Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет Информационных технологий и программирования

Работа: Лабораторная работа №6

Выполнили:

Ерёмин Владимир Ильич

Кириленко Илья Андреевич

Проверил: Ельников Сергей Сергеевич

Санкт-Петербург

2022 г.

Лабораторная работа 6: CI/CD.

1. Вступление.

CI/CD — это комбинация непрерывной интеграции *(continuous integration)* и непрерывного развертывания *(continuous delivery или continuous deployment)* программного обеспечения в процессе разработки.

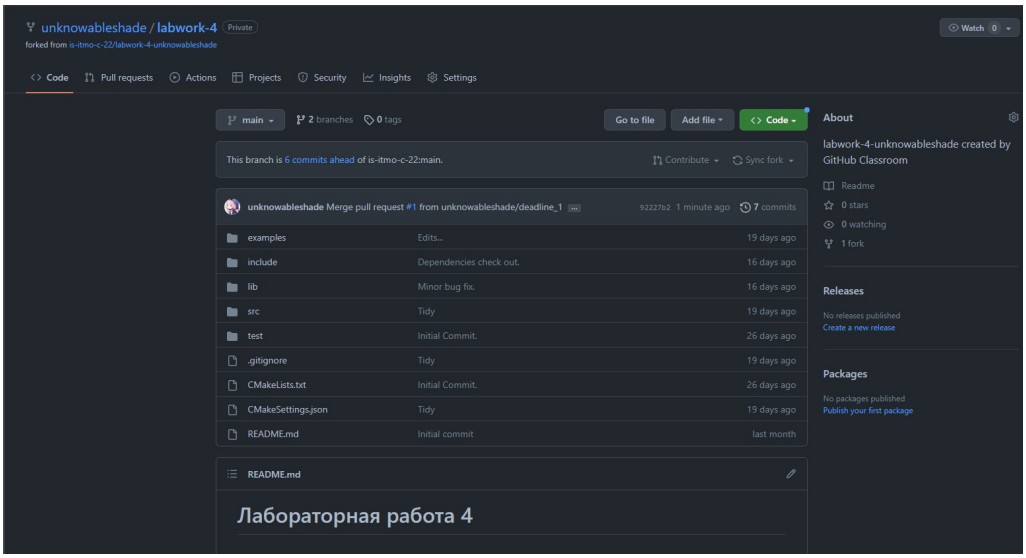
CI/CD объединяет разработку, тестирование и развертывание приложений.

GitHub Actions - это, CI/CD платформа реализующая автоматическую сборку, тестирование и поставку прямо внутри Github репозитория. Может использоваться как для личных, так и для коммерческих проектов. Благодаря такому подходу, можно безопасно делать вклад в проект, а Github actions в случае неудачи на любом этапе, может прервать дальнейшую сборку.

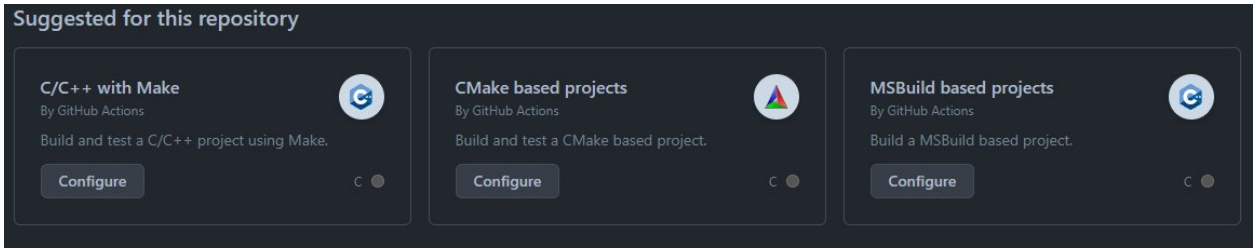
2. Проект и обзор Github Actions.

Имеем проект с основ программирования написанный на C++

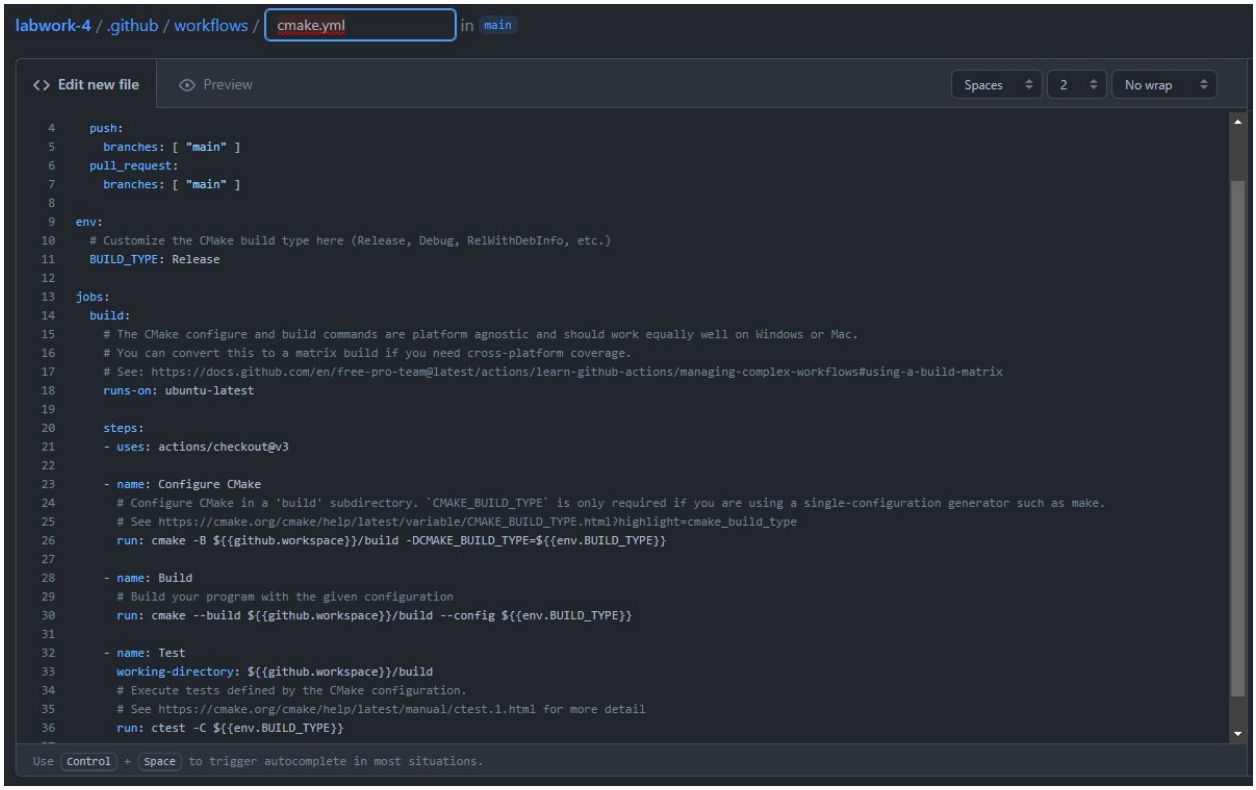
Цель: добавить в репозиторий Github Actions для автоматической сборки, тестирования и генерации исполняемого файла.



Во вкладке “Actions” Github предлагает нам уже готовые шаблоны. Более того, основываясь на том, что содержится в рассматриваемом репозитории он может предложить шаблоны под стать нашим целям.



Я использовал CMake для работы, поэтому будем использовать его и здесь тоже.



Actions построены на YAML файлах, которые описывают окружение, шаги… Чем-то схоже с DockerFile.

Шаблон для CMake говорит делать сборку и тестирование при каждом коммите/pull-request в ветку Main. Для генерации артефактов (в нашем случае исполняемый файл) воспользуемся тем, что предлагает нам сообщество. К уже готовому в репозитории Actions можно добавлять и другие созданные сообществом. (softprops/action-gh-release) будет отвечать за выгрузку артефактов в Release.

Можно отфильтровать запуск Actions по тегам, если добавить данные строки:

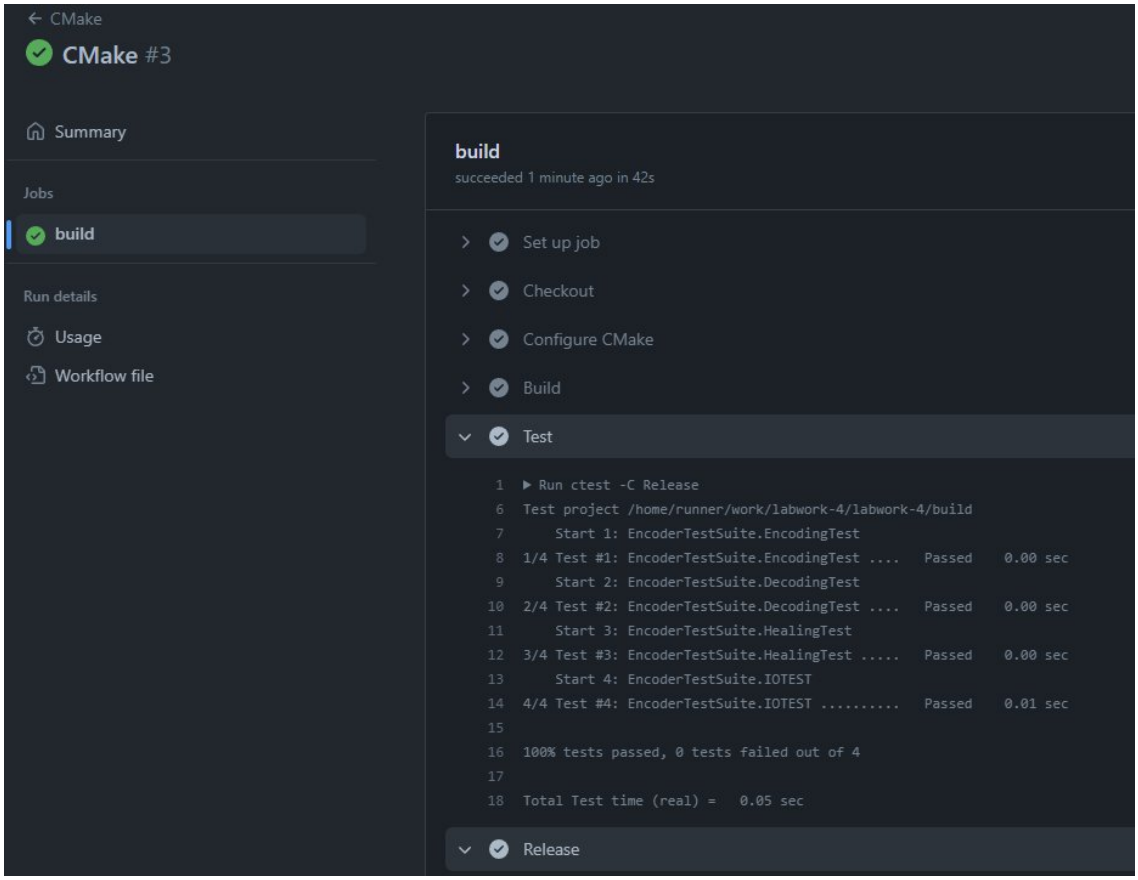
```yaml
on:
  push:
    branches: [ "main" ]
    tags:
      - "v*.*.*"

  pull_request:
    branches: [ "main" ]
    tags:
      - "v*.*.*"
```

Добавим Release в YAML файл:

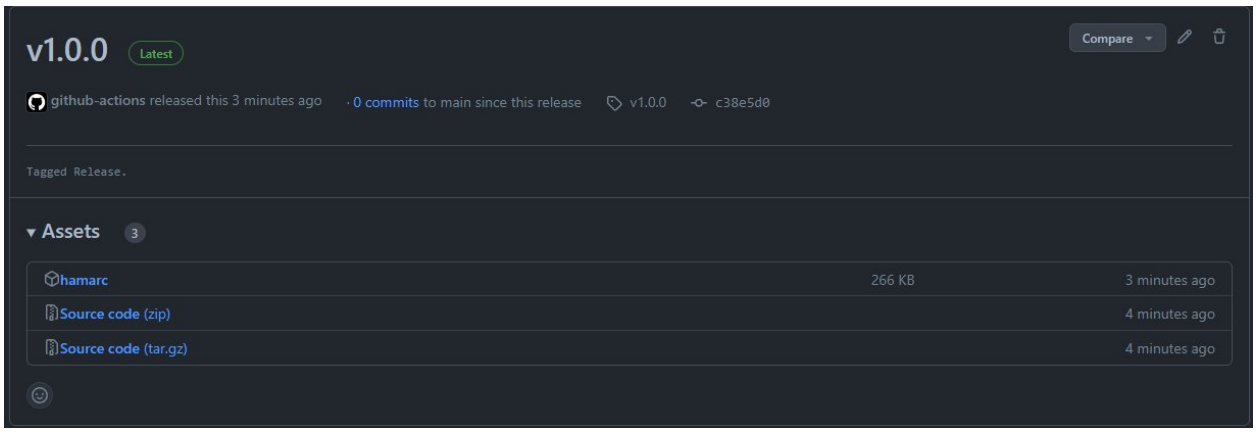
```yaml
- name: Release
  uses: softprops/action-gh-release@v1
  with:
    files: ${{github.workspace}}/build/src/hamarc
```

После добавляем этот файл, по указаниям самого Github, и делаем коммит с тэгом.

Автоматически запускается процесс, описанный в файле:



Ошибок не случилось - наблюдаем наш исполняемый файл во вкладке Release:



Дополнительно заметим, что в данном случае выполнялась сборка только под Linux. Однако с возможностями CMake и Actions можно спокойно собирать под разные ОС и тестировать там ваш проект.