

Andersen DevOps exam

Siarhei Burakouski, 30.07.2021

Задача

Построить рабочую инфраструктуру с автоматической сборкой и деплоем новой версии приложения с использованием докеризации.

В качестве приложения разработайте 2 **web** приложения, которые выводят “Hello World 1” & “Hello World 2” на языках из списка: *Python, .NET Core, PHP, Java, Go (gin framework)*.

- Ci/CD - выбираете сами.
- SCM/ControlVersion - выбираете сами (git based).
- Registry - выбираете сами.
- Инфраструктура - выбираете сами.

Презентация должна содержать в себе схему инфраструктуры и 2 диаграммы: как устроен процесс Ci/CD в вашем решении и как бы вы его строили в идеальном мире и без ограничений по времени исполнения задачи.

Дополнительные плюсы при решении задачи

- Проверка кода анализатором, таким как snyk, sonarqube, linter, etc.
- Нотификация по результатам
- Автоматический запуск pipeline на commit/merge request
- Secret management
- Pipeline as code
- Политика контейнеров в докер реджистри (плановые удаление и тд)
- Использование сервисов по сбору логов
- Проверка автоматических тестов приложения (если реализуете)
- Написать README.md

Приложения

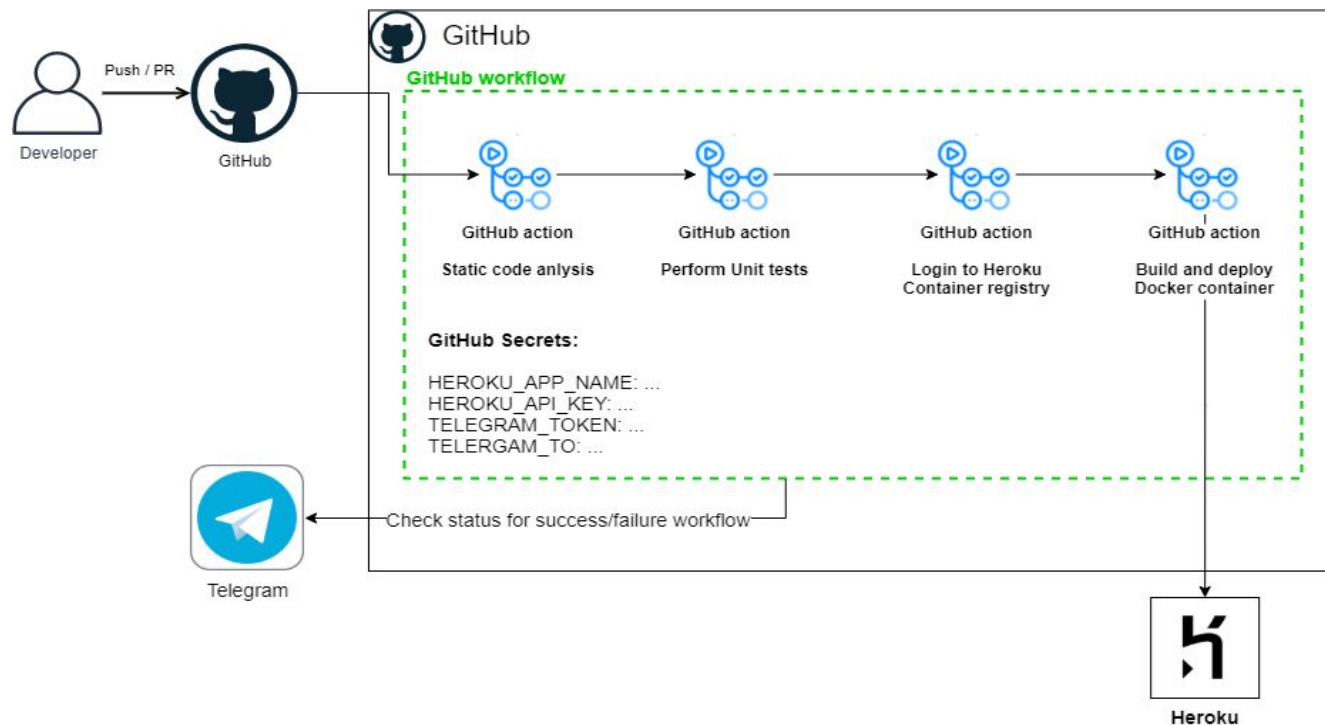
1. PHP

https://github.com/mariohs22/app1_php

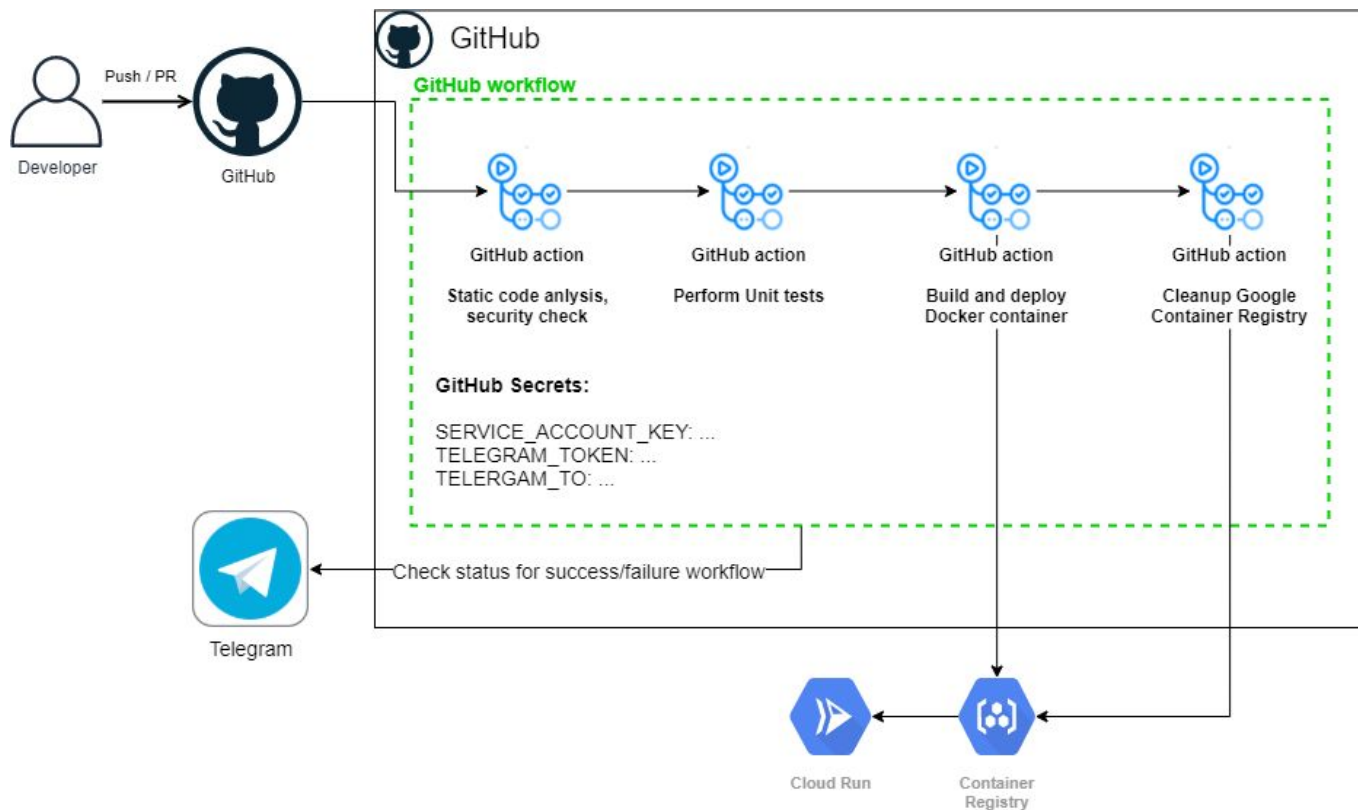
2. Go

https://github.com/mariohs22/app2_go

Приложение 1 (PHP)



Приложение 2 (Go)



Идеальный CI/CD

Чего не хватает:

GitHub-боты для назначения ревьюверов / исполнителей, выставления меток для PR-s (на основе измененных файлов), а по окончании ревью и проверок - для автоматического merge PR и удаления веток;

Создать два workflow:

- default - выполняет линтеры и тесты при поступлении новых изменений;
- release (для ветки release) - то же + сборка и деплой docker контейнера.

Резервное копирование инфраструктуры и приложения;

Настройка уведомлений на ошибки в рабочем приложении.