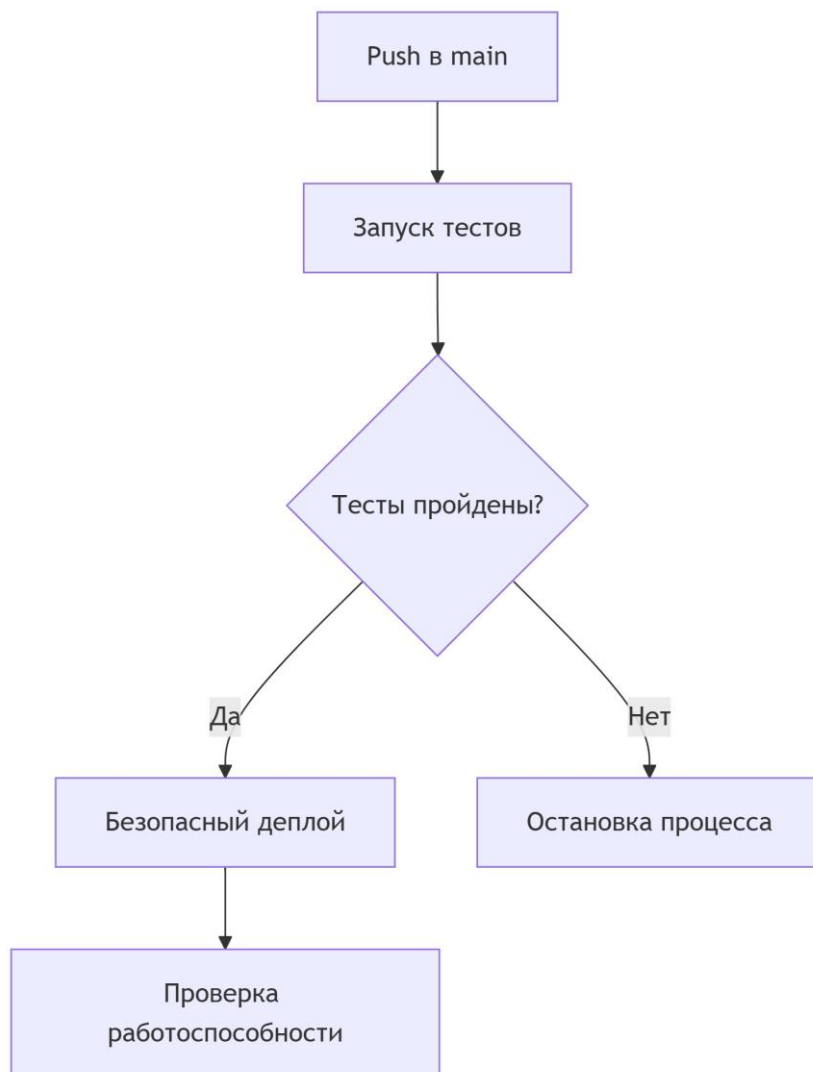


Документация CI/CD Pipeline

На данный момент CI/CD пайплайн, реализованный с помощью Github Actions, применяется командой Reflect в разработке Django-backend приложения для упрощения процесса разработки, так как он обеспечивает автоматизированное тестирование развертывание приложения на сервере

Пайплайн действует по следующей схеме работы:



Тестирование и деплой выполняются последовательно, что позволяет на раннем этапе выявить проблемы в коде и не тратить время на ручной запуск

Также используется безопасный деплой с бэкапом и временной папкой для новой версии, это позволяет избежать утери предыдущей версии файлов сервера при ошибке и последующей остановке пайплайна, замена версий происходит атомарно (одной операцией), а бэкап удаляется только после успешного запуска

Для доступа к серверу используются SSH ключи и Github секреты:

- **SSH_HOST** - IP сервера
- **SSH_USER** - пользователь (goida)
- **SSH_PRIVATE_KEY** - приватный SSH-ключ
- **SSH_KNOWN_HOSTS** - fingerprint сервера

Как упоминалось выше, пайплайн разделён на два этапа:

1. Тестирование

#краткое изложение пайплайна тестирования

```
test:
  runs-on: ubuntu-latest
  services:
    postgres: ... # Тестовая БД
  steps:
    - checkout
    - setup-python
    - install-deps
    - run-tests
```

Здесь разворачивается изолированная среда с PostgreSQL, устанавливаются зависимости Python и запускаются Django тесты

2. Деплой

#краткое изложение пайплайна тестирования

```
deploy:
  needs: test
  steps:
    - secure-copy
    - backup-current
    - deploy-new-version
    - verify-deployment
```

Здесь происходит копирование файлов коммита во временную папку (.new), создание бэкапа текущей версии (.backup), происходит замена версий, запуск docker-контейнеров и проверка работоспособности

Для использования необходимо настроить права доступа и SSH-ключи