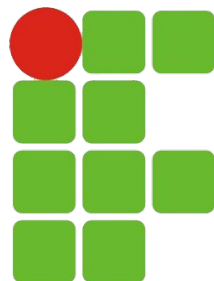


Laravel - conheça seus principais aspectos colocando a mão na massa

Extra: Deploy + SSL

Oi, meu nome é
Jean Jar Pereira de Araújo!





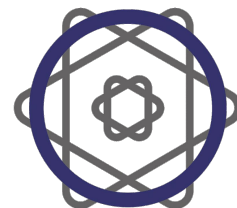
INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE



LAIS
LABORATÓRIO DE INOVAÇÃO
TECNOLÓGICA EM SAÚDE



PPgGIS
Programa de Pós-Graduação em
Gestão e Inovação em Saúde



navi
NÚCLEO AVANÇADO DE
INOVAÇÃO TECNOLÓGICA

Onde me encontrar

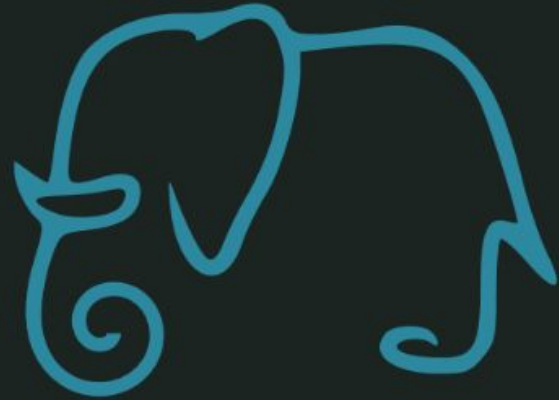
Email: jeanipa@gmail.com

github: <https://github.com/jeanjar>

telegram: <https://t.me/jeanjar>

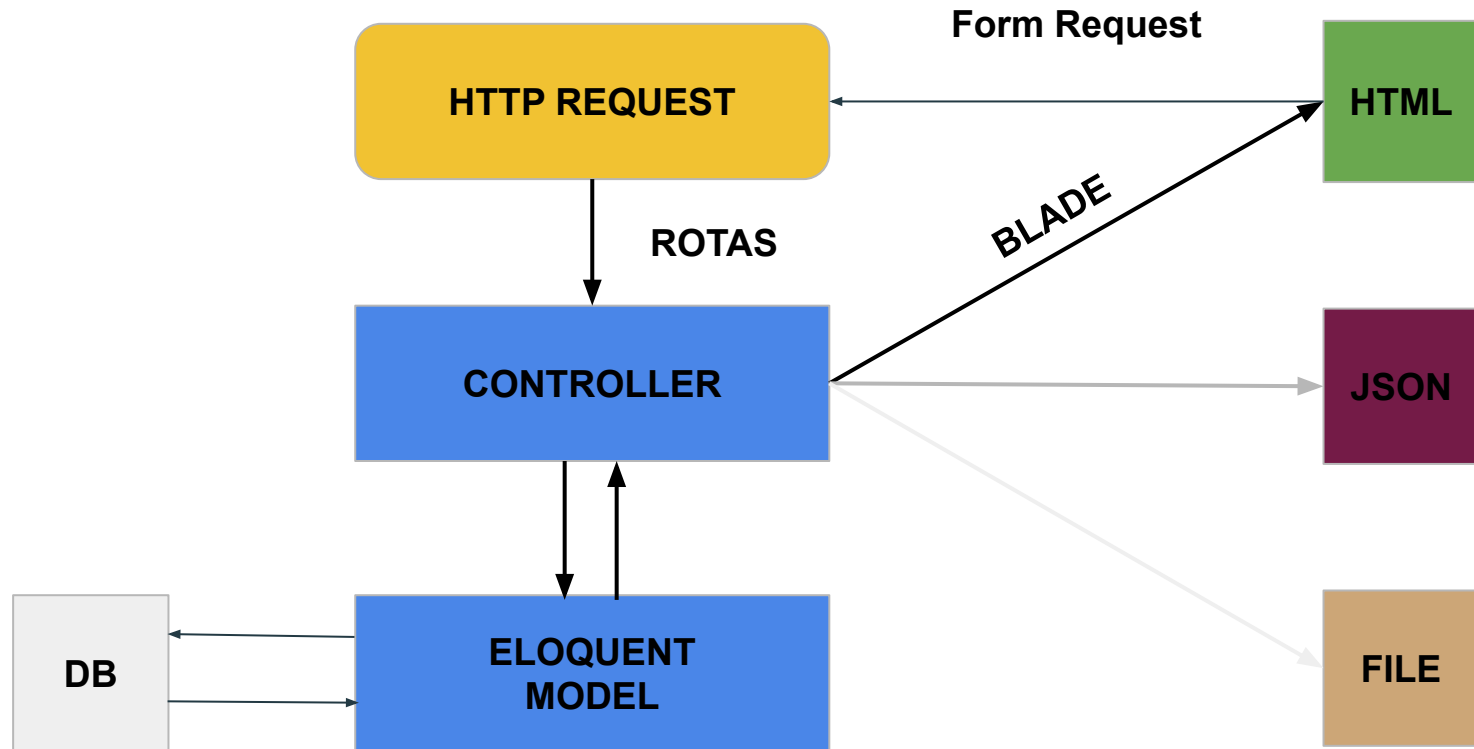
LinkedIn: <https://www.linkedin.com/in/jeanjar/>

Facebook: <https://www.facebook.com/jean.jar.pereira.araujo>



phpnn


Laravel - Ciclo de vida simplificado





Artisan Console

Artisan é uma interface de linha de comando incluída no Laravel, ela fornece vários comandos úteis que podem ajudá-lo enquanto você desenvolve sua aplicação



Artisan Console - Principais Comandos

```
php artisan list
```

Artisan Console - Principais Comandos

```
php artisan tinker
```

```
Psy Shell v0.9.9 (PHP 7.2.17-0ubuntu0.18.04.1 - cli) by Justin Hileman
```

```
>>>
```

Artisan Console - Principais Comandos

```
php artisan make:model Foo
```

Model created successfully.

```
php artisan make:migration foo
```

Created Migration: 2019_05_17_004924_foo

Artisan Console - Principais Comandos

```
php artisan make:model Foo
```

Model created successfully.

```
php artisan make:migration create_foos_table
```

Created Migration: 2019_05_17_005031_create_foos_table

```
php artisan make:factory FooFactory -m Foo
```

Factory created successfully.

Artisan Console - Principais Comandos

```
php artisan make:model Foo -m
```

Model created successfully.

Created Migration: 2019_05_17_010101_create_foos_table

Artisan Console - Principais Comandos

```
php artisan make:controller FooController
```

Controller created successfully.

Artisan Console - Principais Comandos

```
php artisan make:controller FooController -r
```

Controller created successfully.

Artisan Console - Principais Comandos

```
php artisan make:controller \
FooController -r -m Foo
```

Controller created successfully.

Route Model Binding

Artisan Console - Principais Comandos

```
php artisan make:model Foo -crm
```

Model created successfully.

Created Migration: 2019_05_17_013050_create_foos_table

Controller created successfully.

Artisan Console - Principais Comandos

```
php artisan make:model Foo -a
```

Model created successfully.

Factory created successfully.

Created Migration: 2019_05_17_013809_create_foos_table

Controller created successfully.

Artisan Console - Principais Comandos

```
php artisan make:request FooStoreRequest
```

Artisan Console - Principais Comandos

```
php artisan make:model Models/Foo -a
```

```
php artisan make:request Foo/StoreRequest
```

Artisan Console - O PRINCIPAL!

```
php artisan help [comando]
```



Eloquent Laravel ActiveRecord

Eloquent

`Foo::all(): Collection`

`Foo::find($id): Foo`

`Foo::first(): Foo`

`Foo::paginate($perPage = 25): Paginator`

Eloquent - Where Aninhado

```
Foo::where('active', true)  
    ->where('is_admin', false)  
    ->where('foo', '=', 'bar')  
    ->whereIn('id', [1,2,3])  
    ->get()
```


Eloquent - Array de condições

```
Foo::where([  
    ['is_admin', '=', true],  
    ['active', '=', false]  
])  
->get()
```

Eloquent - Agrupamento de Parâmetros

```
Foo::where('is_admin', true)
->where(function(Builder $q){
    $q->where('age', '>', 80);
    $q->orWhere('age' '<' 17),
})->get()
```

Eloquent - whereX

```
Foo::where('email', $email)->first();
```

```
Foo::whereEmail($email);
```

Eloquent - Raw Queries

```
Foo::whereRaw('is_admin = ? and active = ?', [true, true])  
->get()
```

Eloquent - Date Queries

```
Foo::whereRaw('extract(year from created_at) = ?', 2019)  
->whereRaw('extract(month from created_at) = ?', 5)  
->whereRaw('extract(day from created_at) = ?', 18)->toSql()
```

Eloquent - Date Queries

```
Foo::whereYear(2019)->whereMonth(5)->whereDay(18)->get()
```

Eloquent - Query Debug

```
Foo::where('is_admin', true)
->where(function(Builder $q){
    $q->where('age', '>', 80);
    $q->orWhere('age' '<' 17),
})->toSql()
```

```
"select * from `foos` where `is_admin` = ? and (`age` > ? or `age` < ?)"
```

Eloquent - Incremento e Decremento

```
$foo = Foo::find(1);
```

```
$foo->count++;
```

```
$foo->save();
```

```
$foo = Foo::find(1);
```

```
$foo->increment('count');
```


Eloquent - Incremento e Decremento

```
$foo = Foo::find(1);
```

```
$foo->count--;
```

```
$foo->save();
```

```
$foo = Foo::find(1);
```

```
$foo->decrement('count');
```

Eloquent - Incremento e Decremento

```
$foo = Foo::find(1);  
$foo->increment('count', 10);
```

Eloquent - Métodos XorY

```
$foo = Foo::find(1);  
if( ! $foo) {  
    return abort(404);  
}
```

```
$foo = Foo::findOrFail(1);
```

Eloquent - Métodos XorY

```
$foo = Foo::whereEmail('foo@bar.com')->first();  
if( ! $foo) {  
    $foo = Foo::create(['email' => 'foo@bar.com']);  
}
```

```
$foo = Foo::firstOrCreate(['email' => 'foo@bar.com']);
```

Eloquent - Métodos XorY

```
Foo::firstOrCreate([...])
```

```
Foo::firstOrCreate([...])
```

```
Foo::updateOrCreate([...])
```

Eloquent - Query Scope

```
Foo::where('is_admin')->get();  
-----  
public function scopeIsAdmin(Builder $q, $admin = true)  
{  
    $q->where('is_admin', $admin);  
}  
-----  
Foo::isAdmin()->get();
```

Deploy de Aplicação - Manual

1. Acessar servidor Remoto (SSH)
2. Para primeiro deploy:
 - a. Clonar o projeto
 - b. Instalar o projeto
 - c. Configurar variáveis de ambiente (.env)
3. Para os demais deploy's
 - a. atualizar o projeto

Em paralelo já ter o servidor HTTP configurado.

Deploy de Aplicação - Semi-Automático

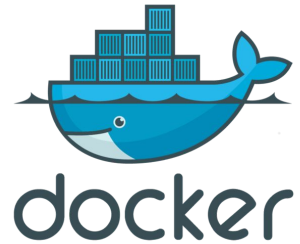
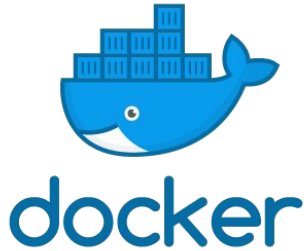
Utilizando o Deployer: <https://deployer.org/>

1. Apenas no primeiro deploy
 - a. `php artisan deploy -s firstdeploy`
 - b. acesso remoto ao servidor e configurar variáveis de ambiente (.env)
2. Para os demais deploy's
 - a. `php artisan deploy`

Deploy de Aplicação - Automático



GitLab





Obrigado!