

Использование шаблонизации в gitlab-ci

CI/CD на основе GitLab



Проверить, идет ли запись

Меня хорошо видно & слышно?






Федосеев Вячеслав

TeamLead DevOps

- Был админом еще в 2007, но его больше не вернуть
- С 2015 работал с highload, понял зачем нужен DevOps
- Разработал несколько курсов на разных платформах
- Выступаю наставником на курсах по DevOps
- Руководитель курса CI/CD на основе GitLab в OTUS

 @msfixer

 msfixer@msfixer.ru

 <https://git.notoxops.com>

Правила вебинара



Активно
участвуем



Off-topic обсуждаем
в учебной группе **#OTUS CICD-GitLab-2025-04**



Задаем вопрос
в чат или голосом



Вопросы вижу в чате,
могу ответить не сразу

Карта курса



СТАРТ

CI/CD - системы,
подходы и workflow

Gitlab CI

Безопасность

Проектная работа

ФИНИШ



Маршрут вебинара

Зачем нужно и почему важно

Предлагаемые варианты

LIVE: примеры использования

Рефлексия

Цели вебинара

К концу занятия вы сможете

1. Будете знать зачем нужна шаблонизация
2. Будете знать варианты предлагаемой шаблонизации в gitlab-ci
3. Сможете определять оптимальный вариант подхода для вашего проекта
4. Будете знать про проблемы, которые могут встретиться на пути



**Давайте сразу: а точно оно
нужно?**



Зачем оно нужно

И нужно ли мне





<https://pragprog.com/titles/tpp20/the-pragmatic-programmer-20th-anniversary-edition/>

DRY



Предлагаемые варианты

YAML anchors

```
.test_template: &test_suite
  image: ruby:2.6

unit_test:
  <<: *test_suite
  script:
    - echo "Running a test here"

end_to_end_test:
  <<: *test_suite
  script:
    - echo "Running a test here"

smoke_test:
  <<: *test_suite
  script:
    - echo "Running a test here"
```

Include

```
include:  
  - template: CI/Build.gitlab-ci.yml  
  - template: CI/Test.gitlab-ci.yml  
  - template: CI/Security.gitlab-ci.yml  
  - template: CD/Deploy.gitlab-ci.yml
```

Include

- On the same line:

```
YAML   
include: 'my-config.yml'
```

- As a single item in an array:

```
YAML   
include:  
  - 'my-config.yml'
```

Include

YAML



```
include:  
  - 'https://gitlab.com/awesome-project/raw/main/.before-script-template.yml'  
  - 'templates/.after-script-template.yml'  
  - template: Auto-DevOps.gitlab-ci.yml  
  - project: 'my-group/my-project'  
    ref: main  
    file: 'templates/.gitlab-ci-template.yml'
```

extends


```
.prepare_deploy:
  stage: deploy
  script:
    - echo "I am preparing the deploy"
  only:
    - main

deploy_to_dev:
  extends: .prepare_deploy
  script:
    - echo "Deploy to dev environment"
  environment: dev

deploy_to_production:
  extends: .prepare_deploy
  script:
    - echo "Deploy to production environment"
  when: manual
  environment: production
```

extends

- `included.yml:`

```
YAML   
  
.template:  
  script:  
    - echo Hello!
```

- `.gitlab-ci.yml:`

```
YAML   
  
include: included.yml  
  
useTemplate:  
  image: alpine  
  extends: .template
```

reference

```
.vars:
  variables:
    DEV_URL: "http://dev-url.com"
    STAGING_URL: "http://staging-url.com"

.setup_env:
  script:
    - echo "Creating Environment"

.teardown_env:
  after_script:
    - echo "Deleting Environment"

integration_test:
  variables: !reference [.vars, variables, DEV_URL]
  script:
    - !reference [.setup_env, script]
    - echo "Run Test"
  after_script:
    - !reference [.teardown_env, after_script]

performance_test:
  variables: !reference [.vars, variables]
  script:
    - !reference [.setup_env, script]
    - echo "Run Test"
  after_script:
    - !reference [.teardown_env, after_script]
```

Downstream pipeline

https://docs.gitlab.com/ci/pipelines/downstream_pipelines/#trigger-a-downstream-pipeline-from-a-job-in-the-gitlab-ci-yml-file

```
trigger_job:
  trigger:
    include:
      - local: path/to/child-pipeline.yml
```

```
microservice_a:
  trigger:
    include:
      - local: path/to/microservice_a.yml
      - template: Jobs/SAST.gitlab-ci.yml
      - project: 'my-group/my-pipeline-library'
        ref: 'main'
        file: '/path/to/child-pipeline.yml'
```

components

<https://docs.gitlab.com/ci/components/>

steps

<https://docs.gitlab.com/ci/steps/>

Вопросы

 если есть вопросы

 если вопросов нет

LIVE

Пример использования



Вопросы



если есть вопросы



если вопросов нет

**Подведём
итоги занятия**

Теперь вы сможете:

Отправьте в чат номера достигнутых целей

1. Знаете, зачем нужна шаблонизация

2. Знаете какие есть инструменты в gitlab-ci для шаблонизации

3. Понимаете как практически применять





Отправьте в чат эмодзи, который отражает ваше настроение после занятия



Продолжите высказывание: теперь я умею/знаю...



Что планируете использовать в работе?

Заполните, пожалуйста, опрос о занятии

Мы читаем все ваши сообщения
и берем их в работу 🖋️❤️

Приглашаем на следующий вебинар

03 июня 2025

Репозиторий с ansible



Ссылка на вебинар будет
в ЛК за 15 минут



Материалы
к занятию в ЛК —
можно изучать



Обязательный материал
обозначен красной
лентой



ЭТОТ КОНТЕНТ КУПЛЕН НА САЙТЕ **SKLADCHIK.ORG**



**ТЫ ЕЩЕ
НЕ В КЛУБЕ?**
ПРИСОЕДИНЯЙСЯ
И НАЧИНАЙ ЭКОНОМИТЬ!



ЧТО ТАКОЕ КЛУБ «СКЛАДЧИК»?



Платформа

Это платформа, где каждый день тысячи людей собираются вместе, чтобы общими усилиями находить, приобретать и изучать курсы интересные именно им.



Сообщество

Это сообщество из 500 000 людей, открывших для себя выгодный способ быть в тренде самых актуальных и получать ценные знания по минимальной цене.



Библиотека

Это крупнейшая библиотека инфопродуктов в которой можно найти практически любой курс или тренинг продававшийся за последние 10 лет, а также уникальные авторские инфопродукты, которые не получится найти больше нигде.

