

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

«Исследование основных возможностей Git и GitHub»

ОТЧЕТ
по лабораторной работе №1
дисциплины
«Основы программной инженерии»

Выполнил:

Луценко Дмитрий Андреевич
2 курс, группа ПИЖ-б-о-21-1,
09.03.04 «Программная инженерия»,
направленность (профиль) «Разработка
и сопровождение программного
обеспечения», очная форма обучения

(подпись)

Проверил:

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2022 г.

Лабораторная работа 1.1 Исследование основных возможностей Git и GitHub.

Цель работы: исследовать основные возможности Git и Git Hub.

Ссылка на репозиторий: <https://github.com/DFooRS/Crimea>

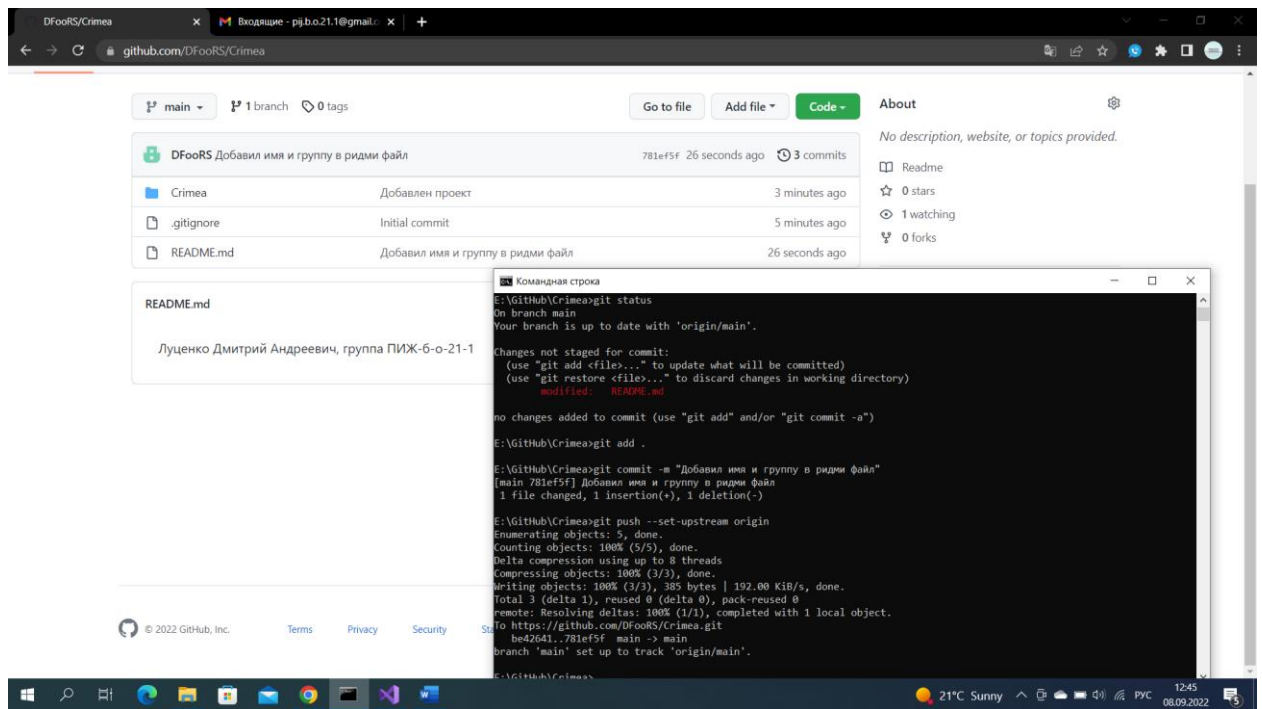


Рисунок 1 – изменения в README файле

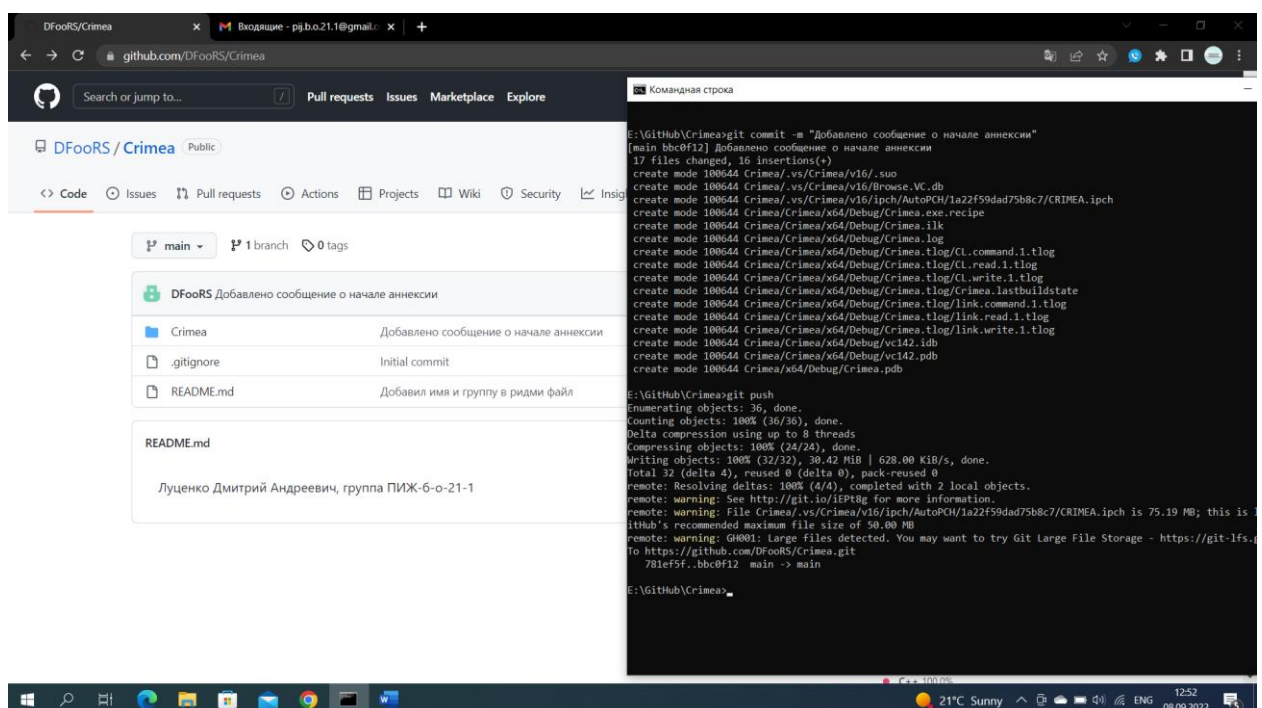


Рисунок 2 – изменения в проекте

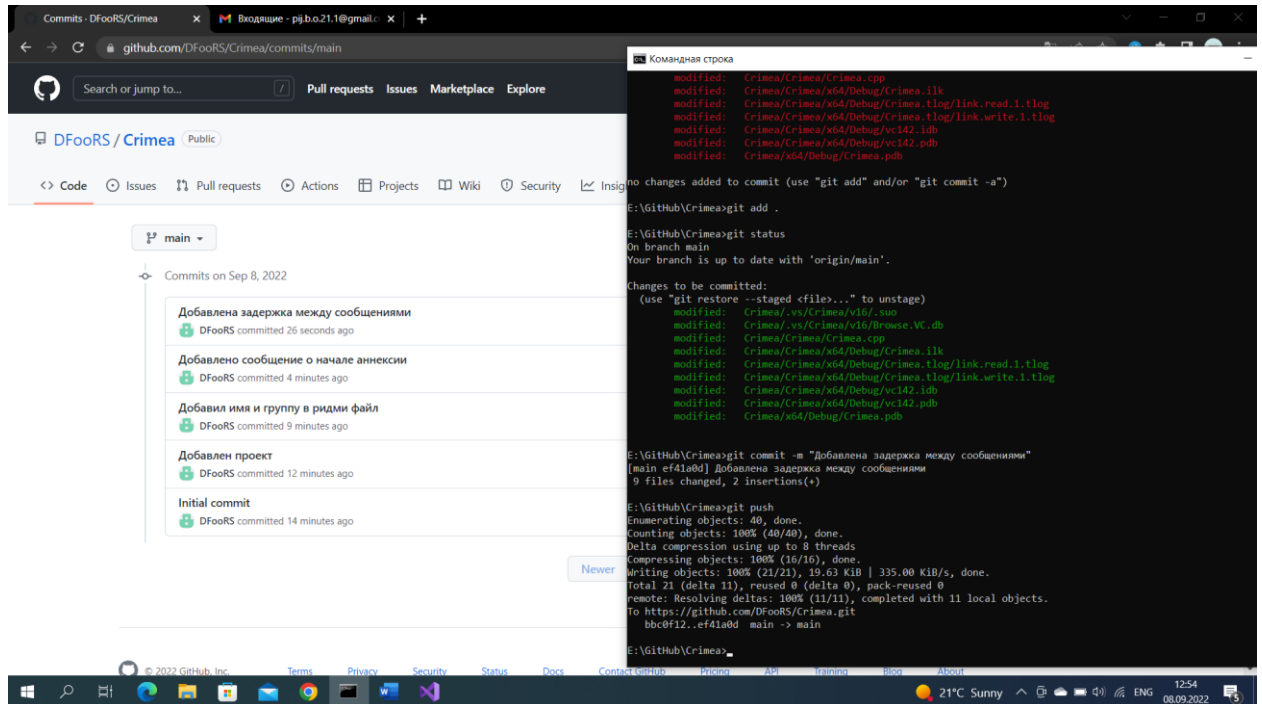


Рисунок 3 - изменения в проекте

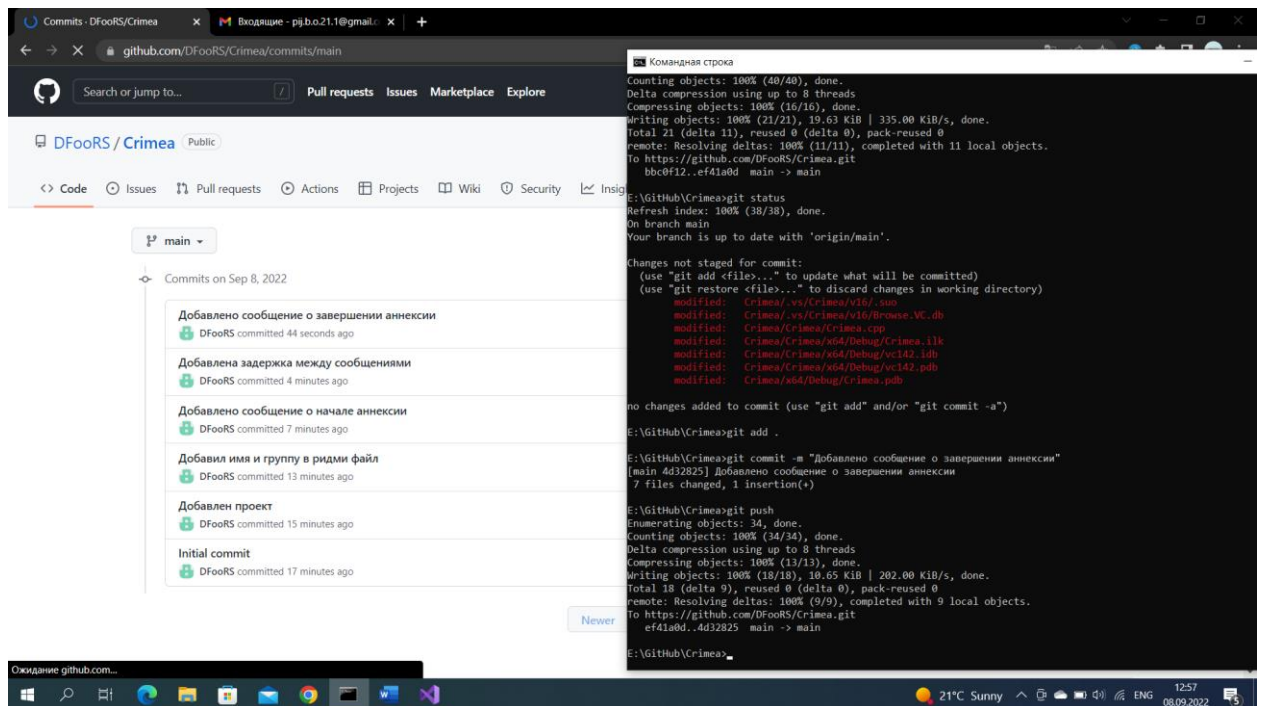


Рисунок 4 - изменения в проекте

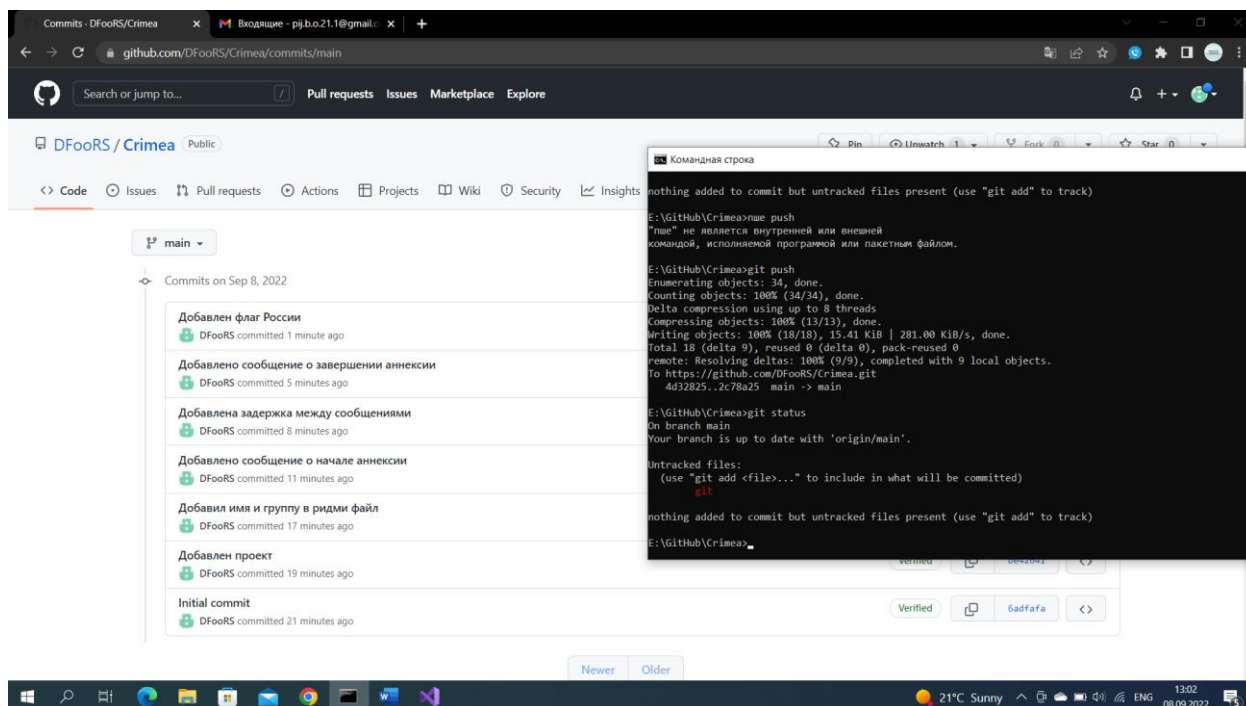


Рисунок 5 - изменения в проекте

Контрольные вопросы:

1. Система контроля версий (СКВ) — это система, регистрирующая изменения в одном или нескольких файлах с тем, чтобы в дальнейшем была возможность вернуться к определённым старым версиям этих файлов.
2. Файлы хранятся в одном месте и в случае неисправности сервера/диска есть риск потерять всё
3. К распределённым системам контроля версий
4. Git переосмысливает практически все аспекты контроля версий. Это делает Git больше похожим на миниатюрную файловую систему с удивительно мощными утилитами, настроенными над ней, нежели просто на СКВ.
5. Git сохраняет все данные не по имени, а хеш-сумме. Благодаря этому без ведома Git не потеряем и не получим поврежденный во время передачи файл

6. У Git есть три основных состояния:

- Зафиксированное (committed). Зафиксированный значит, что файл уже сохранён в вашей локальной базе.
- Измененное (modified). К изменённым относятся файлы, которые поменялись, но ещё не были зафиксированы.
- Подготовленное (staged) Подготовленные файлы — это изменённые файлы, отмеченные для включения в следующий коммит.

7. Профиль – это публичная страница пользователя

8. Репозитории:

- Upstream – оригинальный репозиторий проекта, который мы скопировали
- Origin – копия на GitHub, к которому у нас есть полный доступ

9. Основные этапы работы модели с GitHub:

- Создание репозитория GitHub
- Клонирование репозитория
- Локальное изменение содержимого
- Pull&Push

10. Нужно добавить своё имя и почту, связанные с аккаунтом GitHub

11. Необходимо кликнуть на кнопку «+», затем выбрать New repository, затем ввести название репо, выбрать лицензию, режим доступа, создать readme файл

12. Apache License 2.0, MIT License, GNU General Public License v3.0 итд

13. В командной строке нужно выбрать каталог, куда хотим клонировать репозиторий. После чего прописать `git clone` и вставить ссылку взятую с GitHub

14. В консоли ввести `git status`

15. Красными буквами написано имя файла/зелеными буквами написано имя файла/все изменения внесены

16. `Git clone *ссылка*`, `git add .` (чтобы добавить все файлы), `git commit -m "комментарий"`, `git push --set-upstream origin`, потом можно просто `git push`, `git pull` чтобы внести изменения другого пользователя
17. GitLab, BitBucket
18. Управление через консоль – не самый удобный способ. Есть сервис GitKraken и SourceTree с графическим интерфейсом