

Северо-Кавказский федеральный университет
Институт математики и информационных технологий

ОТЧЕТ
о выполнении лабораторной работы №1
по дисциплине
«Основы Программной Инженерии»

Выполнил:

Ботвинкин Никита Сергеевич

студент 2 курса, ПИЖ-б-о-20-1 группы
бакалавриата «Программная инженерия»
очной формы обучения

Ставрополь, 2021

СКРИНШОТЫ ОКНА БРАУЗЕРА

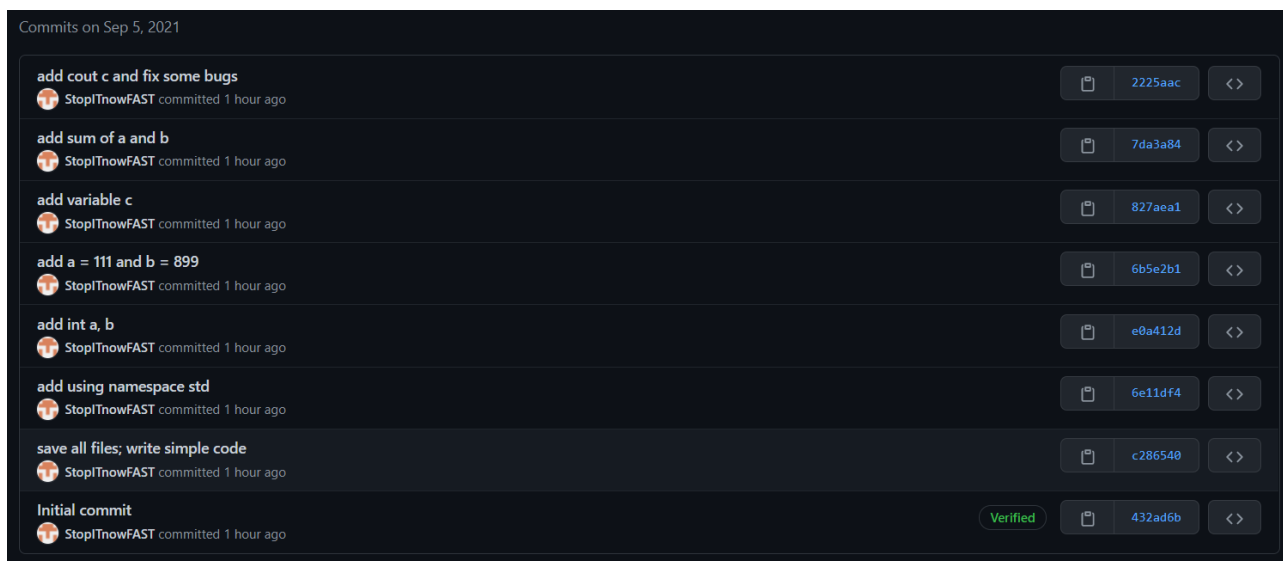


Рисунок 1.1 - коммиты

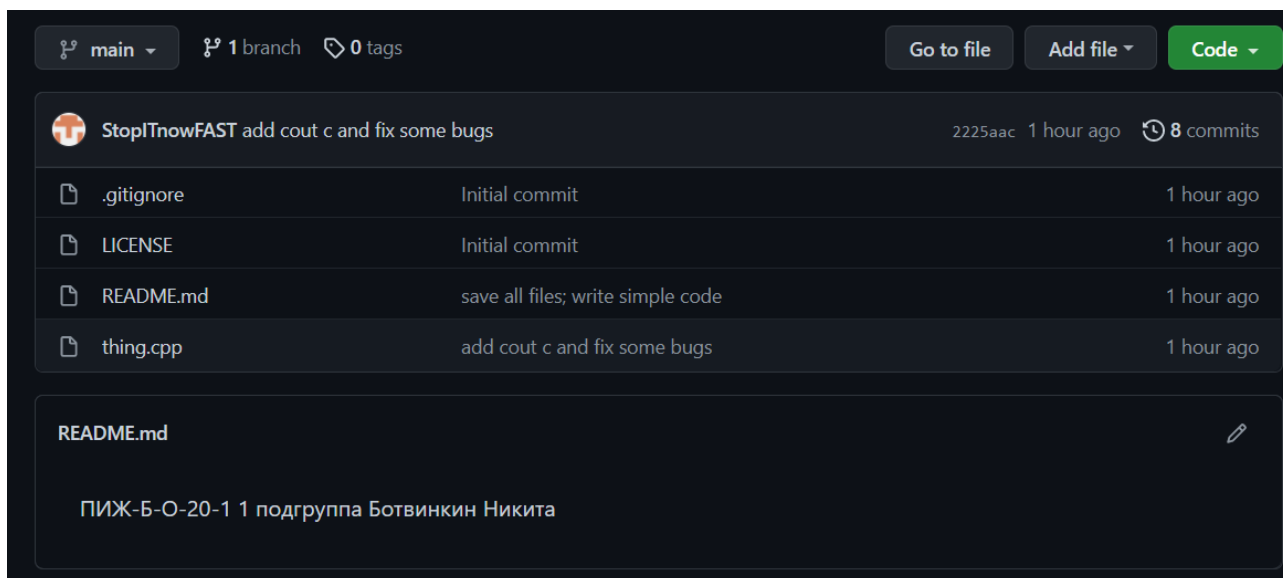


Рисунок 1.2 - репозиторий

СКРИНШОТЫ ТЕРМИНАЛА С КОМАНДАМИ GIT

```
C:\GitStuff\AnotherOne>git add thing.cpp

C:\GitStuff\AnotherOne>git commit -m "add variable c"
[main 827aea1] add variable c
1 file changed, 2 insertions(+)

C:\GitStuff\AnotherOne>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 370 bytes | 370.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/StopITnowFAST/AnotherOne.git
6b5e2b1..827aea1 main -> main

C:\GitStuff\AnotherOne>git add thing.cpp

C:\GitStuff\AnotherOne>git commit -m "add sum of a and b"
[main 7da3a84] add sum of a and b
1 file changed, 2 insertions(+)

C:\GitStuff\AnotherOne>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 301 bytes | 301.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/StopITnowFAST/AnotherOne.git
827aea1..7da3a84 main -> main

C:\GitStuff\AnotherOne>git add thing.cpp

C:\GitStuff\AnotherOne>git commit -m "add cout c and fix some bugs"
[main 2225aac] add cout c and fix some bugs
1 file changed, 3 insertions(+), 1 deletion(-)

C:\GitStuff\AnotherOne>git push
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 322 bytes | 322.00 KiB/s, done.
Total 3 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/StopITnowFAST/AnotherOne.git
7da3a84..2225aac main -> main
```

Рисунок 1.3 — окно cmd

ОТВЕТЫ НА КОНТРОЛЬНЫЕ ВОПРОСЫ

1. СКВ - это система контроля версий. Регистрирует и сохраняет разные версии файла, чтобы при ошибке можно было отследить изменения.
2. локальные - сложность взаимодействия с другими разработчиками, сложность администрирования, риск потери всего
централизованные - единство(сбой сервера - никто не работает), риск потери всего
3. git относится к распределённым системам контроля версий
4. git запоминает снимок каждой версии файла, а при отсутствии изменений, дает ссылку на прошлую версию файла
5. git использует хеш-сумму. хеш-сумма это строка из 40 16-ных символов, которая вычисляется на основе содержимого файла
6. файлы в git могут быть в трех состояниях:
 - а) зафиксированные (committed) - файл сохранен в базе
 - б) изменённые (modified) - файл изменен, но не зафиксирован в базе
 - в) подготовленные (staged) - файл изменен и включен для включения в следующий коммит
7. Такой же как и все другие профили для сайтов, социальных сетей и т.д.
8. Репозитории могут быть приватными и общедоступными
9. работа с gitHub представляет собой:
 - 1) изменение файлом в локальной файле
 - 2) отправка файла на сервер github
 - 3) получение изменений файла от другого пользователя
 - 4) получение файла и изменение его в локальной форме
 - 5) т.д.
10. после установки надо ввести свои данные через команды
`git config --global user.name <YOUR_NAME>`
`git config --global user.email <EMAIL>`
проверить установленную версию можно через команду `git version`
11. для создания репозитория на сайте gitHub надо нажать на "+" >> New repository >> ввести имя, описание, выставить галочки на полях, добавить лицензию >> Create repository

12. Лицензии Git:

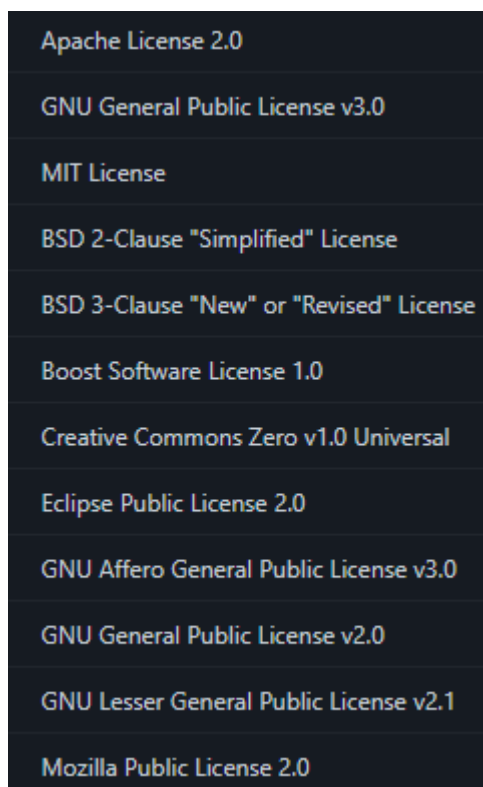


Рисунок 1.4 — виды лицензии git

13. клонирование выполняется командой `git clone`; Команду необходимо выполнить для добавления репозитория на свой ПК.

14. через командную строку в папке с репозиторием надо ввести команду `git status`

15. файл в локальном репозитории изменится в соответствии с изменениями, которые были внесены в файл.

16.

1) прописать на обоих ПК команду `git clone *ссылка*`

2) создать на ПК №1 файл

3) на ПК №1 прописать команду `git add *название файла*`

4) на ПК №1 прописать команду `git push`

5) на ПК №2 для получения изменений файла(и самого файла) пишет `git pull`

17. Gogs - упрощенная версия работы с git

+ встроенная вики система

+ система отслеживания ошибок

+ полностью бесплатный

- упрощенный

18. Sublime Merge:

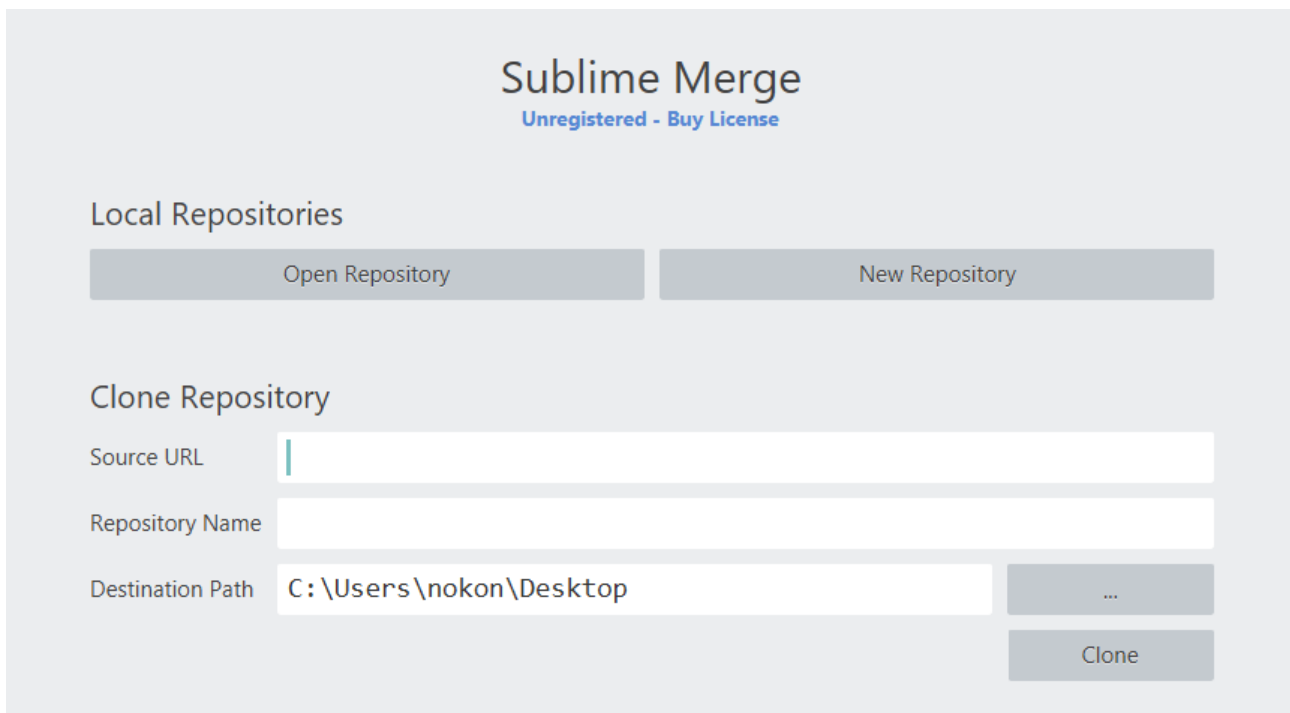


Рисунок 1.5 – добавление репозитория

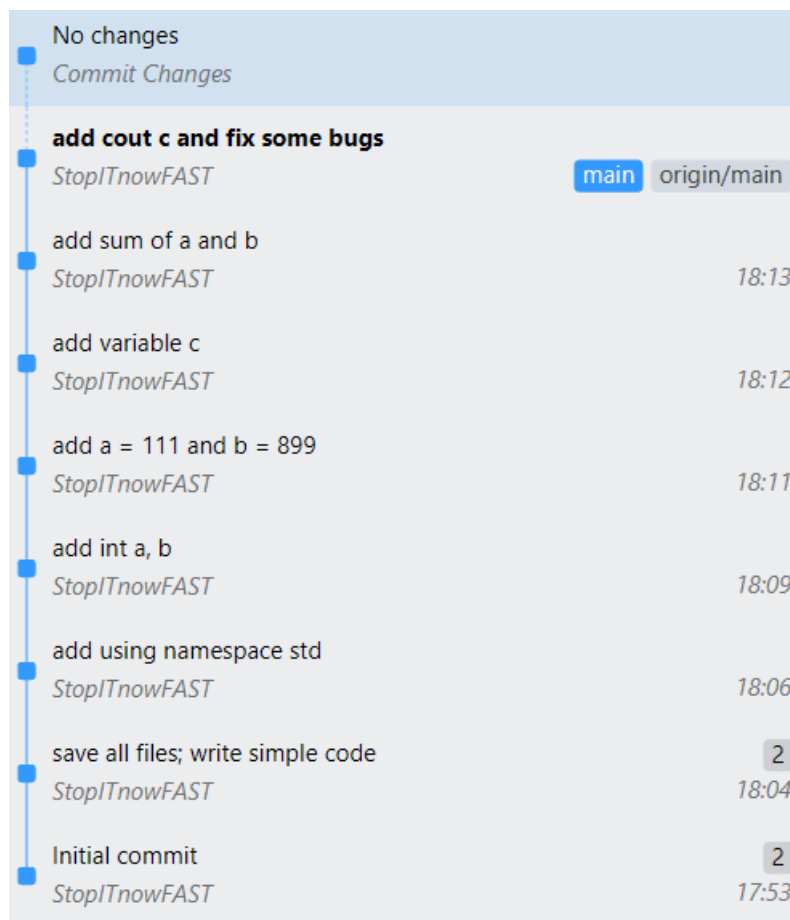


Рисунок 1.6 - коммиты