МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное

образовательное учреждение высшего образования

«СЕВЕРО-КАВКАЗСКИЙ-ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Отчёт по лабораторной работе №1

по дисциплине «Основы программной инженерии»

Выполнил студент

ПИЖ-б-о-21-1

Сотников Андрей Александрович

Работа защищена «  » 2022 г.

Проверил:

Воронкин Р. А.\_\_\_\_\_\_\_

Ставрополь 2022

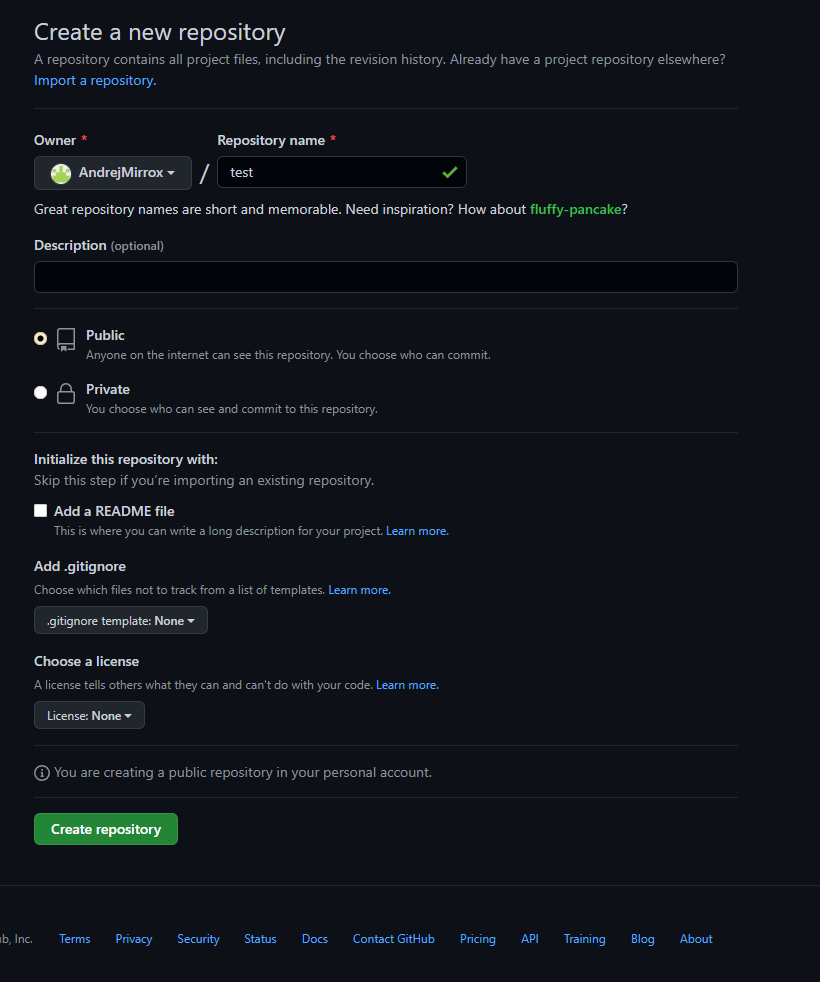
Отсутствие Git на ПК



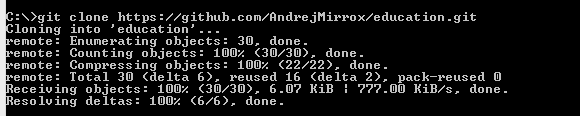
Присутствие Git на ПК:



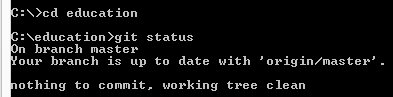
Пример создания репозитория на сайте GitHub



Клонирование репозитория GitHub на локальную маштну:



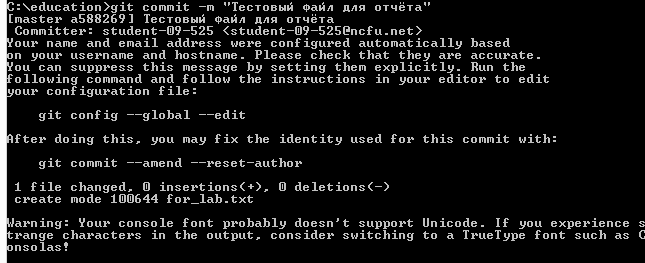
Проварка статуса Git репозитория на локальной машине:



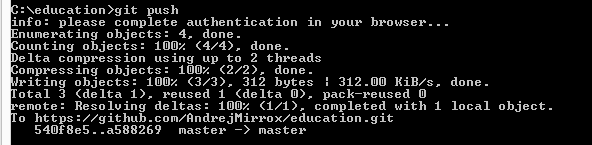
Внесение изменений командой add



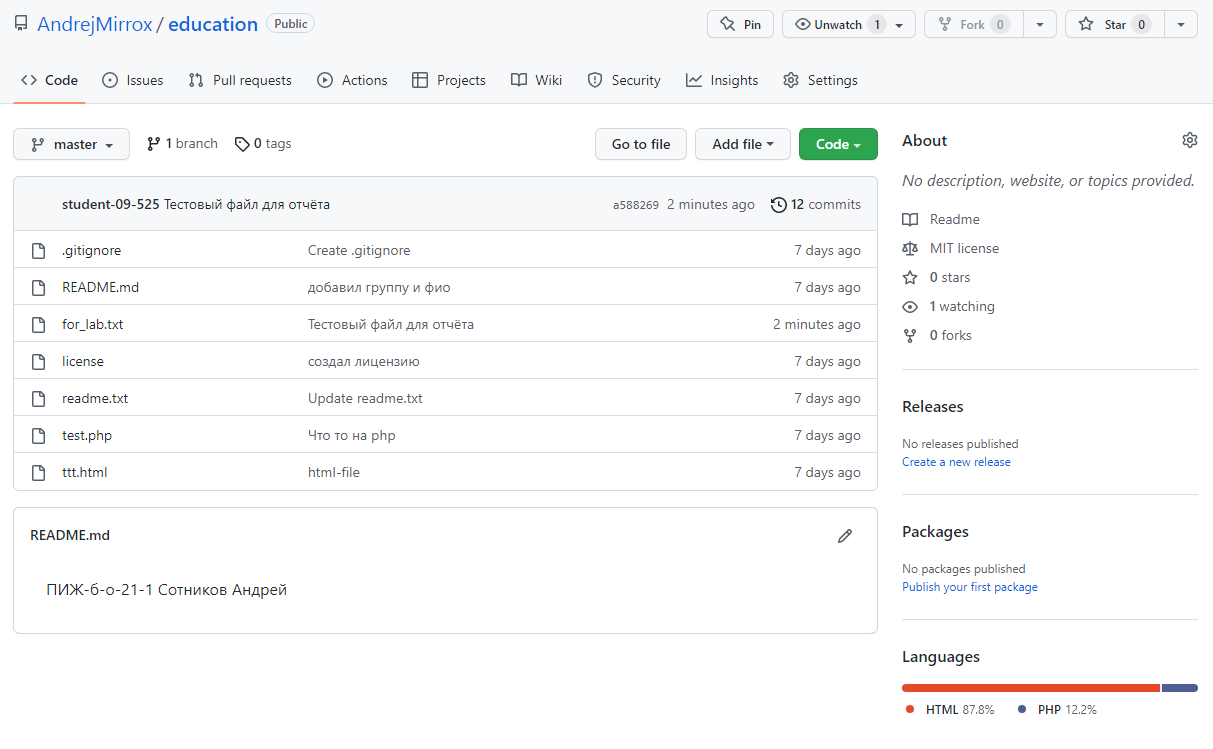
Добавление коммита:



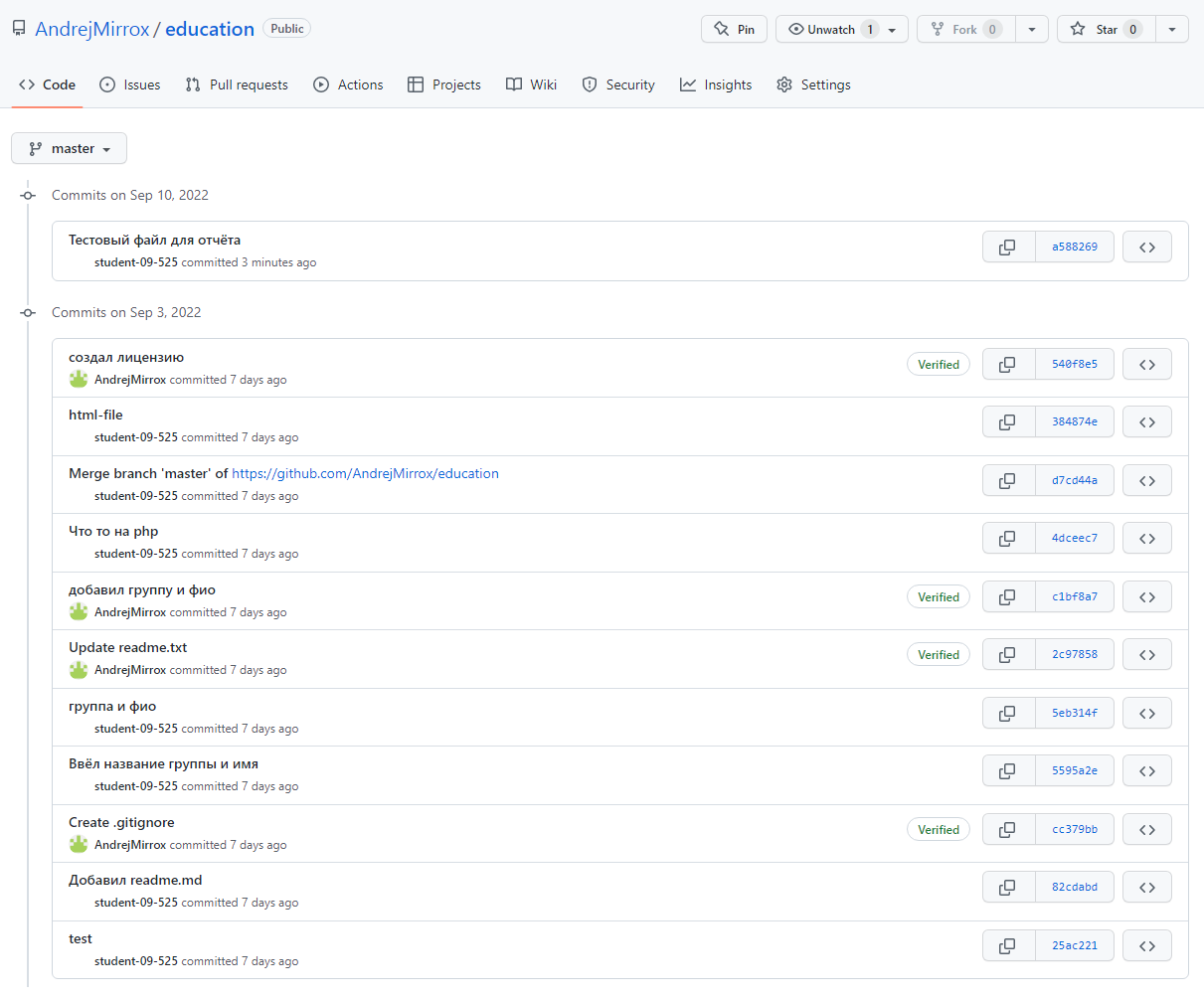
Загрузка изменённых файлов на GitHub:



Скриншот репозитория в GitHub после выполнения работы:



Все коммиты которые были загружены на GitHub:



Ответы на контрольные вопросы:

1. Система контроля версий — это система, записывающая изменения в файл или набор файлов в течение времени и позволяющая вернуться позже к определённой версии файла.

2. Недостатки локальной системы: возможность потери данных вследствие возникновения физических поломок оборудования. Отсутствие возможности совместной разработки.

Недостатки централизованной системы: отсутствие доступа к данным при сбое работы сервера. Снижение скорости за счёт сетевых задержек.

3. Распределённая система контроля версий

4. Подход Git к хранению данных больше похож на набор снимков миниатюрной файловой системы. Каждый раз, когда вы делаете коммит, то есть сохраняете состояние своего проекта в Git, система запоминает, как выглядит каждый файл в этот момент, и сохраняет ссылку на этот снимок. Для увеличения эффективности, если файлы не были изменены, Git не запоминает эти файлы вновь, а только создаёт ссылку на предыдущую версию идентичного файла, который уже сохранён.

6. Файл может находится в двух состояниях, отслеживаемом и не отслеживаемом. В первом случае это те файлы, которые были в последнем снимке состояния проекта, а во втором файлы, которые не входили в последний снимок или не были подготовлены к коммиту.

7. Персональная страница пользователя с настройками и статистикой

8. Публичные и приватные

9. Стандартный подход к работе с проектом состоит в том, чтобы иметь локальную копию репозитория и фиксировать изменения в этой копии, а не в удаленном репозитории, размещенном на GitHub. Этот локальный репозиторий имеет полную историю версий проекта, которая может быть полезна при разработке без подключения к интернету. После того, как мы что-то изменили в локальном, мы можем отправить свои изменения в удаленный репозиторий, чтобы сделать их видимыми для других разработчиков.

10. При помощи команды «git version» можно убедиться в том, что Git был установлен. Далее связываем нужно установить связь с профилем в GitHub для этого используем команды «git config --global user.name <YOUR\_NAME>» и «git config --global user.email <EMAIL>»

11. Для создания репозитория на GitHub нужно нажать на кнопку «New repository» после чего задать его имя, выбрать будет ли репозиторий приватным или публичным и при необходимости установить флажки в пунктах создания «.gitignore» и «LICENSE»

12. MIT, Академическая бесплатная лицензия, Apache license 2.0, Artistic license 2.0 и множество других лицензий

13. Создание локального хранилища проекта происходит при помощи команды «git clone <Ссылка на репозиторий GitHub>»

14. При помощи команды «git status»

15. Добавится/изменится файл в локальном репозитории.

16. Оба компьютера должны создать локальную версию репозитория про помощи команды «git clone» после чего при изменении или добавлении файлов с одного компьютера следует обновить отслеживаемые файлы путем использования команды «git add .», добавлением коммита командой «git commit» и выгрузкой во внешний репозиторий командой «git push», на другом компьютере необходимо использовать команду «git pull» для обновления файлов в локальном репозитории.

17. GitLab и BitBucket

18. GitKraken и SourceTree