

# ОСНОВЫ CI/CD. Знакомство с GitLab.

CI/CD

# Что будет на уроке

1. Узнаем, что такое release engineering и CI/CD.
2. Познакомимся с архитектурой GitLab и основными компонентами.
3. Разберёмся, что такое pipeline.

**Release engineering** или **релизы** — разработка и доставка программного обеспечения.

**SRE (Site Reliability Engineer)** или **релиз-инженеры** определяют все шаги, необходимые для выпуска ПО.

## Popular Employer Salaries for Site Reliability Engineer (SRE)



За первое полугодие 2020 года  
зарплаты айтишников выросли  
на 8%.

Больше всего выросли  
зарплаты администраторов  
(+23%).

[Источник](#)

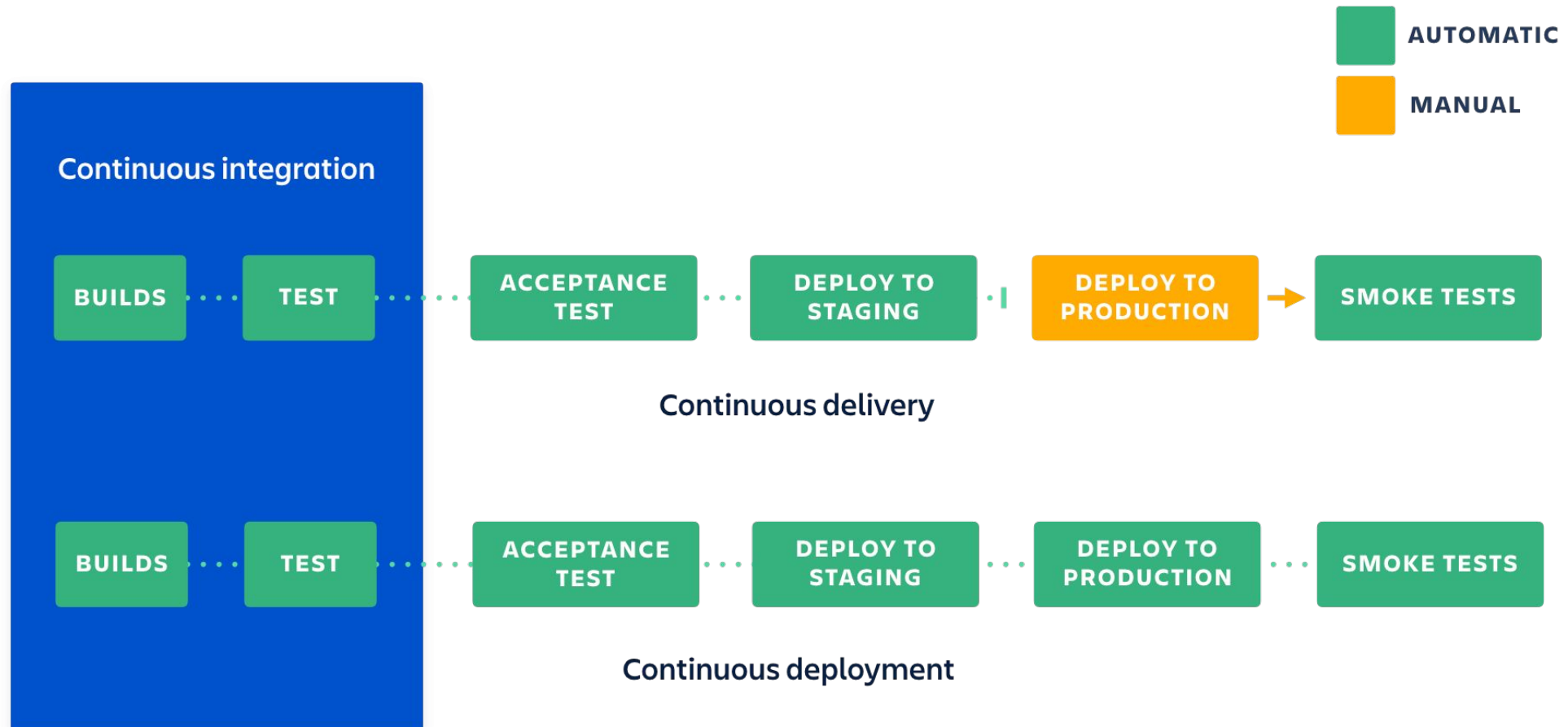


# Что такое CI/CD

**CI/CD** — концепция непрерывной интеграции и доставки кода на продакшн сервера.

**CI (Continuous Integration)** — непрерывная интеграция.

**CD (Continuous Delivery и Continuous Deployment)** — непрерывная доставка и непрерывное развёртывание соответственно.



# Знакомство с GitLab

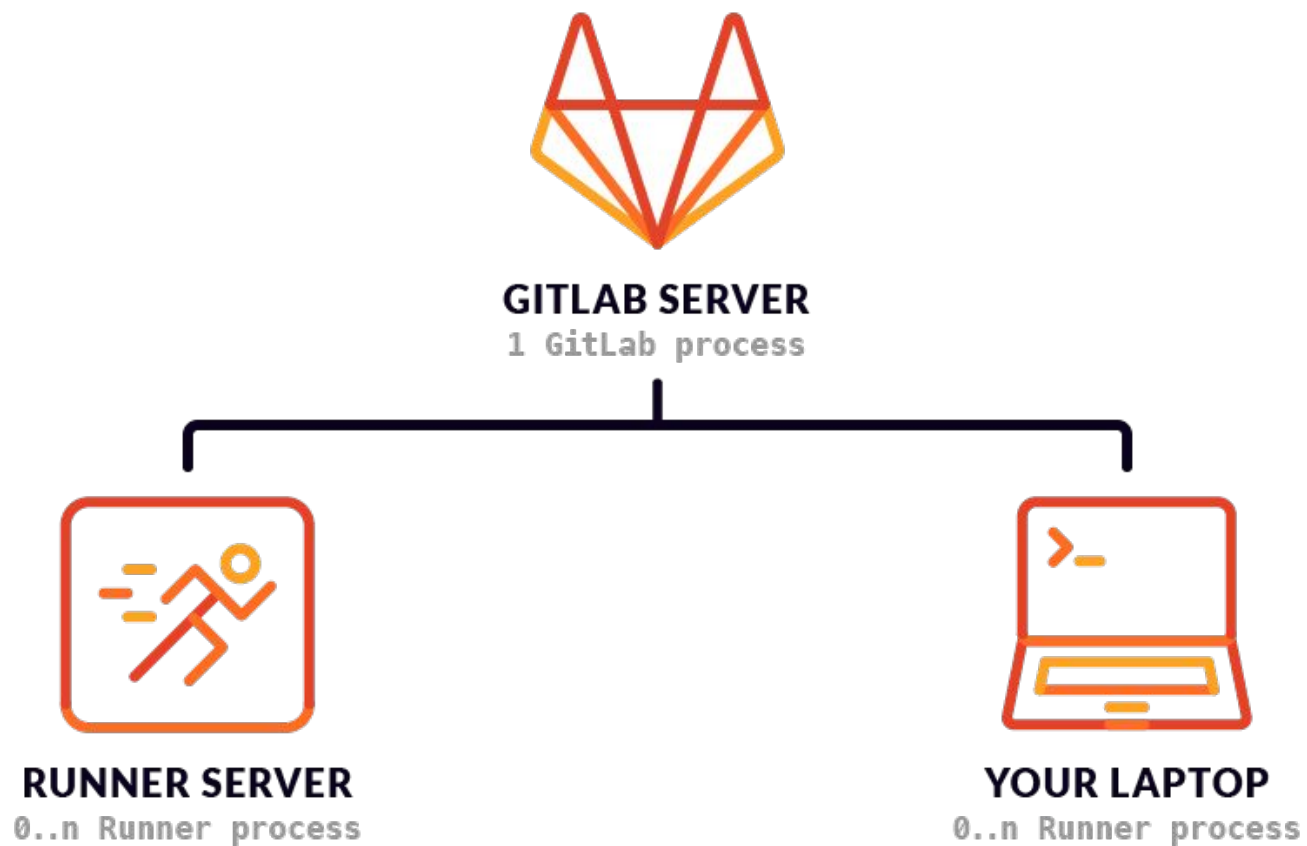
Веб-инструмент жизненного цикла DevOps с открытым исходным кодом.

Состоит из:

- системы управления репозиториями кода для Git с собственной вики;
- системы отслеживания ошибок;
- CI/CD-пайплайнов и др.



# Архитектура GitLab



## GitLab server

**Gitaly** — сервис, предоставляющий доступ к git-репозиториям. Без него компоненты GitLab не смогут читать или записывать данные в git.

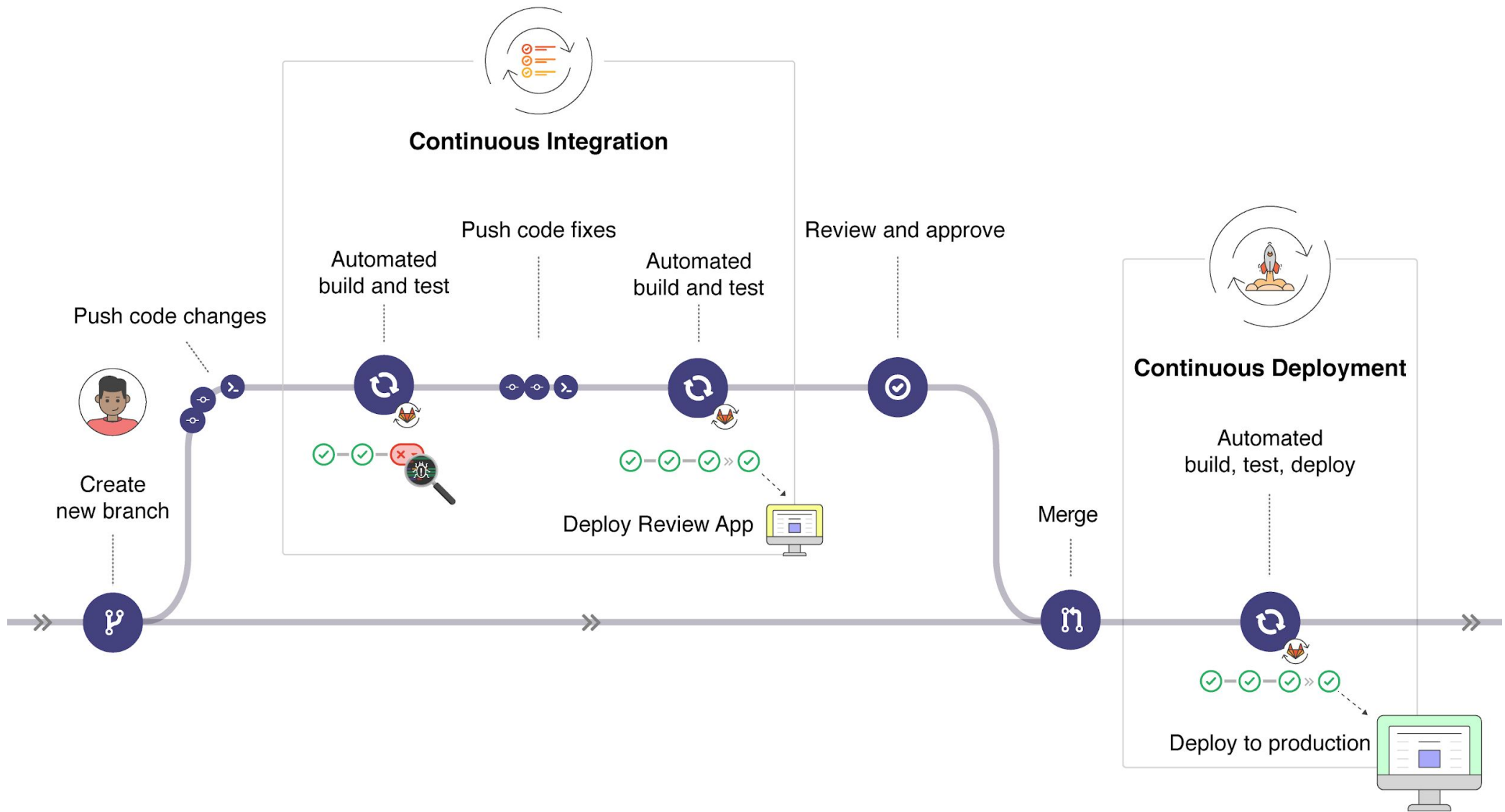
**PostgreSQL** — база данных.

**GitLab Pages** позволяет публиковать статические сайты напрямую из репозитория GitLab.

## GitLab runner

Сервис, который выполняет job'ы и отправляет результат обратно в GitLab server.

Обычно такие сервисы запущены на выделенных машинах.



# Pipeline

Pipeline описывается в файле **gitlab\_ci.yml**, который лежит в корне репозитория.

**Pipeline (пайплайн)** — это высокоуровневый компонент CI/CD.

Пайплайн состоит из:

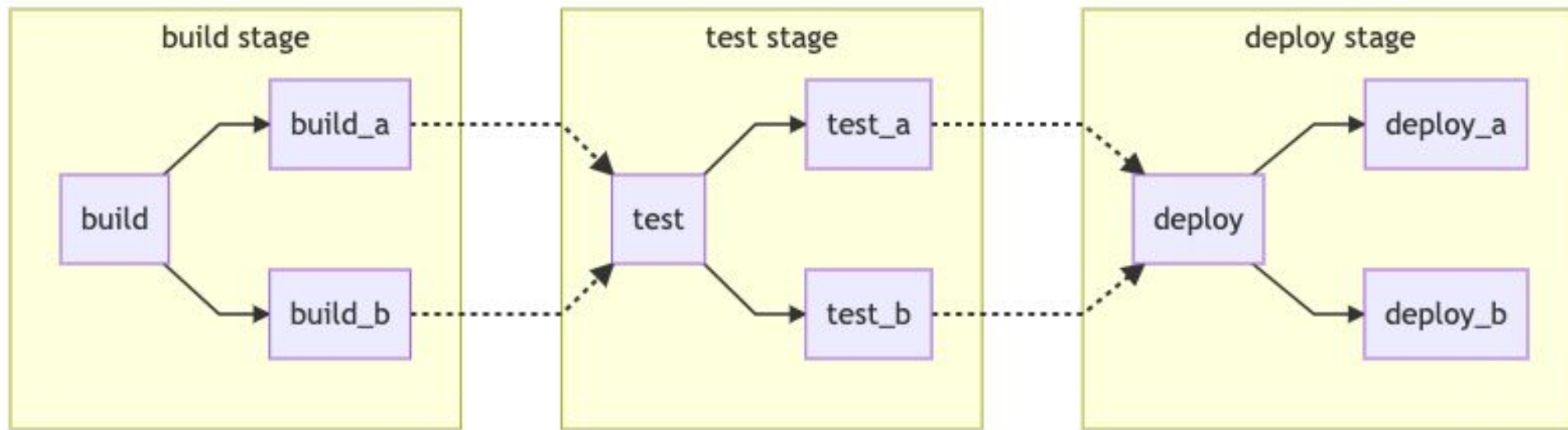
1. **Jobs**, которые определяют **что** нужно сделать. Например, скомпилировать или протестировать код.
2. **Stages**, которые определяют **когда** job'ы должны быть запущены. Например, stage, который запускает тесты после stage компиляции кода.

# Pipeline

Типичный pipeline состоит из четырёх stage, выполняющихся в следующем порядке:

- build stage, с job compile;
- test stage, с job'ами тестов (test1, test2);
- staging stage, с job deploy-to-stage;
- production stage, с job deploy-to-prod.

# Базовый pipeline



Если все job'ы в stage успешно отработали, pipeline переходит на следующий stage.

Если хоть одна job сфейлилась, следующий stage обычно **не** выполняется и pipeline завершается ошибкой.

## gitlab-ci.yml pipeline

диаграммы с  
предыдущего слайда

```
1 stages:
2   - build
3   - test
4   - deploy
5
6 image: alpine
7
8 build_a:
9   stage: build
10  script:
11    - echo "This job builds something."
12
13 build_b:
14   stage: build
15   script:
16     - echo "This job builds something else."
17
18 test_a:
19   stage: test
20   script:
21     - echo "This job tests something. It will only run when all jobs in the"
22     - echo "build stage are complete."
23
24 deploy_a:
25   stage: deploy
26   script:
27     - echo "This job deploys something. It will only run when all jobs in the"
28     - echo "test stage complete."
```