

# Como configurar e utilizar o GitLab CI no Linux

GitLab CI (Continuous Integration) é uma ferramenta poderosa que permite automatizar o processo de integração e entrega contínua de software. Com o GitLab CI, você pode definir pipelines que são executados automaticamente em resposta a eventos no seu repositório Git, como commits ou pull requests. Este artigo irá mostrar como configurar e utilizar o GitLab CI em um ambiente Linux, destacando sua importância para desenvolvedores que buscam melhorar a eficiência e a qualidade do processo de desenvolvimento.

## Exemplos:

### 1. Instalação do GitLab Runner no Linux:

Primeiro, vamos instalar o GitLab Runner, que é o agente responsável por executar os jobs definidos no seu pipeline CI/CD.

```
# Adiciona o repositório oficial do GitLab Runner
curl -L --output /usr/local/bin/gitlab-runner https://gitlab-runner-downloads.s3.amazonaws.com/latest/binaries/gitlab-runner-linux-amd64

# Dá permissão de execução
chmod +x /usr/local/bin/gitlab-runner

# Registra o GitLab Runner
gitlab-runner register
```

Durante o processo de registro, você precisará fornecer a URL do GitLab e um token de registro que pode ser encontrado nas configurações do seu projeto no GitLab.

### 2. Configuração do .gitlab-ci.yml:

O arquivo .gitlab-ci.yml é onde você define os pipelines e jobs. Aqui está um exemplo básico:

```
stages:
  - build
  - test
  - deploy

build_job:
  stage: build
  script:
    - echo "Compilando o código..."
```

```
- make
```

```
test_job:  
  stage: test  
  script:  
    - echo "Executando testes..."  
    - make test
```

```
deploy_job:  
  stage: deploy  
  script:  
    - echo "Fazendo deploy..."  
    - ./deploy.sh
```

Neste exemplo, temos três estágios: build, test e deploy. Cada estágio contém um job que executa comandos específicos.

### 3. Executando o Pipeline:

Após configurar o `.gitlab-ci.yml` e registrar o GitLab Runner, qualquer commit no repositório irá desencadear a execução do pipeline definido. Você pode acompanhar o progresso e os resultados dos jobs diretamente na interface do GitLab.

```
git add .  
git commit -m "Adiciona configuração do GitLab CI"  
git push origin main
```

### 4. Monitorando e Depurando Jobs:

Para monitorar e depurar jobs, você pode acessar a seção "CI/CD" no seu projeto GitLab. Lá, você encontrará logs detalhados de cada job executado, o que facilita a identificação de problemas e a tomada de ações corretivas.