserController;
10 use App\Http\Controllers\Auth\VerifyEmailController;
11 use Illuminate\Support\Facades\Route;
12
13 Route::middleware('guest')->group(function () {
14     Route::get('register', [RegisteredUserController::class, 'create'])
15         ->name('register');
16
17     Route::post('register', [RegisteredUserController::class, 'store']);
18
19     Route::get('login', [AuthController::class, 'index'])
20         ->name('login');
```

# Arquivos Gerados (Controle)

- **/app/Http/Controllers/auth/AuthenticatedSessionController.php** –
  - responsável por configurar a sessão de autenticação do usuário.
- **/app/Http/Controllers/auth/ConfirmablePasswordController.php** –
  - responsável pelos eventos de confirmação de senha.
- /app/Http/Controllers/auth>EmailVerificationNotificationController.php** - responsável
  - pelos eventos de verificação de e-mail do usuário que se registrou recentemente.

# Arquivos Gerados (Controle)

- **/app/Http/Controllers/auth>EmailVerificationPromptController.php**
- **/app/Http/Controllers/auth/NewPasswordController.php** - responsável pelos eventos de recebimento e atualização da nova senha do usuário.
- **/app/Http/Controllers/auth>PasswordResetLinkController.php** - responsável pelos eventos de solicitação de redefinição de senha - envio do link de reset.
- **/app/Http/Controllers/auth/RegisteredUserController.php** - responsável...
- **/app/Http/Controllers/auth/VerifyEmailController.php** - responsável pelos eventos de verificação de e-mail do usuário que se registrou recentemente.

The screenshot shows a Visual Studio Code (VS Code) interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** hdcevents
- Explorer:** Shows the project structure under "HDCEVENTS". A red circle highlights the "Http" folder, which contains "Controllers". Inside "Controllers", there is a folder "Auth" containing several files: AuthenticatedSessionController.php, ConfirmablePasswordController.php, EmailVerificationNotificationController.php, EmailVerificationPromptController.php, NewPasswordController.php, PasswordResetLinkController.php, RegisteredUserController.php, VerifyEmailController.php, and Controller.php.
- Editor:** The "auth.php" file is open in the editor. The code defines routes for user authentication:

```
<?php  
use App\Http\Controllers\Auth\AuthenticatedSessionController;  
use App\Http\Controllers\Auth\ConfirmablePasswordController;  
use App\Http\Controllers\Auth>EmailVerificationNotificationController;  
use App\Http\Controllers\Auth>EmailVerificationPromptController;  
use App\Http\Controllers\Auth>NewPasswordController;  
use App\Http\Controllers\Auth>PasswordResetLinkController;  
use App\Http\Controllers\Auth\RegisteredUserController;  
use App\Http\Controllers\Auth\VerifyEmailController;  
use Illuminate\Support\Facades\Route;  
  
Route::middleware('guest')->group(function () {  
    Route::get('register', [RegisteredUserController::class, 'create'])  
        ->name('register');  
  
    Route::post('register', [RegisteredUserController::class, 'store']);  
  
    Route::get('login', [AuthenticatedSessionController::class, 'create'])  
        ->name('login');  
});
```
- Terminal:** The terminal shows the command "PHP ARTISAN SERVE" being run, starting the Laravel development server at http://127.0.0.1:8000. It also shows the output of "artisan route:list".

# Arquivos Gerados (View)

- **/app/View/Components/AppLayout.php** - define um componente que será utilizado na renderização do layout da aplicação para usuários, com autenticação.
- **/app/View/Components/GuestLayout.php** - define um componente que será utilizado na renderização do layout da aplicação para visitantes, sem autenticação.
- **resources/views/dashboard.blade.php** - tela apresentada logo após a autenticação ser efetuada com sucesso.
- **/resources/views/auth/confirm-password.blade.php** - tela utilizada para confirmação de senha.

# Arquivos Gerados (View)

- **/resources/views/forgot-password.blade.php** - tela utilizada para recuperação de senha, quando usuário a esqueceu.
- **/resources/views/auth/login.blade.php** - tela de autenticação (solicita e-mail e senha) apresentada ao usuário.
- **/resources/views/auth/register.blade.php** - tela de registro de novo usuário, apresentada quando um novo cadastro é solicitado.
- **/resources/views/auth/reset-password.blade.php** - tela que permite ao usuário redefinir sua nova senha.
- **/resources/views/auth/verify-email.blade.php** - tela que permite efetuar a confirmação de e-mail do usuário que se registrou recentemente.

The screenshot shows the VS Code interface with the following details:

- File Explorer (Left):** Shows the project structure under "HDCEVENTS". A red circle highlights the "views" folder, which contains sub-folders "auth" and "components". The "auth" folder is circled in red and has a yellow arrow pointing to it labeled "LOGIN". The "components" folder is circled in red and has a yellow arrow pointing to it labeled "INDEX".
- Editor (Top Right):** Displays the file "auth.php" with its content:

```
1 <?php
2
3 use App\Http\Controllers\Auth\AuthenticatedSessionController;
4 use App\Http\Controllers\Auth\ConfirmablePasswordController;
5 use App\Http\Controllers\Auth\EmailVerificationNotificationController;
6 use App\Http\Controllers\Auth\EmailVerificationPromptController;
7 use App\Http\Controllers\Auth\NewPasswordController;
8 use App\Http\Controllers\Auth>PasswordResetLinkController;
9 use App\Http\Controllers\Auth\RegisteredUserController;
10 use App\Http\Controllers\Auth\VerifyEmailController;
11 use Illuminate\Support\Facades\Route;
```
- Terminal (Bottom):** Shows the output of running artisan commands:

```
PS C:\xampp\htdocs\hdcevents> PHP ARTISAN SERVE
Starting Laravel development server: http://127.0.0.1:8000
[Mon Aug 28 07:48:37 2023] 127.0.0.1:50085 [200]: /favicon.ico
PS C:\xampp\htdocs\hdcevents> php artisan route:list
```

Domain	Method	URI	Name	Action
			Middleware	

```
> Console  
> Exceptions  
> Http  
> Models  
> Providers  
> View  
> bootstrap  
> config  
✓ database  
> factories  
✓ migrations  
  2014_10_12_000000_create_users_table.php  
  2014_10_12_100000_create_password_resets_table.p...  
  2019_08_19_000000_create_failed_jobs_table.php  
  2019_12_14_000001_create_personal_access_tokens_...  
  2023_08_27_171730_create_produtos_tabela.php  
> seeders  
❖ .gitignore  
> node_modules  
> public
```

```
2  
3  use Illuminate\Database\Migrations\Migration;  
4  use Illuminate\Database\Schema\Blueprint;  
5  use Illuminate\Support\Facades\Schema;  
6  
7  class CreateUsersTable extends Migration  
8  {  
9      /**  
10       * Run the migrations.  
11       *  
12       * @return void  
13       */  
14      public function up()  
15      {  
16          Schema::create('users', function (Blueprint $table) {  
17              $table->id();  
18              $table->string('name');  
19              $table->string('email')->unique();  
20              $table->timestamp('email_verified_at')->nullable();  
21              $table->string('password');  
22              $table->rememberToken();  
23              $table->timestamps();  
24          });  
25      }  
26  
27      /**  
28       * Reverse the migrations.  
29       *  
30       * @return void  
31       */  
32      public function down()  
33      {  
34          Schema::dropIfExists('users');  
35      }  
36  }  
37
```

## Migração - Criação das Tabelas de Usuários

- (No terminal - dentro da pasta do projeto)  
(Considerando que o “.env” já está configurado)
- **PHP ARTISAN MIGRATE**

File Edit Selection View Go Run ...

hdcevents

Explorer (Ctrl+Shift+E) - 1 unsaved file

HDCEVENTS

- confirm-password.blade.php
- forgot-password.blade.php
- login.blade.php
- register.blade.php
- reset-password.blade.php
- verify-email.blade.php
- > components
- > layouts
  - app.blade.php
  - guest.blade.php
  - navigation.blade.php
  - dashboard.blade.php
  - welcome.blade.php
- > routes
  - api.php
  - auth.php
  - channels.php
  - console.php
  - web.php
- > storage
- > tests
- > vendor
- editorconfig
- .env
- .env.example

resources > views > dashboard.blade.php

dashboard.blade.php M X

web.php

2023\_08\_27\_171730\_create\_produtos\_tabela.php

```
1 <x-app-layout>
2   <x-slot name="header">
3     <h2 class="font-semibold text-xl text-gray-800 leading-tight">
4       {{ __('TELA DE ABERTURA BACKEND') }}
5     </h2>
6   </x-slot>
7
8   <div class="py-12">
9     <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
10       <div class="bg-white overflow-hidden shadow-sm sm:rounded-lg">
11         <div class="p-6 bg-white border-b border-gray-200">
12           VOCÊ ESTÁ LOGADO NO PAINEL ADMISTRATIVO!
13         </div>
14       </div>
15     </div>
16   </x-app-layout>
```

ROTAS

Mon Aug 28 15:38:46 2023] 127.0.0.1:50870 [200]: /favicon.ico  
[Mon Aug 28 15:38:47 2023] 127.0.0.1:50874 [200]: /favicon.ico  
[Mon Aug 28 15:42:26 2023] 127.0.0.1:50902 [200]: /favicon.ico

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under "HDCEVENTS". The "views" folder contains "auth" (with files: confirm-password.blade.php, forgot-password.blade.php, login.blade.php, register.blade.php, reset-password.blade.php, verify-email.blade.php), "components", "layouts" (with files: app.blade.php, guest.blade.php, navigation.blade.php), and "dashboard.blade.php" (which is selected and highlighted with a red circle). Other files like api.php, auth.php, channels.php, console.php, and web.php are also listed.
- Editor:** The "dashboard.blade.php" file is open. The code is as follows:

```
<x-app-layout>
    <x-slot name="header">
        <h2 class="font-semibold text-xl text-gray-800 leading-tight">
            {{ __('TELA DE ABERTURA BACKEND') }}
        </h2>
    </x-slot>

    <div class="py-12">
        <div class="max-w-7xl mx-auto sm:px-6 lg:px-8">
            <div class="bg-white overflow-hidden shadow-sm p-6 bg-white border-b border-gray-100 rounded-lg">
                VOCÊ ESTÁ LOGADO NO PAINEL ADMINISTRATIVO
            </div>
        </div>
    </div>
</x-app-layout>
```

- Terminal:** Shows log entries from the terminal window:

- [Mon Aug 28 15:38:46 2023] 127.0.0.1:50870 [200]: /favicon.ico
- [Mon Aug 28 15:38:47 2023] 127.0.0.1:50874 [200]: /favicon.ico
- [Mon Aug 28 15:42:26 2023] 127.0.0.1:50902 [200]: /favicon.ico

## LOGIN->DASHBOARD

- Quando estiver logado no Laravel Breeze ele irá criar o redirecionamento para o Dashboard
- Neste código você irá criar o painel de controle BACKEND.

# DEFINIÇÕES DE ROTAS



# /ROUTES/WEB.PHP

- As rotas tradicionais dentro /routes/web.php:

- Respostas de requisições;
- cookies;
- Sessões.



## Middleware Auth nas Rotas

- (Arquivo: “/routes/web.php”)
- O “Middleware”, também é possível aplicar o “auth” via definição no construtor da classe que manipula as rotas que devem ser restritas
- **a middleware** serve para agrupar as camadas do sistema entre aplicação final e o fornecimento de dados.



# /ROUTES/API.PHP

- São interceptadas dentro do Framework e tratadas diferentes:
- não aceitam cookies e sessões.
- Laravel sanctum verifica se o usuário está autenticado:
- Isso separa o FRONTEND DO BACKEND

```
13 | is assigned the "api" middleware group. Enjoy building your API!
14 |
15 */
16
17 Route::middleware('auth:sanctum')->get('/user', function (Request $request) {
18     return $request->user();
19 });
20 |
```

# /ROUTES/CHANNELS.PHP

- Serve para rotas de comunicação Broadcasting:
- A finalidade é trabalhar com eventos em tempo Real
- ex: tem 2 páginas abertas uma que lista os usuários e outra que adiciona, ao adicionar um novo, a página que lista automaticamente recebe o novo item.
- <https://laravel.com/docs/5.5/broadcasting>

# /ROUTES/CONSOLE.PHP

- Serve para comandos personalizados através do comando ARTISAN

# ROTAS



The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under "HDCEVENTS".
- Editor:** The "routes/web.php" file is open. The code defines two routes:

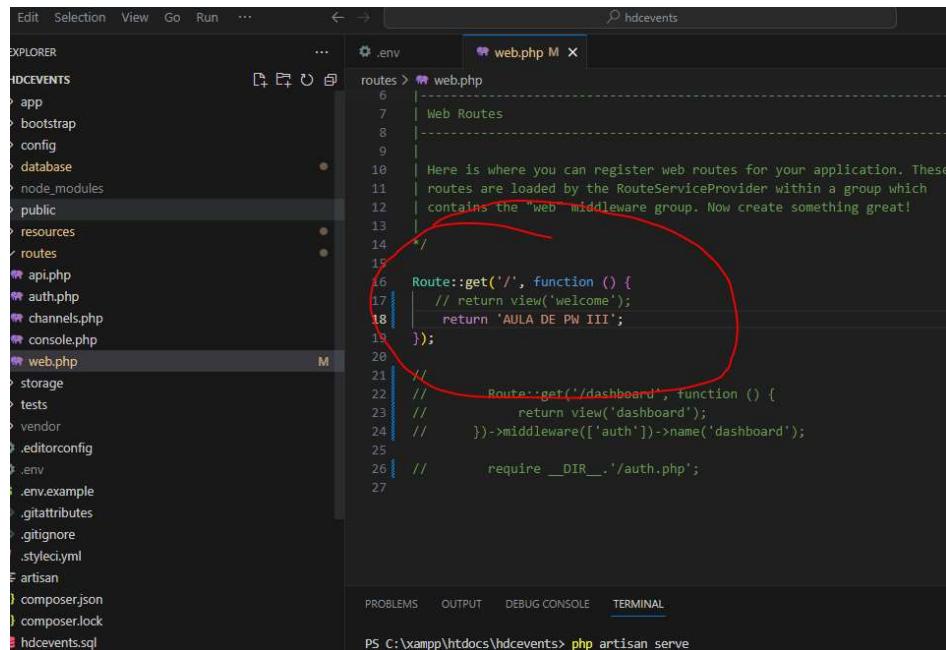
```
Route::get('/', function () {
    return view('welcome');
});
Route::get('/dashboard', function () {
    return view('dashboard');
})->middleware(['auth'])->name('dashboard');
```
- Terminal:** Shows log entries from a previous run.

## /ROUTES/WEB.PHP

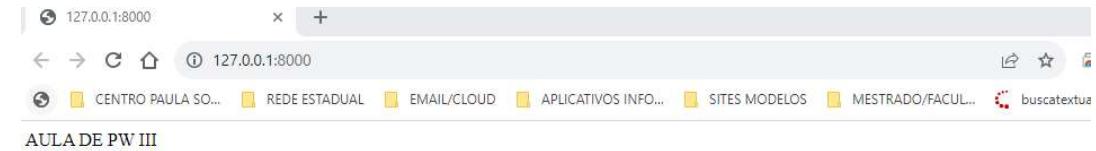
- O Cria as rotas de direcionamento das VIEWS no frontend;
- Todas as ROTAS tem que ter uma View publicada para funcionar.

# EDITAR /ROUTES/WEB.PHP

- `Route::get('/', function () {`
- `return 'AULA DE PW III';`
- `});`



```
routes > web.php M
6 | --- Web Routes
7 | |
8 | |
9 |
10 | Here is where you can register web routes for your application. These
11 | routes are loaded by the RouteServiceProvider within a group which
12 | contains the "web" middleware group. Now create something great!
13 |
14 */
15
16 Route::get('/', function () {
17     // return view('welcome');
18     return 'AULA DE PW III';
19 });
20
21 /**
22 // Route::get('/dashboard', function () {
23 //     return view('dashboard');
24 // })->middleware(['auth'])->name('dashboard');
25
26 // require __DIR__ . '/auth.php';
```

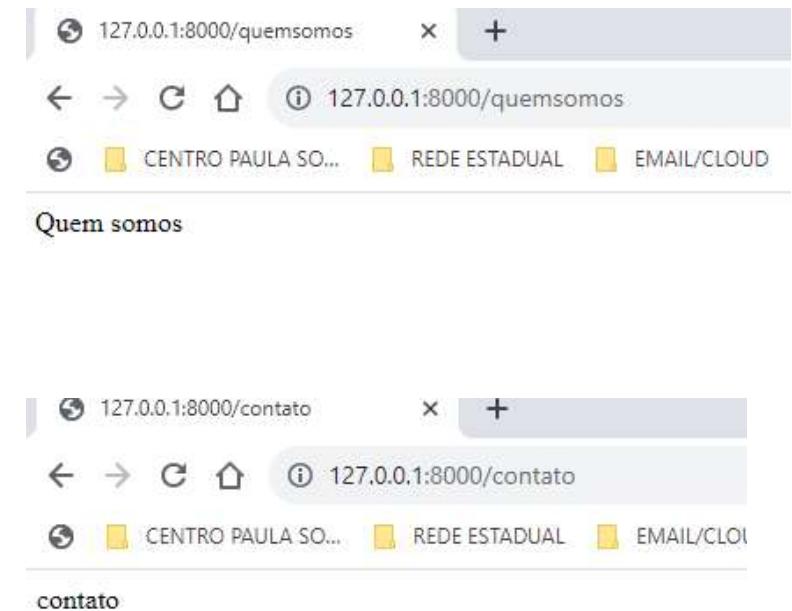


# EDITAR /ROUTES/WEB.PHP

```
> resources
  > routes
    api.php
    auth.php
    channels.php
    console.php
    web.php M
  > storage
  > tests
  > vendor
  .editorconfig
  .env
  .env.example
  .gitattributes
  .gitignore
  .htaccess

13  /*
14  *
15  */
16 Route::get('/', function () {
17     // return view('welcome');
18     return 'AULA DE PW III';
19 });
20
21 Route::get('/quemsomos', function () {
22     // return view('welcome');
23     return 'Quem somos';
24 });
25
26 Route::get('/contato', function () {
27     // return view('welcome');
28     return 'contato';
29 });
30
```

- Iremos criar as rotas contato e sobre nos;



## No objeto Route:

---

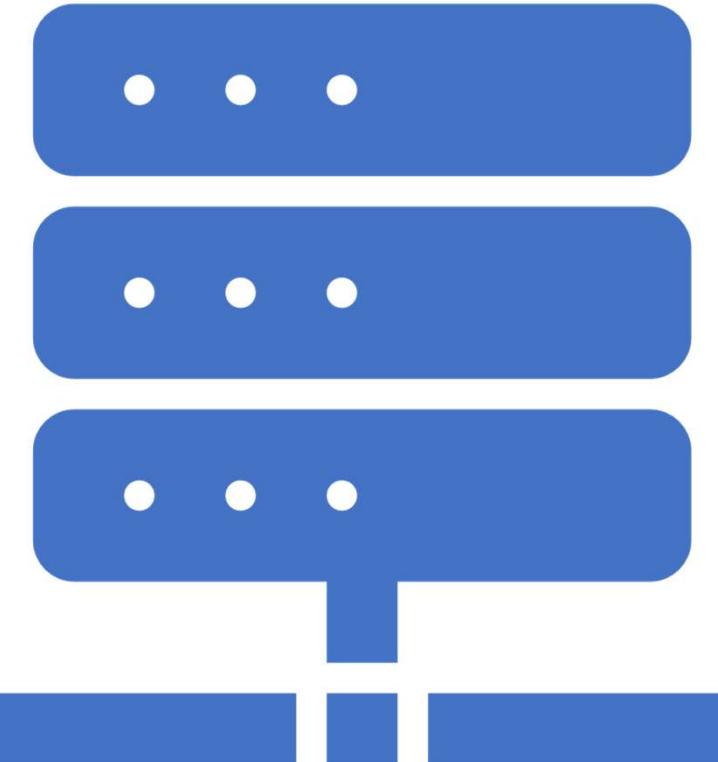
Chamamos os métodos:

- **Get;**
- **Post;**
- **Put;**
- **Patch;**
- **Delete;**
- **Options;**

## OBJETO ROUTE

---

- Route::get(**\$uri**, **\$callback**);
- **\$uri**: nossa rota.
- **\$callback**: ação que deve ser tomada



# CONTROLLERS

- As requisições serão feitas pelos controladores;
- Ao acessar uma rota estará associada um controller e as actions (dentro do controller);
- Os controladores serão CLASSES;



# CONTROLLES

Todos os controllers terão que ter a seguinte sintaxe:

PrincipalController

principal()

SobreNosController

sobrenos()

ContatoController

contato()

File Edit Selection View Go Run ...

hdcevents

EXPLORER

HDCEVENTS

app

- > Console
- > Exceptions
- > Http
  - > Controllers
    - > Auth
    - Controller.php
    - PrincipalController.php
  - > Middleware
  - > Requests
  - Kernel.php
  - > Models
  - > Providers
  - > View
- > bootstrap
- > config
- > database
- > node\_modules
- > public
- > resources
- > routes
  - api.php
  - auth.php
  - channels.php

web.php M PrincipalController.php X

1 //php

2

3 namespace App\Http\Controllers;

4

5 use Illuminate\Http\Request;

6

7 class PrincipalController extends Controller

8 {

9 }

10 //

11

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\xampp\htdocs\hdcevents> php artisan make:controller PrincipalController  
Controller created successfully.  
PS C:\xampp\htdocs\hdcevents> [ ]

php artisan make:controller PrincipalController

A screenshot of the Visual Studio Code (VS Code) interface. The left sidebar shows a project structure under the 'EXPLORER' tab, with the 'HDCEVENTS' folder expanded. Inside 'app', the 'Http' folder is selected, revealing its subfolders: 'Controllers', 'Middleware', 'Requests', 'Kernel.php', 'Models', 'Providers', 'View', 'bootstrap', 'config', and 'database'. The 'Controllers' folder contains two files: 'Controller.php' and 'PrincipalController.php'. A red arrow points from the 'Controller.php' file in the sidebar to the code editor. In the code editor, the file 'PrincipalController.php' is open, showing the following PHP code:

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class PrincipalController extends Controller
8 {
9     //
10 }
```

The code editor has tabs for '.env', 'web.php M', and 'PrincipalController.php U X'. The status bar at the bottom shows the terminal output:

```
PS C:\xampp\htdocs\hdcevents> php artisan make:controller PrincipalController
Controller created successfully.
PS C:\xampp\htdocs\hdcevents>
```

# ROUTE

- Route::get('/', '**PrincipalController@principal**');
- **PrincipalController**: nome do controller;
- **@principal**: action dentro do controller;

HDCEVENTS

app

> Console

> Exceptions

> Http

> Controllers

> Auth

ContatoController.php

Controller.php

PrincipalController.php

SobreNosController.php

> Middleware

> Requests

Kernel.php

> Models

> Providers

> View

> bootstrap

> config

> database

> node\_modules

> public

> resources

> routes

api.php

auth.php

channels.php

> OUTLINE

routes > web.php

```
21 //SLIDE 21
22 //CRIANDO ROTAS
23 //Route::get('/', function () {
24 //    return 'AULA DE PW III';
25 //});
26
27
28 //SLIDE 27
29 //DEPOIS DE CRIAR O CONTROLLER PRINCIPAL
30 //COMENTA A ROUTE ACIMA
31 Route::get('/', 'App\Http\Controllers\PrincipalController@principal');
32
33
34
35 Route::get('/quemsomos', function () {
36     // return view('welcome');
37     return 'Quem somos';
38 });
39
40 Route::get('/contato', function () {
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

PS C:\xampp\htdocs\hdcevents> php artisan make:controller SobreNosController  
Controller created successfully.

PS C:\xampp\htdocs\hdcevents> php artisan make:controller ContatoController  
Controller created successfully.

php artisan make:controller **SobreNosController**  
php artisan make:controller **ContatoController**

A screenshot of the Visual Studio Code (VS Code) interface. The top bar shows the file path: C:\xampp\htdocs\hdcevents. The title bar displays the project name: hdcevents. The left sidebar contains the Explorer, Search, Problems, and Terminal sections. The Explorer section shows the project structure under HDCEVENTS, including subfolders like Exceptions, Http, Controllers, Middleware, Requests, Kernel.php, Models, Providers, and Views. A red arrow points from the 'ContatoController.php' entry in the Explorer to the same file in the main editor area. The main editor area shows the code for ContatoController.php:

```
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class ContatoController extends Controller
8 {
9
10    //MÉTODO ACTIONS
11    public function principal(){
12        echo 'controller contato';
13    }
14
15
16 }
```

The code is annotated with a red circle around the method definition and its body. The bottom of the screen shows the terminal output: PS C:\xampp\htdocs\hdcevents> php artisan make:controller ContatoController

The screenshot shows a code editor interface with a dark theme. On the left is a sidebar with project navigation, showing a tree structure of files and folders. A red oval highlights the `SobreNosController.php` file in the `Http\Controllers` folder. The main panel displays the contents of `SobreNosController.php`:

```
app > Http > Controllers > SobreNosController.php
2
3     namespace App\Http\Controllers;
4
5     use Illuminate\Http\Request;
6
7     class SobreNosController extends Controller
8     {
9         //MÉTODO ACTIONS
10        public function principal(){
11            echo 'controller sobre nos';
12        }
13    }
14
15
```

Below the code editor is a terminal window showing the command `php artisan serve` being run twice, indicating the application is running locally.

**php artisan make:controller SobreNosController**



VIEWS



The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure under the 'HDCEVENTS' namespace. Key folders include 'bootstrap', 'config', 'database', 'node\_modules', 'public', 'resources' (containing 'css', 'js', 'lang'), and 'views' (containing 'app' and 'site' subfolders). The 'app' and 'site' folders are circled in red.
- Editor:** The 'welcome.blade.php' file is open in the main editor area. The code displays a simple layout with sections for 'Laravel' and 'Laracasts'.
- Terminal:** The terminal shows the command 'PS C:\xampp\htdocs\hdcevents>'.

# VIEWS

- Dentro do diretório **views**:
- Criar pasta **site**;
- Criar pasta **app**;

A screenshot of a file explorer interface, likely from a code editor like Laravel IDE or similar. The sidebar shows the following directory structure:

- resources
  - css
  - js
  - lang
- views
  - app
  - auth
  - components
  - layouts
    - principal.blade.php
    - dashboard.blade.php
    - welcome.blade.php
  - site
- routes
  - api.php
  - auth.php
  - channels.php
  - console.php
  - web.php
- storage
- tests
- vendor
- .editorconfig

The file `principal.blade.php` under the `layouts` folder is highlighted with a red oval.

M PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL  
PS C:\xampp\htdocs\hdcevents>

/views/site/principal.blade.php

The screenshot shows a code editor interface with a dark theme. On the left is a file tree sidebar with the following structure:

- HDCEVENTS
  - > config
  - > database
  - > node\_modules
  - > public
  - > resources
    - > css
    - > js
    - > lang
    - > views
      - > app
      - > auth
      - > components
      - > layouts
      - > site
        - contato.blade.php
        - principal.blade.php
        - sobreos.blade.php
        - dashboard.blade.php
        - welcome.blade.php
      - > routes
      - api.php
    - auto.php
    - channels.php
    - console.php
    - web.php
  - > storage
  - > tests

A red oval highlights the "site" folder and its contents. The main editor area shows the file "contato.blade.php" with the following content:

```
resources > views > site > contato.blade.php
```

```
1 |
```

The bottom navigation bar includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL. The TERMINAL tab is active, displaying the command prompt:

```
PS C:\xampp\htdocs\hdcevents>
```

Below the terminal, the URL `/views/site/contato.blade.php` is displayed.

The screenshot shows a code editor interface with a dark theme. On the left is a file tree sidebar with the following structure:

- HDCEVENTS
  - > config
  - > database
  - > node\_modules
  - > public
  - > resources
    - > css
    - > js
    - > lang
    - > views
      - > app
      - > auth
      - > components
      - > layouts
      - > site
        - contato.blade.php
        - principal.blade.php
        - sobreNos.blade.php
        - dashboard.blade.php
        - welcome.blade.php
      - > routes
      - api.php
  - > auto.php
  - > channels.php
  - > console.php
  - > web.php

A red oval highlights the 'site' folder and its contents. Below the file tree is a terminal window showing the command PS C:\xampp\htdocs\hdcevents>. The main workspace shows the file contato.blade.php with the following content:

```
resources > views > site > contato.blade.php
1 |
```

The bottom navigation bar includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL, with TERMINAL being the active tab.

/views/site/sobreNos.blade.php

# CONECTANDO A VIEW NO CONTROLLER

- No controller PrincipalController.php:

```
public function principal(){  
    return view('site.principal');
```

- Site = diretório dentro da view
- Principal = principal.blade.php

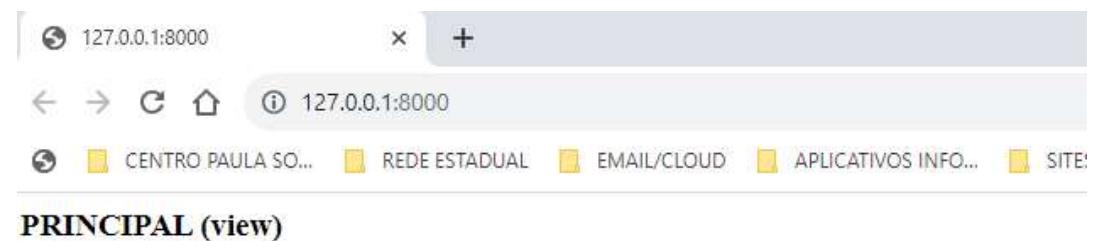
# /view/site/principal.blade.php

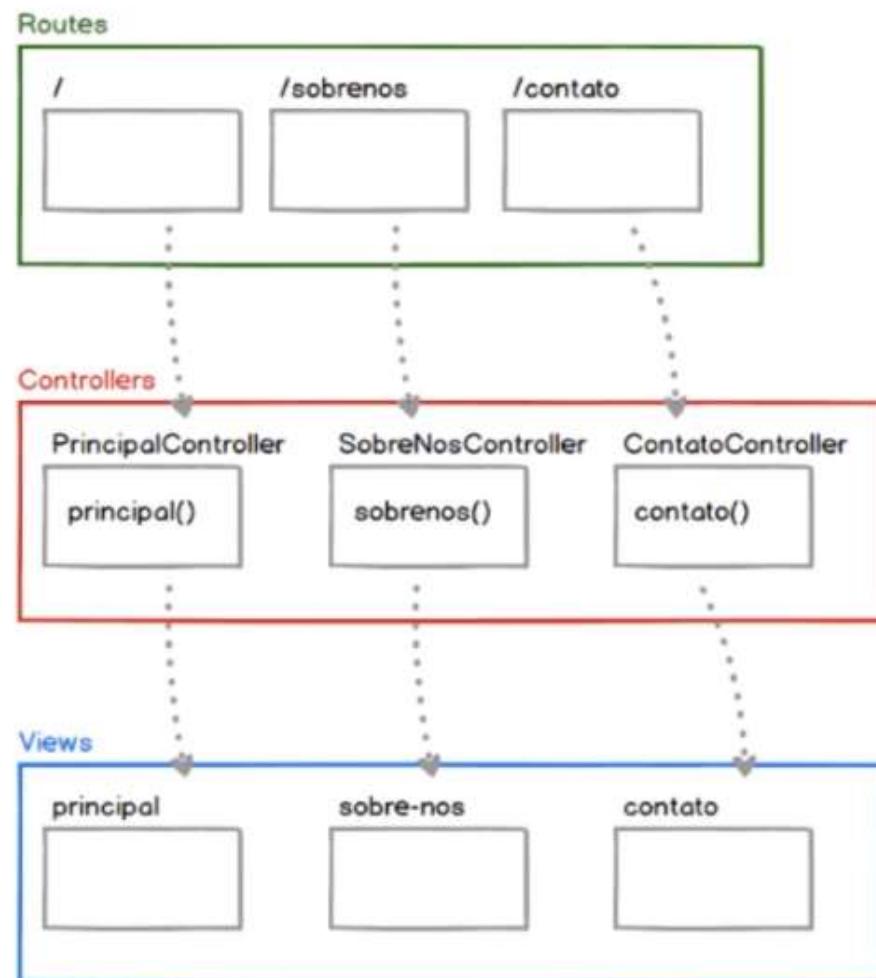
The screenshot shows the VS Code interface with the following details:

- EXPLORER:** Shows the project structure:
  - HDCEVENTS
  - Providers
  - View
  - bootstrap
  - config
  - database
  - node\_modules
  - public
  - resources
    - css
    - js
    - lang
    - views
      - app
      - auth
      - components
      - layouts
      - site
        - contato.blade.php
        - principal.blade.php
        - sobreos.blade.php
        - dashboard.blade.php
- CODE EDITOR:** The file `principal.blade.php` is open, showing the following code:

```
<h3>PRINCIPAL (view)</h3>
```

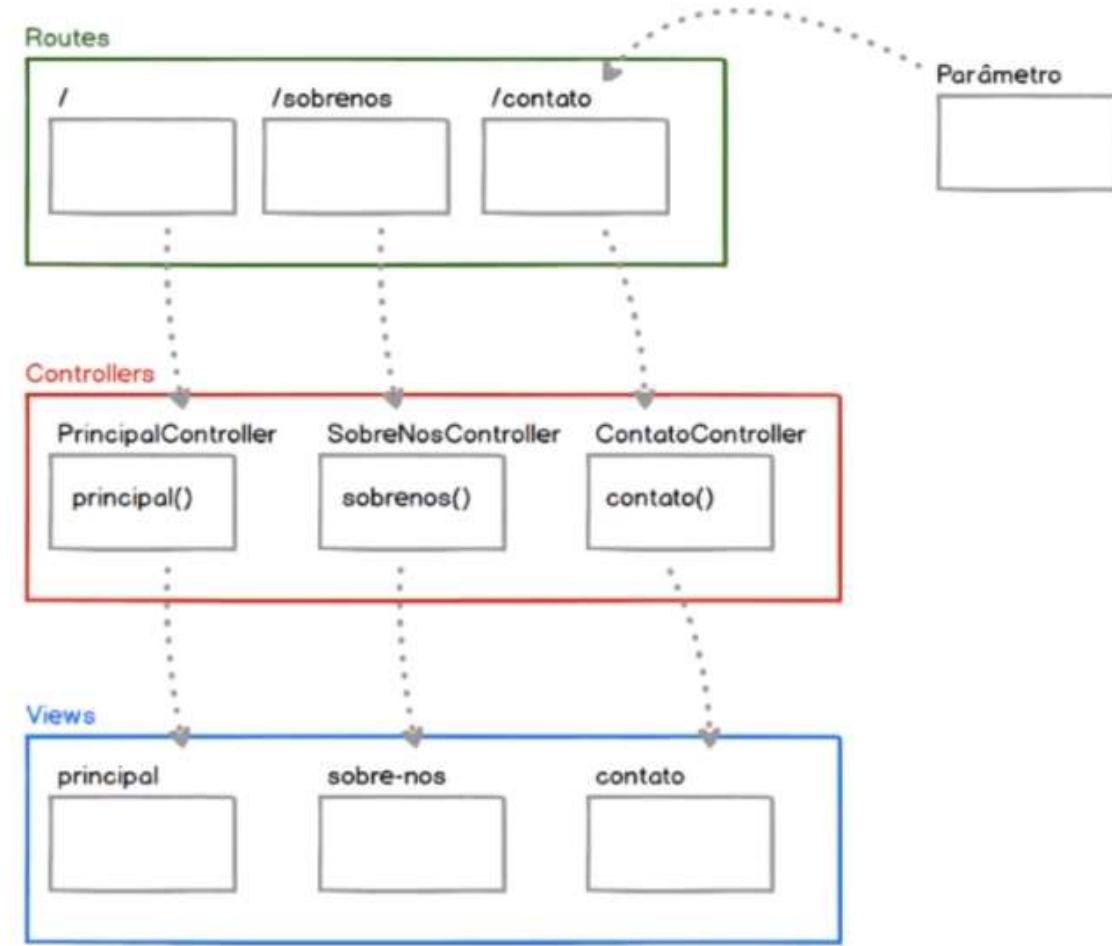
A red circle highlights the text `<h3>PRINCIPAL (view)</h3>`.
- TERMINAL:** Shows the command `PS C:\xampp\htdocs\hdcevents> php artisan serve` and the output: `Starting Laravel development server: http://127.0.0.1:8000 [Mon Aug 28 18:57:25 2023] 127.0.0.1:52314 [200]: /favicon.ico`.





Enviando parâmetros





The screenshot shows the VS Code interface with the following details:

- EXPLORER:** Shows the project structure:
  - HDCEVENTS (highlighted with a red circle)
  - app (highlighted with a red circle)
  - Console
  - Exceptions
  - Http
  - Models
  - Providers
  - View
  - bootstrap
  - config
  - database
  - node\_modules
  - public
  - resources (highlighted with a red circle)
  - routes
    - api.php
    - auth.php
    - channels.php
    - console.php
  - web.php (highlighted with a red circle)
  - storage
  - tests
  - vendor
  - .editorconfig
  - .env
  - .env.example
  - .gitattributes
- EDITOR:** Shows the content of `web.php`:

```
routes > web.php
38: //    return 'contato';
39: //});
40:
41: //SLIDE 27
42: //DEPOIS DE CRIAR O CONTROLLER PRINCIPAL
43: //COMENTA O ROUTE ACIMA
44: Route::get('/', 'App\Http\Controllers\HomeController@index');
45: Route::get('/sobreNos', 'App\Http\Controllers\HomeController@sobreNos');
46: Route::get('/contato', 'App\Http\Controllers\HomeController@contato');
47:
48: //
49: //        Route::get('/dashboard', function() {
50: //            return view('dashboard');
51: //        })->middleware(['auth'])->name('dashboard');
52:
53: //        require __DIR__.'/auth.php';
```
- TAB BAR:** Shows tabs for `SobreNosController.php`, `web.php`, and `hdcevents`.
- PANELS:** Shows panels for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL.
- TERMINAL:** Shows the command `PS C:\xampp\htdocs\hdcevents>`.

# Recebimento de parâmetros

# PASSAGEM DE PARÂMETROS NO BROWSER

```
routes > web.php
41 //SLIDE 27
42 //DEPOIS DE CRIAR O CONTROLLER PRINCIPAL
43 //COMENTA O ROUTE ACIMA
44 Route::get('/', 'App\Http\Controllers\PrincipalController@principal');
45 Route::get('/sobreNos', 'App\Http\Controllers\SobreNosController@principal');
46 Route::get('/contato', 'App\Http\Controllers\ContatoController@principal');
47
48 //passagem de parametros no browser
49 Route::get('/contato/{nome}', [
50     'callback' => function(string $nome){echo 'passagem de parametros via browser: ' . $nome;};
51 ]);
52
53 );
54
55 /**
56 //      Route::get('/dashboard', function () {
57 //          return view('dashboard');
58 //      })->middleware(['auth'])->name('dashboard');
59 //
```

parâmetro

callback

variável

← → C ⌂

① 127.0.0.1:8000/contato



CENTRO PAULA SO...

REDE ESTADUAL

EMAIL/CLOUD

APLICATIVOS INFO...

SITES MODELOS

MESTRADO/FACU

## CONTATO (view)

← → C ⌂ ⓘ 127.0.0.1:8000/contato/MARCELO

⟳ CENTRO PAULA SO... REDE ESTADUAL EMAIL/CLOUD APlicativos info... SITES MODELOS MESTRADO

passagem de parametros via browser:MARCELO

File Edit Selection View Go Run ... < > hdcevents

EXPLORER

HDCEVENTS

- app
  - Console
  - Exceptions
  - Http
  - Models
  - Providers
  - View
- bootstrap
- config
- database
- node\_modules
- public
- resources
- routes
  - api.php
  - auth.php
  - channels.php
  - console.php
  - web.php
- storage
- tests
- vendor
- .editorconfig
- .env
- .env.example
- .gitattributes

routes > web.php

```
41 //SLIDE 27
42 //DEPOIS DE CRIAR O CONTROLLER PRINCIPAL
43 //COMENTA A ROUTE ACIMA
44 Route::get('/', 'App\Http\Controllers\PrincipalController@principal');
45 Route::get('/sobreNos', 'App\Http\Controllers\SobreNosController@principal');
46 Route::get('/contato', 'App\Http\Controllers\ContatoController@principal');
47
48 //passagem de parametros no browser
49 Route::get('/contato/{nome}/{mensagem?}', [
50     function(string $nome, string $mensagem = 'sem texto')
51     {
52         echo "passagem de parametros via browser: $nome - $mensagem";
53     }
54 ]);
55
56 //
57 // Route::get('/dashboard', function () {
58 //     return view('dashboard');
59 //});
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Starting Laravel development server: http://127.0.0.1:8000  
[Thu Aug 31 17:06:18 2023] 127.0.0.1:63076 [200]: /favicon.ico  
[Thu Aug 31 17:06:32 2023] 127.0.0.1:63082 [200]: /favicon.ico  
[Thu Aug 31 17:07:10 2023] 127.0.0.1:63087 [200]: /favicon.ico  
PS C:\xampp\htdocs\hdcevents> PHP ARTISAN SERVE  
Starting Laravel development server: http://127.0.0.1:8000  
[Thu Aug 31 17:21:26 2023] 127.0.0.1:63133 [200]: /favicon.ico  
[Thu Aug 31 17:21:40 2023] 127.0.0.1:63137 [200]: /favicon.ico  
[Thu Aug 31 17:21:51 2023] 127.0.0.1:63139 [200]: /favicon.ico  
[Thu Aug 31 17:22:08 2023] 127.0.0.1:63143 [200]: /favicon.ico  
[Thu Aug 31 17:27:05 2023] 127.0.0.1:63160 [200]: /favicon.ico  
[Thu Aug 31 17:27:14 2023] 127.0.0.1:63162 [200]: /favicon.ico

Utilizo na passagem de parâmetro o sinal de ?

Coloco um texto qualquer para aparecer caso esquecer passar o parâmetro

Passo a variável



passagem de parametros via browser: MARCELO - sem texto



127.0.0.1:8000/contato/MARCELO/AGORA COM TEXTO



CENTRO PAULA SO...

REDE ESTADUAL

EMAIL/CLOUD

APLICATIVOS INFO...

SITES MODELOS



passagem de parametros via browser: MARCELO - sem texto

← → C ⌂

① 127.0.0.1:8000/contato/MARCELO/AGORA%20COM%20TEXTO



CENTRO PAULA SO... REDE ESTADUAL EMAIL/CLOUD APlicativos INFO... SITES MODELOS MES

passagem de parametros via browser: MARCELO - AGORA COM TEXTO



ROTAS

agrupamentos

EXPLORER

DCEVENTS

- > auth
- < components
  - application-logo.blade.php
  - auth-card.blade.php
  - auth-session-status.blade.php
  - auth-validation-errors.blade.php
  - button.blade.php
  - dropdown-link.blade.php
  - dropdown.blade.php
  - input.blade.php
  - label.blade.php
  - nav-link.blade.php
  - responsive-nav-link.blade.php
- > layouts
- < site
  - contato.blade.php M
  - principal.blade.php M
  - sobreos.blade.php M
  - welcome.blade.php
  - dashboard.blade.php
- < routes
  - api.php

principalController.php

resources > views > site > principal.blade.php

```
1 <h3>PRINCIPAL (view) </h3>
2 <ul>
3   <li>
4     <a href="/"> PRINCIPAL </a>
5   </li>
6
7   <li>
8     <a href="/sobreos"> SOBRE NÓS </a>
9   </li>
10
11   <li>
12     <a href="/contato"> CONTATO </a>
13   </li>
14
15 </ul>
16
17
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

php +

# agrupamento

---

```
label.blade.php
nav-link.blade.php
responsive-nav-link.blade.php
> layouts
  site
    contato.blade.php
    principal.blade.php
    sobreNos.blade.php
    dashboard.blade.php
    welcome.blade.php
  routes
    api.php
    auth.php
    channels.php
    messages.php

43 //DEPOIS DE CRIAR O CONTROLLER PRINCIPAL
44 //COMENTA O ROUTE ACIMA
45 Route::get('/', 'App\Http\Controllers\PrincipalController@principal');
46 Route::get('/sobreNos', 'App\Http\Controllers\SobreNosController@principal');
47 Route::get('/contato', 'App\Http\Controllers\ContatoController@principal');
48 //SLIDE 52
49 Route::prefix('/admin')->, group(function(){
50   Route::get('/clientes', function() {return 'Clientes'});
51   Route::get('/fornecedores', function() {return 'Fornecedores'});
52   Route::get('/produtos', function() {return 'produtos'});
53 });
54
55
56
57
58 //SLIDE 26
59 //passagem de parametros no browser
```

```
47 Route::get('/contato', 'App\Http\Controllers\ContatoController@principal');
48 //SLIDE 52
49 Route::prefix('/admin')->group (function(){
50     Route::get('/clientes', function(){return 'Clientes';});
```

M PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

powershell +

	GET HEAD	admin/clientes		Closure	W
	GET HEAD	admin/fornecedores		Closure	W
	GET HEAD	admin/produtos		Closure	W
	GET HEAD	api/user		Closure	ap
M					Ap
e\Authenticate:sanctum					Ap
	GET HEAD	contato		App\Http\Controllers\ContatoController@principal	W
	GET HEAD	sanctum/csrf-cookie		Laravel\Sanctum\Http\Controllers\CsrfCookieController@show	W
	GET HEAD	sobreNos		App\Http\Controllers\SobreNosController@principal	W
+-----+-----+-----+-----+					

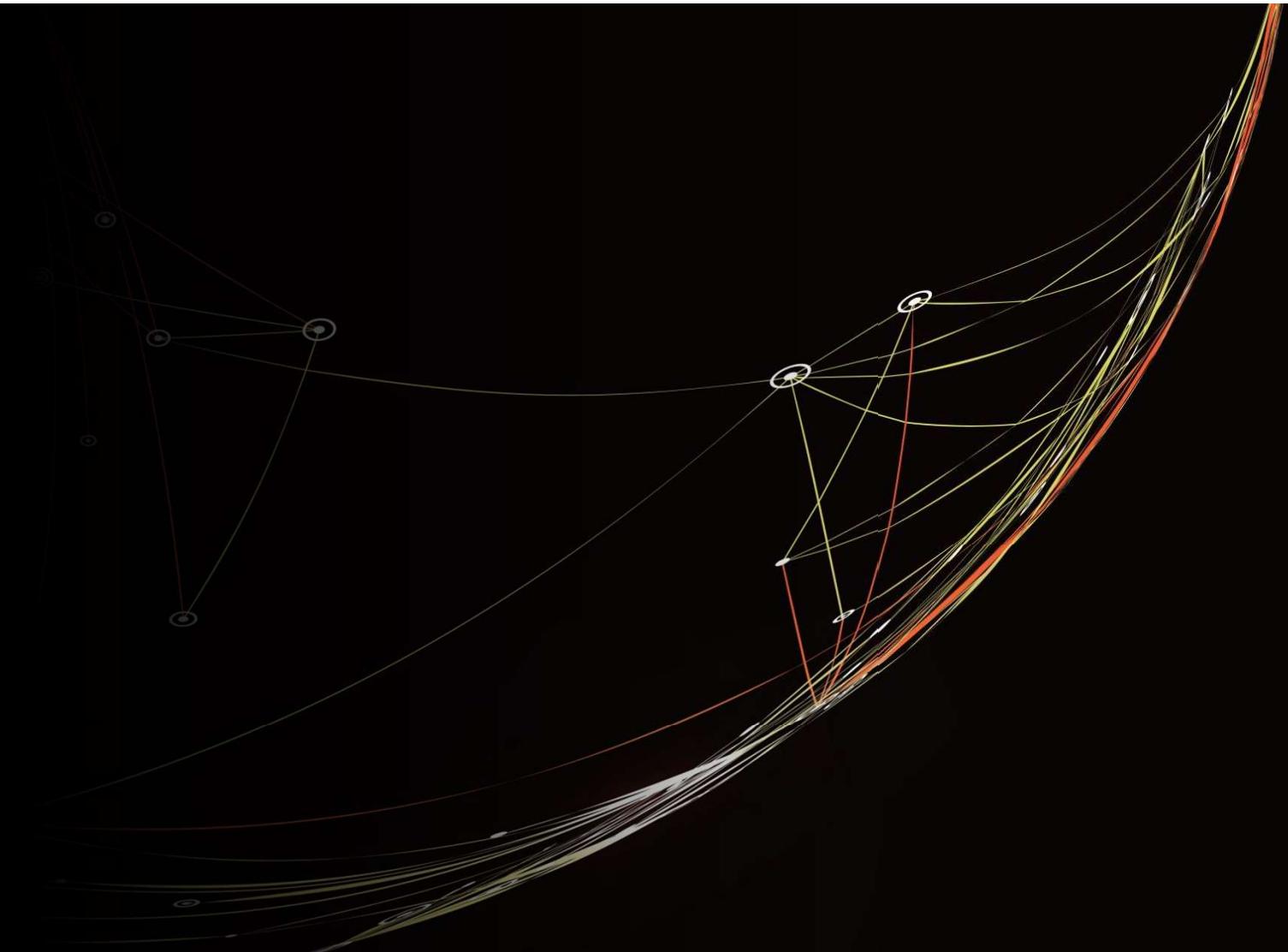
```
PS C:\xampp\htdocs\hdcevents> php artisan route:list
```

Ln 53, Col 1 Spaces: 2 UTF-8



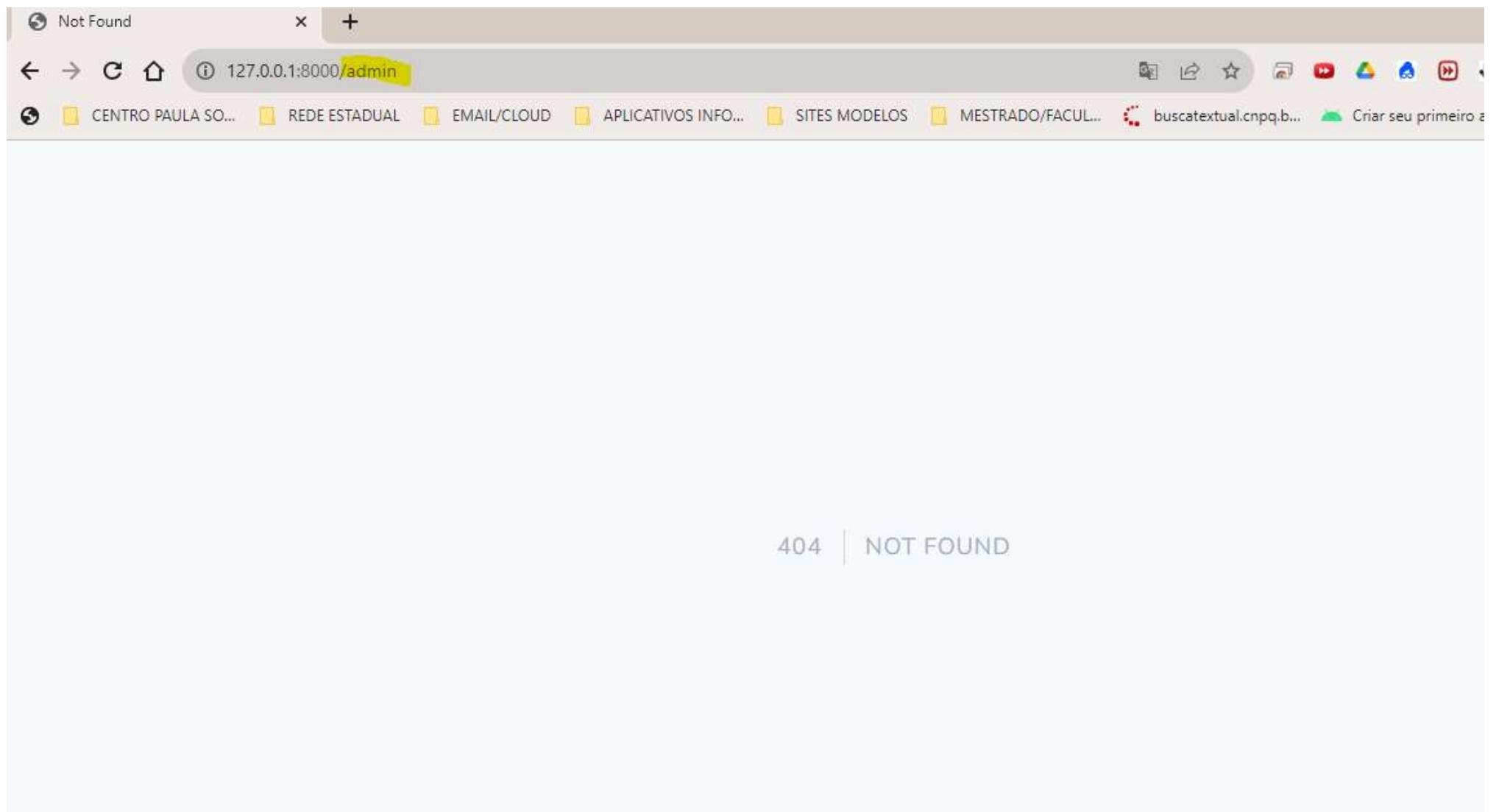
# Rota de contigência

Página não encontrada e  
redirecionamento.

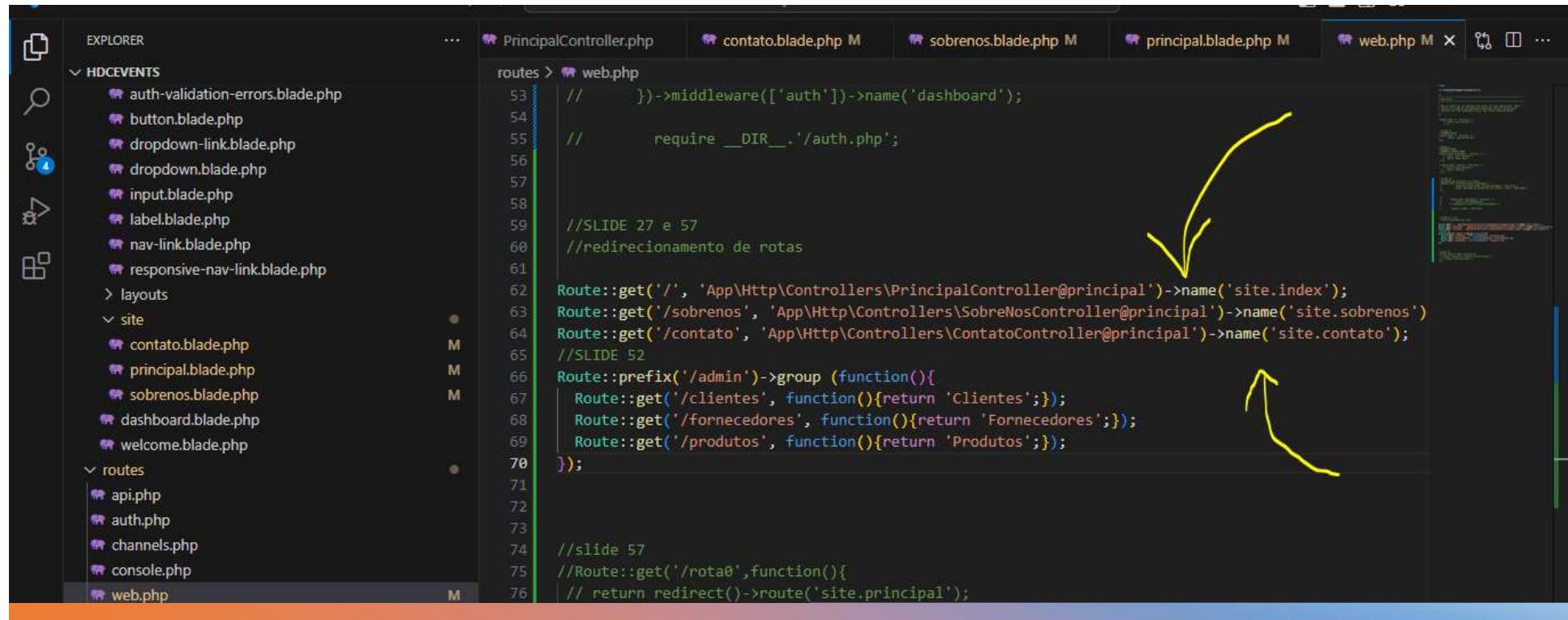


```
routes > web.php
53     // }->middleware(['auth'])->name('dashboard');
54
55     // require __DIR__.'/auth.php';
56
57
58
59 //SLIDE 27 e 57
60 //redirecionamento de rotas
61
62 Route::get('/', 'App\Http\Controllers\PrincipalController@principal')->name('site.index');
63 Route::get('/sobreNos', 'App\Http\Controllers\SobreNosController@principal')->name('site.sobreNos');
64 Route::get('/contato', 'App\Http\Controllers\ContatoController@principal')->name('site.contato');
65 //SLIDE 52
66 Route::prefix('/admin')->group (function(){
67     Route::get('/clientes', function(){return 'Clientes';});
68     Route::get('/fornecedores', function(){return 'Fornecedores';});
69     Route::get('/produtos', function(){return 'Produtos';});
70 });
71
72
73 //slide 57
74 //Route::get('/rota0',function(){
75 //    return redirect()->route('site.principal');
76 //    // }->name('site.principal');
```

Observamos que não existe rota de redirecionamento se digitar na url o localhost/admin e provavelmente o MVC irá mostrar uma página 404



404 | NOT FOUND



```
53 // }->middleware(['auth'])->name('dashboard');
54 // require __DIR__.'/auth.php';
55
56
57
58
59 //SLIDE 27 e 57
60 //redirecionamento de rotas
61
62 Route::get('/', 'App\Http\Controllers\PrincipalController@principal')->name('site.index');
63 Route::get('/sobreNos', 'App\Http\Controllers\SobreNosController@principal')->name('site.sobreNos');
64 Route::get('/contato', 'App\Http\Controllers\ContatoController@principal')->name('site.contato');
65 //SLIDE 52
66 Route::prefix('/admin')->group(function(){
67     Route::get('/clientes', function(){return 'Clientes';});
68     Route::get('/fornecedores', function(){return 'Fornecedores';});
69     Route::get('/produtos', function(){return 'Produtos';});
70 });
71
72
73
74 //slide 57
75 //Route::get('/rota0',function(){
76 //    // return redirect()->route('site.principal');
```

Colocamos apelidos nas rotas

A screenshot of a web browser window. The address bar shows the URL `127.0.0.1:8000/sobreNos`. The page content is titled "SOBRE NÓS (view)" and contains a bulleted list of links:

- [PRINCIPAL](#)
- [SOBRE NÓS](#) ← (This link is highlighted with a yellow circle and arrow)
- [CONTATO](#)



File Edit Selection View Go Run ... ← → 🔍 hcevents

EXPLORER

HDCEVENTS

- > auth
- components
  - application-logo.blade.php
  - auth-card.blade.php
  - auth-session-status.blade.php
  - auth-validation-errors.blade.php
  - button.blade.php
  - dropdown-link.blade.php
  - dropdown.blade.php
  - input.blade.php
  - label.blade.php
  - nav-link.blade.php
  - responsive-nav-link.blade.php
- > layouts
- site
  - contato.blade.php M
  - principal.blade.php M
  - sobreNos.blade.php M
  - dashboard.blade.php
  - welcome.blade.php

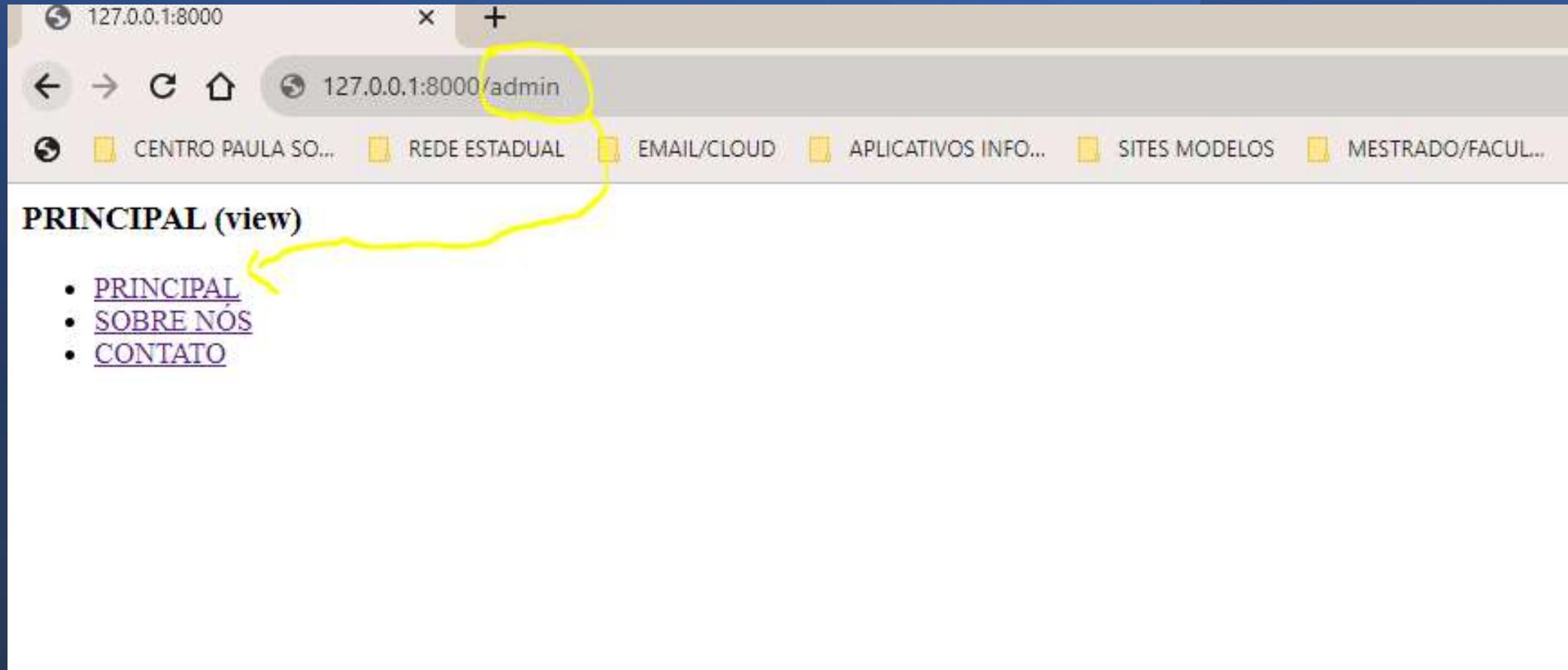
routes > web.php

```
50
51
52
53
54
55
56
57
58
59 //SLIDE 27 e 57
60 //redirecionamento de rotas
61
62 Route::get('/', 'App\Http\Controllers\PrincipalController@principal')->name('site.index');
63 Route::get('/sobreNos', 'App\Http\Controllers\SobreNosController@principal')->name('site.sobreNos');
64 Route::get('/contato', 'App\Http\Controllers\ContatoController@principal')->name('site.contato');
65
66 //SLIDE 52
67 Route::prefix('/admin')->group(function(){
68     Route::get('/clientes', function(){return 'Clientes'});
69     Route::get('/fornecedores', function(){return 'Fornecedores'});
70     Route::get('/produtos', function(){return 'Produtos'});
71 });
72
73 //slide 57
74 Route::get('/admin', function(){
75     return redirect()->route('site.index');
76 });
77
78 
```

hcevents

Criamos uma rota utilizando a função **redirect** onde se digitar uma url inexistente ou forçar um parâmetro ele irá redirecionar para onde você desejar

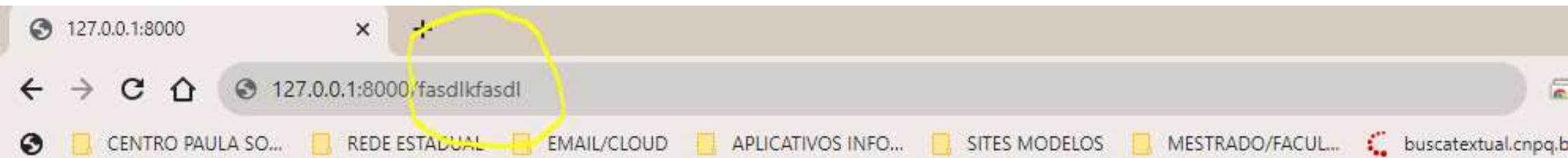
```
[Sat Sep 9 10:30:45 2023] 127.0.0.1:61381 [200]: /favicon.ico
[Sat Sep 9 10:30:52 2023] 127.0.0.1:61384 [200]: /favicon.ico
[Sat Sep 9 10:31:14 2023] 127.0.0.1:61387 [200]: /favicon.ico
[Sat Sep 9 10:31:22 2023] 127.0.0.1:61390 [200]: /favicon.ico
```





# Rota Fallback

Redireciona quando não encontra os parâmetros.



Se digitar uma url que não existe a função **fallback** direciona para a url que eu desejar



a rota não existe clique aqui

File Edit Selection View Go Run ... ← → 🔍 hdcevents

EXPLORER

HDCEVENTS

- > auth
- ✓ components
  - application-logo.blade.php
  - auth-card.blade.php
  - auth-session-status.blade.php
  - auth-validation-errors.blade.php
  - button.blade.php
  - dropdown-link.blade.php
  - dropdown.blade.php
  - input.blade.php
  - label.blade.php
  - nav-link.blade.php
  - responsive-nav-link.blade.php
- > layouts
- ✓ site
  - contato.blade.php M
  - principal.blade.php M
  - sobrenos.blade.php M
  - dashboard.blade.php
  - welcome.blade.php
- ✓ routes
  - api.php
  - auth.php
  - channels.php
  - console.php
  - web.php M

PrincipalController.php contato.blade.php M sobrenos.blade.php M principal.blade.php M

routes > web.php

```
66 Route::prefix('/admin')->group(function(){  
67     Route::get('/clientes', function(){return 'Clientes';});  
68     Route::get('/fornecedores', function(){return 'Fornecedores';});  
69     Route::get('/produtos', function(){return 'Produtos';});  
70 });  
71  
//slide 61  
72 Route::get('/admin',function(){  
73     return redirect()->route('site.index');  
74 } );  
75  
//slide 62  
76 Route::fallback(function() {  
77     echo 'a rota não existe <a href= "'.route('site.index').'"> clique aqui </a> ';  
78 } );  
79  
80  
81  
82  
83  
84  
85  
86
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

[Sat Sep 9 11:29:47 2023] 127.0.0.1:61522 [200]: /favicon.ico  
[Sat Sep 9 11:29:53 2023] 127.0.0.1:61541 [200]: /favicon.ico

The screenshot shows a code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** hdcevents
- Toolbar:** Includes icons for file operations like Open, Save, and Close.
- Explorer:** Shows a tree view of files and folders:
  - HDCEVENTS**: auth, components, application-logo.blade.php, auth-card.blade.php, auth-session-status.blade.php, auth-validation-errors.blade.php, button.blade.php, dropdown-link.blade.php, dropdown.blade.php, input.blade.php, label.blade.php, nav-link.blade.php, responsive-nav-link.blade.php.
  - layouts**
  - site**: contato.blade.php (M), principal.blade.php (M), sobrenos.blade.php (M).
  - routes**: api.php, auth.php, channels.php, console.php, web.php (M).
- Code Editor:** The current file is `routes/web.php`. The code defines routes for an admin section and handles route fallbacks.

```
Route::prefix('/admin')->group(function(){
    Route::get('/clientes', function(){return 'Clientes'});
    Route::get('/fornecedores', function(){return 'Fornecedores'});
    Route::get('/produtos', function(){return 'Produtos'});
});

//slide 61
Route::get('/admin',function(){
    return redirect()->route('site.index');
});

//slide 62
Route::fallback(function() {
    echo 'a rota não existe <a href= "'.route('site.index').'"> clique aqui </a> ';
});
```
- Terminal:** Shows logs from a terminal session:

```
[Sat Sep 9 11:29:47 2023] 127.0.0.1:61522 [200]: /favicon.ico
[Sat Sep 9 11:29:53 2023] 127.0.0.1:61541 [200]: /favicon.ico
```
- Bottom Bar:** PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and a PHP configuration icon.



# Sintaxe Blade

Comentários, @if, @else, @unless, @isset, @empty

```

✓ HDCEVENTS
  auth-session-status.blade.php
  auth-validation-errors.blade.php
  button.blade.php
  dropdown-link.blade.php
  dropdown.blade.php
  input.blade.php
  label.blade.php
  nav-link.blade.php
  responsive-nav-link.blade.php
> layouts
  ●
  ✓ site
    contato.blade.php
    principal.blade.php
    sobrenos.blade.php
    dashboard.blade.php
    welcome.blade.php
  routes > web.php
  60 //ROTA DE MONTAGEM DE PÁGINAS
  61
  62 Route::get('/', 'App\Http\Controllers\PrincipalController@principal')->name('site.index');
  63 Route::get('/sobreNos', 'App\Http\Controllers\SobreNosController@principal')->name('site.sobreNos');
  64 Route::get('/contato', 'App\Http\Controllers\ContatoController@principal')->name('site.contato');
  65 //SLIDE 52
  66 Route::prefix('/admin')->group (function(){
  67   Route::get('/clientes', function(){return 'Clientes';});
  68   // Route::get('/fornecedores', function(){return 'Fornecedores';});
  69   Route::get('/fornecedores', function(){return 'Fornecedores';})->name('admin.fornecedores');
  70   Route::get('/produtos', function(){return 'Produtos';});
  71 });
  72
  73
  74
  75 //slide 61
  76 Route::get('/admin', function(){
    return view('admin.index');
  });

```

## Sintaxe Blade:

Como incluir comentários no Laravel:

- Vamos primeiro implementar um controlador;
- ❑ CONTROLLER: fornecedorController para a
- ❑ ROTA: fornecedores que irá apontar um método para uma
- ❑ VIEW: admin.fornecedores.index;

The screenshot shows a code editor interface with a dark theme. The top bar includes standard menu items like File, Edit, Selection, View, Go, Run, and a search bar containing the text "hdcevents". Below the search bar is a tab bar with five tabs: PrincipalController.php, contato.blade.php M, sobreNos.blade.php M, principal.blade.php M, and web.php M. The "web.php" tab is currently active.

The left sidebar is an "EXPLORER" view showing a file tree. The "HDCEVENTS" folder contains several blade files: auth-session-status.blade.php, auth-validation-errors.blade.php, button.blade.php, dropdown-link.blade.php, dropdown.blade.php, input.blade.php, label.blade.php, nav-link.blade.php, and responsive-nav-link.blade.php. It also includes layouts, site, routes, and storage folders. The "routes" folder contains the "web.php" file, which is the focus of the screenshot.

The main editor area displays the content of the "web.php" file:

```
// redirecionamento de rotas
Route::get('/', 'App\Http\Controllers\PrincipalController@principal')->name('site.index');
Route::get('/sobreNos', 'App\Http\Controllers\SobreNosController@principal')->name('site.sobreNos');
Route::get('/contato', 'App\Http\Controllers\ContatoController@principal')->name('site.contato');
// SLIDE 52
Route::prefix('/admin')->group(function(){
    Route::get('/clientes', function(){return 'Clientes';});
    // Route::get('/fornecedores', function(){return 'Fornecedores';});
    Route::get('/fornecedores', function(){return 'Fornecedores';})->name('admin.fornecedores');
    Route::get('/produtos', function(){return 'Produtos';});
});
// slide 61
Route::get('/admin', function(){
    return redirect()->route('site.index');
});
// slide 62
Route::fallback(function() {
    echo 'a rota não existe <a href="'.route('site.index').'"> clique aqui </a> ';
});
```

The code editor features syntax highlighting for PHP and Blade templating. A vertical green margin line is positioned on the left side of the code. The bottom of the editor has a navigation bar with tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL. On the far right, there are icons for switching between PHP and other languages, opening files, and other editor functions.

The screenshot shows a code editor interface with several tabs open. The active tab is `routes/web.php`, which contains the following PHP code:

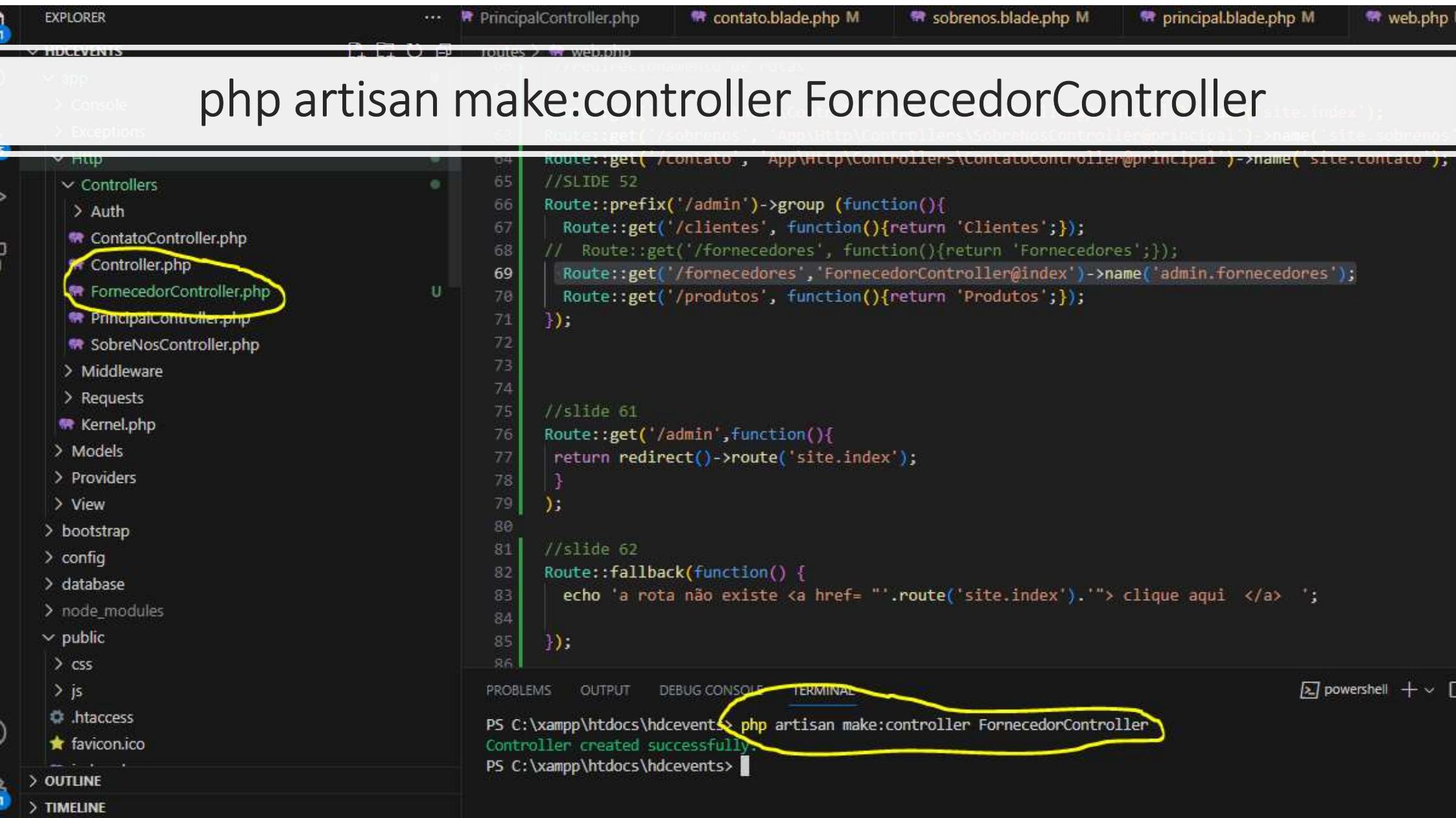
```
00 //redirecionamento de rotas
01
02 Route::get('/', 'App\Http\Controllers\PrincipalController@principal')->name('site.index');
03 Route::get('/sobreNos', 'App\Http\Controllers\SobreNosController@principal')->name('site.sobreNos');
04 Route::get('/contato', 'App\Http\Controllers\ContatoController@principal')->name('site.contato');
05 //SLIDE 52
06 Route::prefix('/admin')->group(function(){
07     Route::get('/clientes', function(){return 'Clientes';});
08     // Route::get('/fornecedores', function(){return 'Fornecedores';});
09     Route::get('/fornecedores', 'FornecedorController@index')->name('admin.fornecedores');
10     Route::get('/produtos', function(){return 'Produtos';});
11 });
12
13 //slide 61
14 Route::get('/admin', function(){
15     return redirect()->route('site.index');
16 });
17
18 //slide 62
19 Route::fallback(function() {
20     echo "A rota não existe. Clique <--> clique aqui ";
21 });
22
23 
```

The code defines various routes, including a main site index, a 'sobreNos' page, a 'contato' page, and an 'admin' section with 'clientes' and 'fornecedores' pages. It also includes a fallback route for non-existent paths.

Below the code editor, the terminal window shows the following log output:

```
[Sun Sep 10 10:41:39 2023] 127.0.0.1:63508 [200]: /favicon.ico
[Sun Sep 10 10:41:44 2023] 127.0.0.1:63511 [200]: /favicon.ico
[Sun Sep 10 10:41:51 2023] 127.0.0.1:63515 [200]: /favicon.ico
```

Troca a função para o controller: FornecedorController



File Edit Selection View Go Run ... ← → ⌂ hdcevents

EXPLORER

- HDCEVENTS
  - > components
  - > layouts
  - > site
    - contato.blade.php
    - principal.blade.php
    - sobreNos.blade.php
    - dashboard.blade.php
    - welcome.blade.php
  - > routes
    - api.php
    - auth.php
    - channels.php
    - console.php
    - web.php
  - > storage
  - > tests
  - > vendor
  - .editorconfig
  - .env
  - .env.example
  - .gitattributes
  - .gitignore
  - / styleci.yml
  - artisan
  - { composer.json
  - { composer.lock
  - hdcevents.sql
  - { package-lock.json
  - { package.json
  - phpunit.xml
  - postcss.config.js
  - README.md
  - server.php

routes > web.php

```
// SLIDE 27 e 57
//redirecionamento de rotas

Route::get('/', 'App\Http\Controllers\PrincipalController@principal')->name('site.index');
Route::get('/sobreNos', 'App\Http\Controllers\SobreNosController@principal')->name('site.sobreNos');
Route::get('/contato', 'App\Http\Controllers\ContatoController@principal')->name('site.contato');
//SLIDE 52
Route::prefix('/admin')->group (function(){
    Route::get('/clientes', function(){return 'Clientes'});
    / Route::get('/fornecedores', function(){return 'Fornecedores'});
    Route::get('/fornecedores', 'App\Http\Controllers\FornecedorController@index')->name('admin.fornecedores');
    Route::get('/produtos', function(){return 'Produtos'});
});;

//slide 61
Route::get('/admin',function(){
    return redirect()->route('site.index');
});
;

//slide 62
Route::fallback(function() {
    echo 'a rota não existe <a href="'.route('site.index').'"> clique aqui </a> ';
});
;
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
Starting Laravel development server: http://127.0.0.1:8000
[Sun Sep 10 11:51:34 2023] 127.0.0.1:64046 [200]: /favicon.ico
[Sun Sep 10 11:51:50 2023] 127.0.0.1:64049 [200]: /favicon.ico
```

```
principal.blade.php M welcome.blade.php web.php M index.blade.php
app > Http > Controllers > FornecedorController.php
1 <?php
2
3 namespace App\Http\Controllers;
4
5 use Illuminate\Http\Request;
6
7 class FornecedorController extends Controller
{
8     public function index(){
9         return view('admin.fornecedor.index');
10    }
11 }
12
13 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
Starting Laravel development server: http://127.0.0.1:8000
[Sun Sep 10 11:25:29 2023] 127.0.0.1:63658 [200]: /favicon.ico
[Sun Sep 10 11:25:42 2023] 127.0.0.1:63680 [200]: /favicon.ico
[Sun Sep 10 11:25:53 2023] 127.0.0.1:63685 [200]: /favicon.ico
```

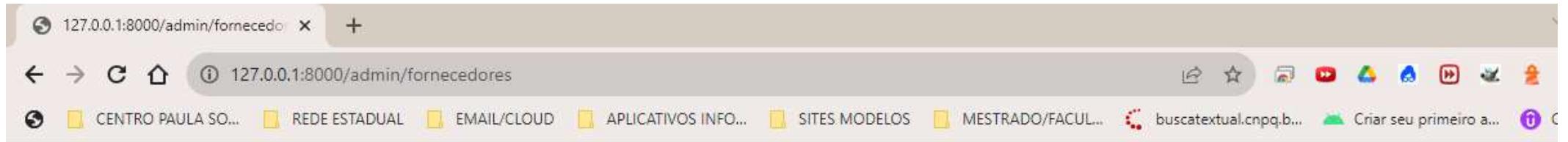
/admin/fornecedor/Index.blade.php

- No FornecedorController você chama a função callback da:
- **FornecedorController@index;**
- Que renderiza a view index.

The screenshot shows a code editor interface with a dark theme. The top navigation bar includes File, Edit, Selection, View, Go, Run, and a three-dot menu. A search bar on the right contains the text "hdcevents". The left sidebar has icons for Explorer, Search, Issues, and Folders. The Explorer view shows a project structure:

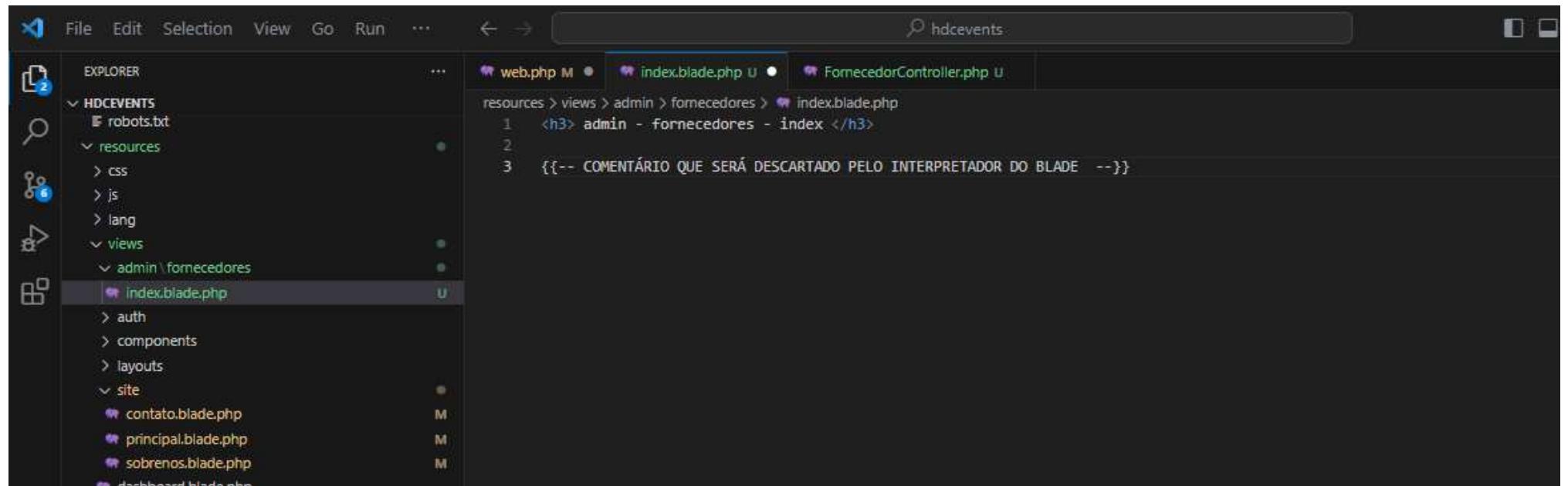
- HDCEVENTS
  - robots.txt
- resources
  - css
  - js
  - lang
  - views
    - admin\fornecedores
      - Index.blade.php
      - auth
      - components
      - layouts
      - site
        - contato.blade.php
        - principal.blade.php
        - sobreos.blade.php
        - dashboard.blade.php
- routes
  - api.php
  - auth.php
  - channels.php
  - console.php

## Criar a pasta views/admin/fornecedor



## admin - fornecedores - index

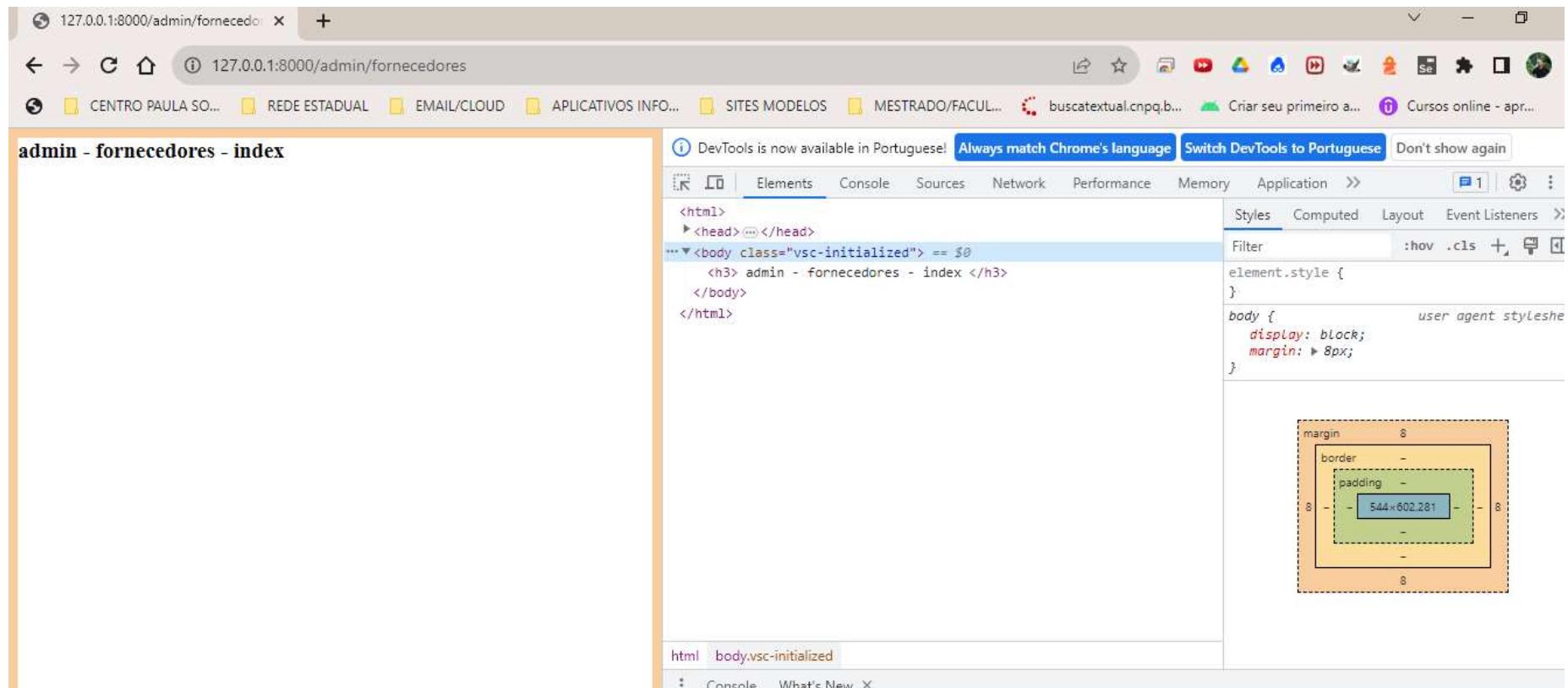
# INSPECIONAR NO NAVEGADOR



The screenshot shows a code editor interface with a dark theme. On the left is the Explorer sidebar, which lists project files and folders. The main area displays a file named `index.blade.php`. The code content is as follows:

```
resources > views > admin > fornecedores > index.blade.php
1 <h3> admin - fornecedores - index </h3>
2
3 {{-- COMENTÁRIO QUE SERÁ DESCARTADO PELO INTERPRETADOR DO BLADE --}}
```

# Não aparece nos Elements da inspeção do navegador



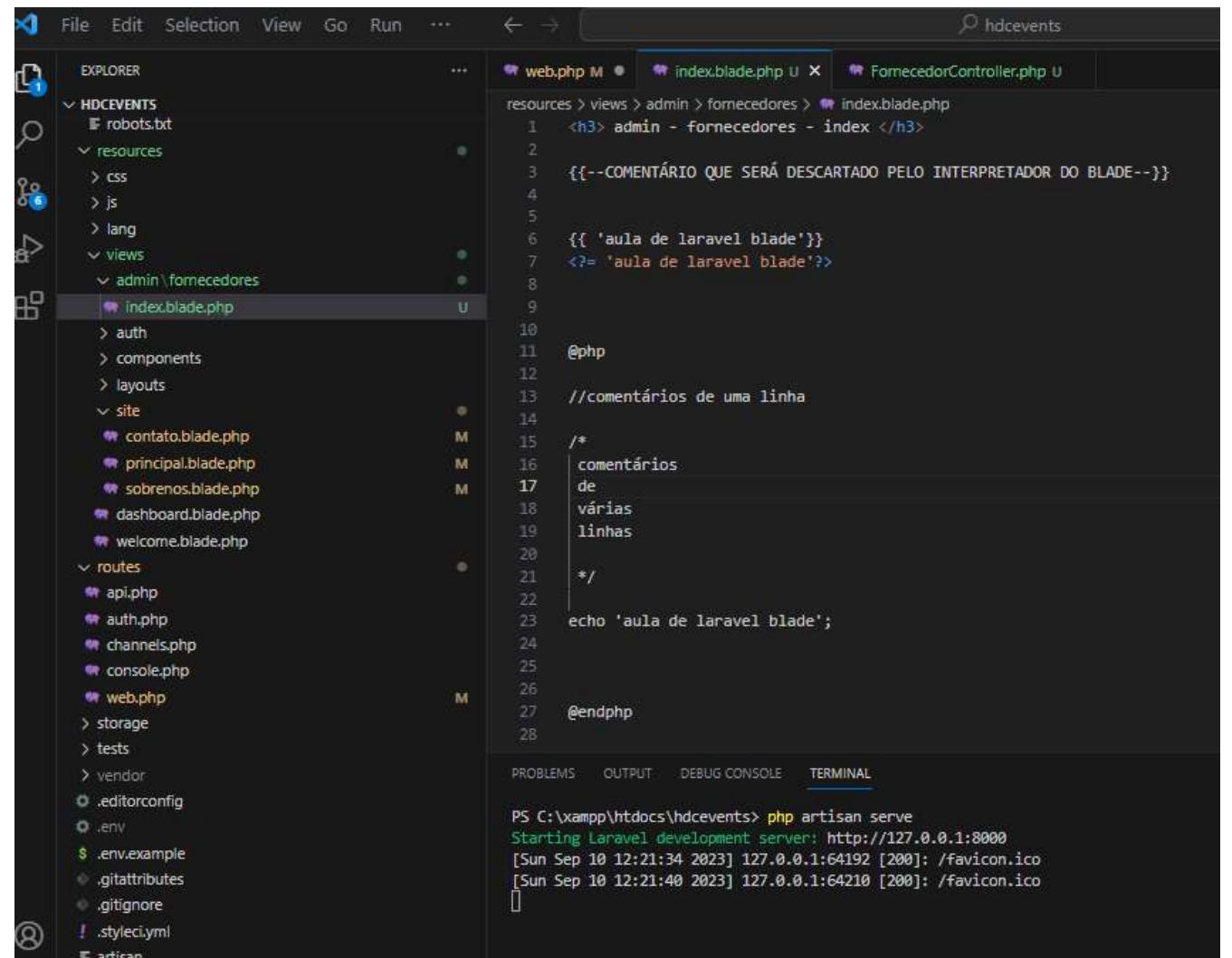
A screenshot of a dark-themed code editor, likely Visual Studio Code, displaying a PHP Blade template file named `index.blade.php`. The file is located in the `resources/views/admin/fornecedores` directory. The code includes various comment types:

```
1 <h3> admin - fornecedores - index </h3>
2
3 {{--COMENTÁRIO QUE SERÁ DESCARTADO PELO INTERPRETADOR DO BLADE--}}
4
5 @php
6
7 //comentários de uma linha
8
9 /*
10  comentários
11  de
12  várias
13  linhas
14 */
15 */
16
17 @endphp
18
```

The code editor interface shows the project structure on the left, with files like `web.php`, `index.blade.php`, and `FornecedorController.php` open in tabs at the top. The bottom navigation bar includes `PROBLEMS`, `OUTPUT`, `DEBUG CONSOLE`, and `TERMINAL`.

# comentários

# Maneiras de comentários



The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows the project structure of a Laravel application named "HDCEVENTS". The "views" directory contains "admin" and "fornecedores" sub-directories. Inside "fornecedores", there is an "index.blade.php" file.
- Code Editor:** The "index.blade.php" file is open. It contains the following Blade code:

```
<h3> admin - fornecedores - index </h3>
<!--COMENTÁRIO QUE SERÁ DESCARTADO PELO INTERPRETADOR DO BLADE--}
{{ 'aula de laravel blade'}}
&lt;?= 'aula de laravel blade'?&gt;

//comentários de uma linha
/*
comentários
de
várias
linhas
*/
echo 'aula de laravel blade';

@endphp</pre>
```
- Terminal:** At the bottom, the terminal window shows the command being run: `PS C:\xampp\htdocs\hdcevents> php artisan serve`. It also displays the output of the server starting: `Starting Laravel development server: http://127.0.0.1:8000` and log entries: `[Sun Sep 10 12:21:34 2023] 127.0.0.1:64192 [200]: /favicon.ico` and `[Sun Sep 10 12:21:40 2023] 127.0.0.1:64210 [200]: /favicon.ico`.

127.0.0.1:8000/admin/fornecedores

127.0.0.1:8000/admin/fornecedores

CENTRO PAULA SO... REDE ESTADUAL EMAIL/CLOUD APlicativos INFO... SITES MODELOS MESTRADO/FACUL... buscatextual.cnpq.b... Criar seu primeiro a...

## admin - fornecedores - index

aula de laravel blade aula de laravel blade aula de laravel blade

DevTools is now available in Portuguese! [Always match Chrome's language](#) [Switch DevTools to Portuguese](#)

Elements Console Sources Network Performance Memory Application >

<html>  
  <head>...</head>  
  ...<body class="vsc-initialized"> == \$0  
    <h3> admin - fornecedores - index </h3>  
    " aula de laravel blade aula de laravel blade aula de laravel  
      blade"  
  </body>  
</html>

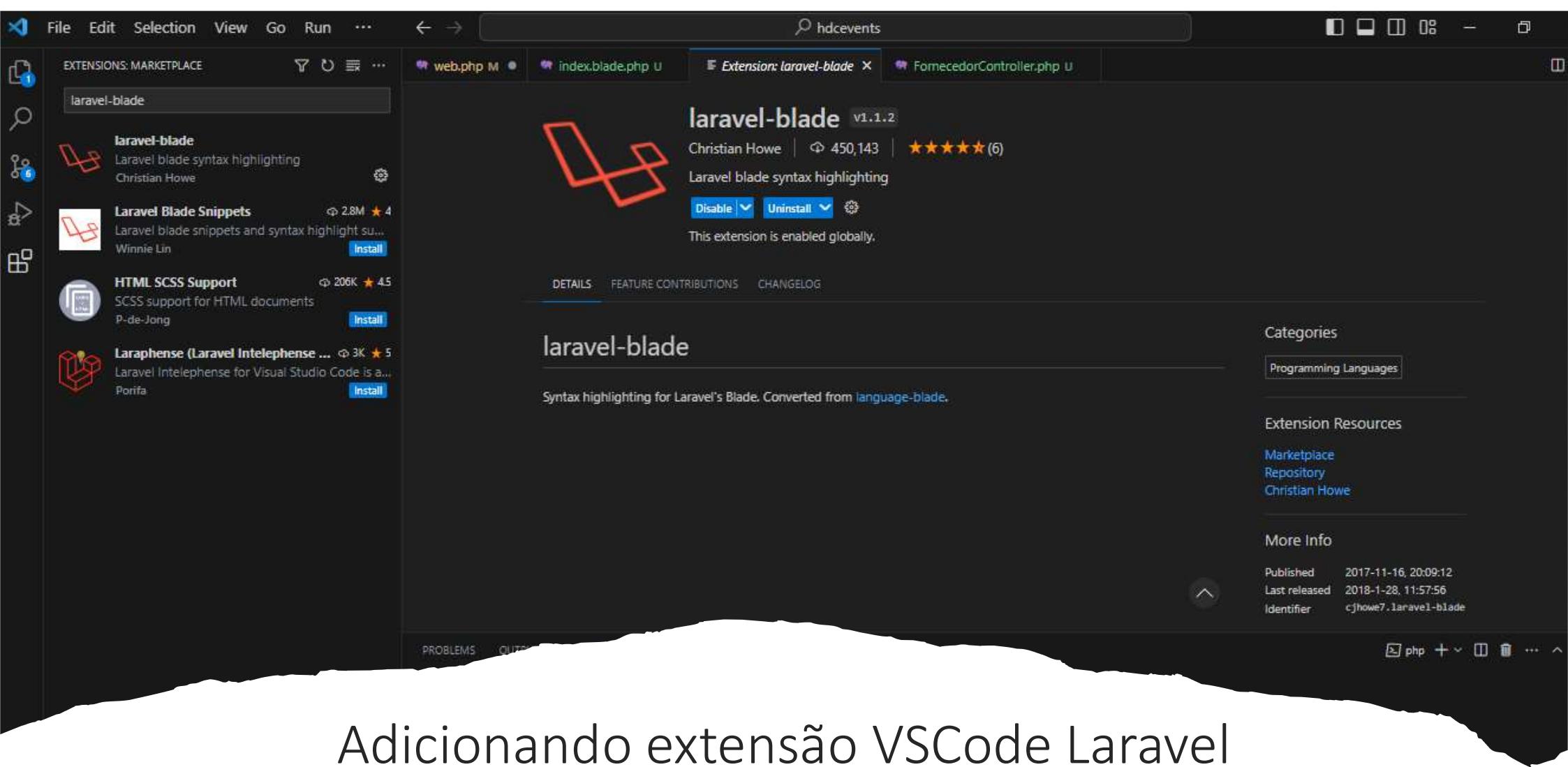
element.style {  
}  
body {  
 display: block;  
 margin: ▶ 8px;  
}

margin  
border  
padding  
8 - 54

html body.vsc-initialized

Console What's New X

Highlights from the Chrome 116 update



# Adicionando extensão VSCode Laravel

# Adicionando extensão VSCode Laravel

The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "HDCEVENTS".
  - public: database, node\_modules, .htaccess, favicon.ico, index.php, mix-manifest.json, robots.txt.
  - resources: css, js, lang, views: admin\fornecedores (containing index.blade.php), auth, components, layouts, site (containing contato.blade.php, principal.blade.php, sobreos.blade.php, dashboard.blade.php, welcome.blade.php).
  - routes: api.php, auth.php, channels.php, console.php, web.php.
- Code Editor:** Displays the content of index.blade.php.

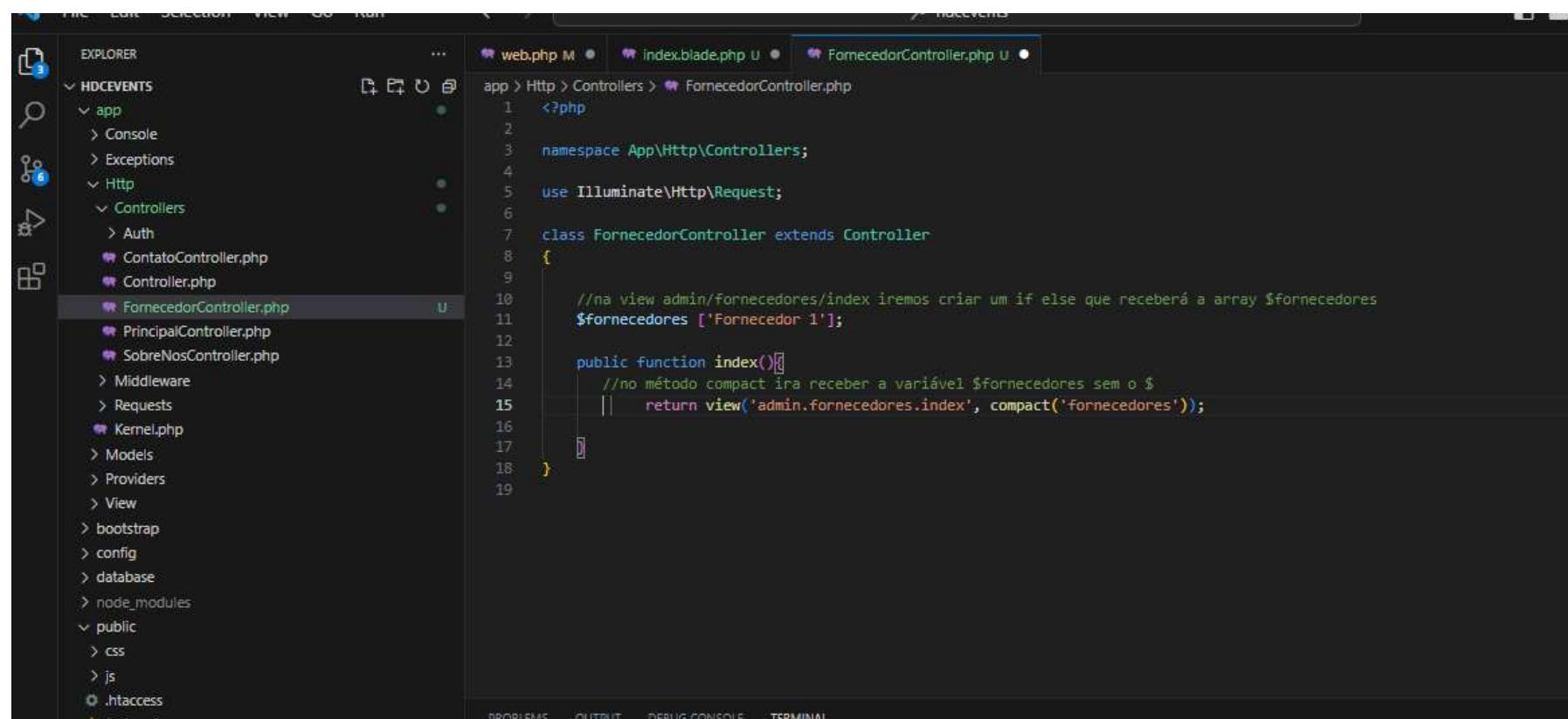
```
<h3> admin - fornecedores - index </h3>
{{-- COMENTÁRIO QUE SERÁ DESCARTADO PELO INTERPRETADOR DO BLADE --}}
{{ 'aula de laravel blade'}}
<?= 'aula de laravel blade'?>
//comentários de uma linha
/*
comentários
de |
várias
linhas
*/
echo 'aula de laravel blade';
@endphp
```
- Terminal:** Shows the command line output:

```
PS C:\xampp\htdocs\hdcevents> php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Sun Sep 10 12:21:34 2023] 127.0.0.1:64192 [200]: /favicon.ico
[Sun Sep 10 12:21:40 2023] 127.0.0.1:64210 [200]: /favicon.ico
```

# Estrutura de decisão

@dd, @unless, @if, @else,

Vamos criar uma variável no Fornecedor Controller para receber um array (\$fornecedor)



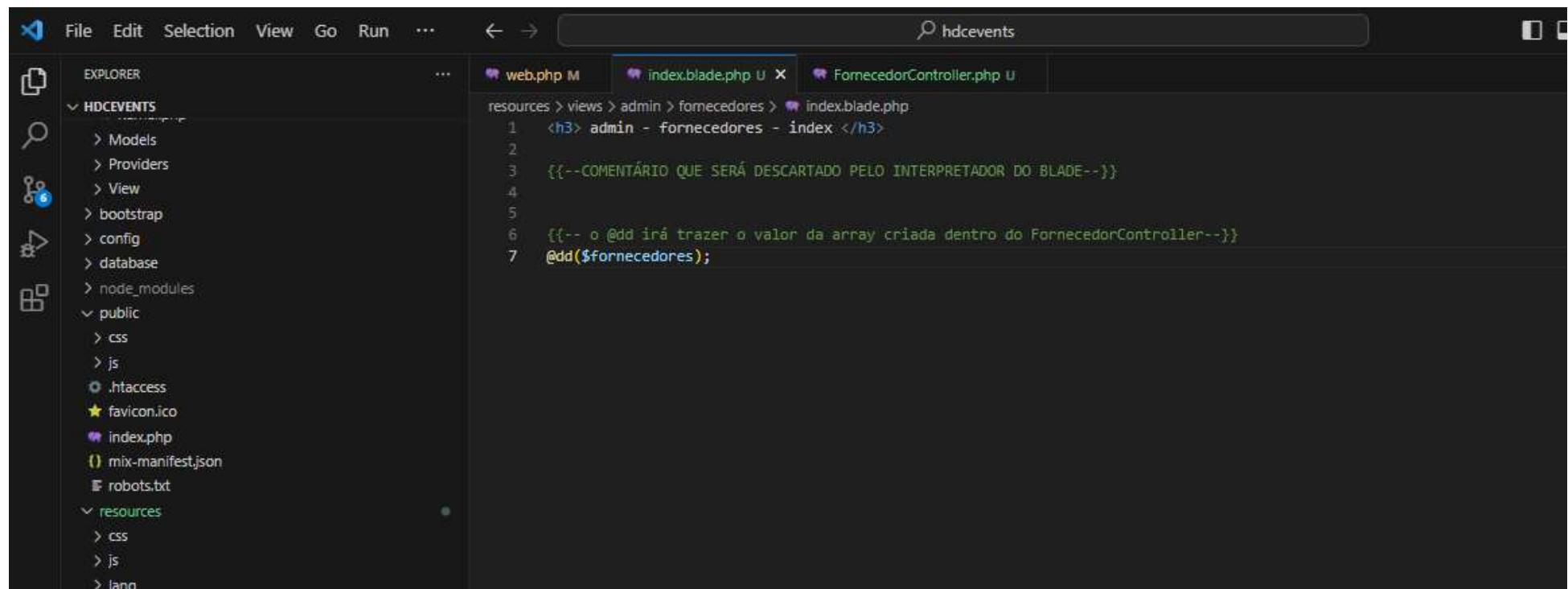
The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "HDCEVENTS". The "app" folder is expanded, showing "Http" and "Controllers". Inside "Controllers", "FornecedorController.php" is selected.
- Code Editor:** Displays the content of `FornecedorController.php`. The code defines a controller class `FornecedorController` extending `Controller`. It includes a comment about creating an array `$fornecedores` in the `index` method if it's not received via compact.
- Bottom Bar:** Shows tabs for "PROBLEMS", "OUTPUT", "DEBUG CONSOLE", and "TERMINAL".

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
class FornecedorController extends Controller
{
    //na View admin/fornecedores/index iremos criar um if else que receberá a array $fornecedores
    $fornecedores ['Fornecedor 1'];

    public function index()
    {
        //no método compact ira receber a variável $fornecedores sem o $ 
        || return view('admin.fornecedores.index', compact('fornecedores'));
    }
}
```

# A concatenação @dd irá trazer as instruções desta variável dentro do Controller



The screenshot shows a code editor interface with the following details:

- File Menu:** File, Edit, Selection, View, Go, Run, ...
- Search Bar:** hdcevents
- Explorer:** Shows the project structure:
  - HDCEVENTS (selected)
  - Models
  - Providers
  - View
  - bootstrap
  - config
  - database
  - node\_modules
  - public
    - css
    - js
    - .htaccess (highlighted)
    - favicon.ico
    - index.php
    - mix-manifest.json
    - robots.txt
  - resources
    - css
    - js
    - lang
- Editor Area:** Displays the `index.blade.php` file content:

```
resources > views > admin > fornecedores > index.blade.php
1 <h3> admin - Fornecedores - index </h3>
2
3 {{-- COMENTÁRIO QUE SERÁ DESCARTADO PELO INTERPRETADOR DO BLADE--}}
4
5
6 {{-- o @dd irá trazer o valor da array criada dentro do FornecedorController--}}
7 @dd($fornecedores);
```

A concatenação @dd irá trazer as instruções desta variável dentro do Controller



The screenshot shows a browser window with the URL `127.0.0.1:8000/admin/fornecedores`. The page title is "admin - fornecedores - index". Below the title, there is a dark red box containing a PHP array dump:

```
array:1 ▾
  0 => "Fornecedor 01"
]
```

The screenshot shows a code editor interface with the following details:

- File Explorer:** Shows the project structure of "HDCEVENTS".
- Active File:** "index.blade.php" (marked with a "U" icon).
- Code Content:** Blade template logic.

```
resources > views > admin > fornecedores > index.blade.php
6   {{--slide 88  o @dd ira trazer o valor da array criada dentro do FornecedorController.php
7
8     @dd($fornecedores);
9
10    --}}
11
12  {{--slide 90    bloco if e else  se acrescentar varios itens na variavel $fornecedores
13
14
15  @if (count($fornecedores) > 0 && count($fornecedores)< 10)
16    <h3>Existem alguns fornecedores cadastrados</h3>
17
18    @elseif(count($fornecedores) > 10)
19      <h3> Existem vários fornecedores cadastrados</h3>
20
21    @else
22      <h3> Não existem fornecedores cadastrados</h3>
23
24  @endif
25
26
27
28
```
- Terminal:** Shows logs from artisan serve.

```
[Sun Sep 10 13:04:58 2023] 127.0.0.1:64560 [200]: /favicon.ico
PS C:\xampp\htdocs\hdcevents> php artisan serve
Starting Laravel development server: http://127.0.0.1:8000
[Sun Sep 10 16:47:59 2023] 127.0.0.1:54806 [200]: /favicon.ico
[Sun Sep 10 16:47:59 2023] 127.0.0.1:54807 [200]: /favicon.ico
[Sun Sep 10 16:51:32 2023] 127.0.0.1:54819 [200]: /favicon.ico
[Sun Sep 10 16:52:45 2023] 127.0.0.1:54827 [200]: /favicon.ico
```

## @if e @else

- @ na sintaxe do Blade serve para comando.
- Inserimos o comando **@if**, **@elseif** e **@else** para mostrar o método COUNT contar a quantidade dentro da variável `$fornecedores`.



## admin - fornecedores - index

Existem alguns fornecedores cadastrados