

# # Envio de Emails com Laravel

D S T Q Q S S

na Controller:

Arr. assoc. com view.



```
Mail::to($email) -> send(new FalsaEmail([ ]));
```

↳ Facade

↳ Classe Mail

Classe Mail:

# php artisan make:mail NomeEmail

↳ Cria uma classe mailable

```
namespace App\Mail;
```

↳ Classe que torna

```
class NomeEmail extends Mailable
```

uma view  
enviável

```
{
```

```
use Queueable, SerializesModels;
```

```
public $data;
```

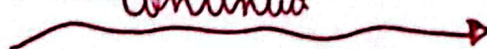
```
public function __construct($data){
```

```
$this->data = $data
```

```
}
```

retendo os dados  
passados no momento  
em que se instancia  
a classe.

Continua



```
public function build() {
    $address = "merely@icbrazil.com";
    $subject = "Shepherd - Tab";
    $name = "Shepherd";
```

seta  
atributos  
do email

→ vem com o corpo

```
$message = $this->view(emails.falha) do email
```

```
→ from($address, $name)
```

```
→ replyto($address, $name)
```

```
→ subject($subject)
```

```
→ with($this->data); → parametros a
```

passar para a

view que tem

ser acessados com

\$nomeVar;

```
return $message;
}
```

## # Manipulando datas com Carbon

- Carbon é uma interface que implem. func. datetime

→ pega a data atual

Ex: \$agora = Carbon::now(); → adiciona

\$amanha = Carbon::now() → add("day", 1);

\$mespassado = Carbon::now() → sub("month", 1)

\* Para mais uses,  
consultar a doc.

↓  
unid. valor  
Subtrai



## # Verificando se é mobile

# composer require reverskies/laravel-fmobile-detect

use: @ desktop

<span>

@ eldesktop

<span>

@ encldesktop

@ mobile

<span>

@ elmobile

<span>

@ enclmobile

\* Para mais uses, consultar a doc

## # Redirecionamento pós middleware

• O tratamento a seguir deve ser implementado quando quisermos que, para acessar uma determinada rota, o usuário se cadastre ou faça login e seja jogado de volta para a página que ele desejava.

No middleware: → Armazena a url que se pretende acessar

Redirect :: set('IntendedUrl', url() → current());

return() → redirect() → route("login"); → Resgata a url em questão

Nos métodos login e cadastro:

if (login) { → Joga para a url pretendida  
return redirect() → intended(route("site")); (se tiver)

}

→ Rota padrão que será disparada quando n houver