

05

## Faça o que eu fiz na aula

Precisamos garantir que todos os envolvidos no projeto sejam informados sobre o andamento das entregas. Vamos então adicionar ao nosso pipeline um job que notificará a todos sempre que tivermos um problema, e um outro job para notificar a todos sempre que uma pipeline se encerra com sucesso, o que significa que temos uma entrega para homologar. Para isso, vamos no arquivo `.gitlab-ci.yml` e vamos adicionar mais um stage, vamos colocar o stage de notificações:

```
image: docker:stable
```

```
stages:
- pre-build
- build
- test
- deploy
- notificacao
```

Agora que temos o stage de notificação, vamos criar 2 jobs no final da pipeline, um para quando tudo der certo, e outro para quando tudo der errado. Então no final do arquivo digite:

```
notificacao-sucesso:
  stage: notificacao
  tags:
  - executor-deploy
  script:
  - echo "tudo deu certo"
```

```
notificacao-falhas:
  stage: notificacao
  tags:
  - executor-deploy
  script:
  - echo "tudo deu errado"
```

Notem que os dois jobs, tanto o `notificacao-sucesso`, quanto o `notificacao-falhas`, estão com o mesmo stage “`notificacao`”. Dessa forma, a tarefa de notificação será executada em paralelo, sem que uma dependa da outra. Na verdade, o que precisamos é que a tarefa de sucesso seja executada quando tudo der certo e a tarefa de falha quando tudo der errado. Então vamos adicionar o atributo “`when`” para as duas tarefas, dessa forma conseguiremos controlar o momento em que deverão ser iniciadas. Para a tarefa “`notificacao-sucesso`” vamos adicionar o “`when: on_success`”:

```
notificacao-sucesso:
  stage: notificacao
  when: on_success
  tags:
  - executor-deploy
  script:
  - echo "tudo deu certo"
```

E para a tarefa de “`notificacao-falhas`” vamos adicionar o “`when: on_failure`”:

```
notificacao-sucesso:  
  stage: notificacao  
  when: on_failure  
  tags:  
    - executor-deploy  
  script:  
    - echo "tudo deu errado"
```

Com as tarefas criadas vamos dar um commit e um push no gitlab e veremos nossa pipeline sendo executada novamente. Caso tudo ocorra sem problemas, a tarefa “notificacao-sucesso” será processada; caso aconteça alguma falha, a tarefa “notificacao-falha” será executada.

```
git add --all  
git commit -m 'adicionando notificações de falhas e de sucesso'  
git push gitlab master
```