#### Министерство образования и науки Российской Федерации

#### Федеральное государственное бюджетное образовательное учреждение

#### Высшего образования

#### «Владимирский государственный университет

#### имени Александра Григорьевича и Николая Григорьевича Столетовых»

**(ВлГУ)**

**Кафедра информационных систем и программной инженерии**

Лабораторная работа № 5

по дисциплине «Технологии программирования»

Тема: «Ведение программных проектов

с использованием систем контроля версий»

Выполнил:

ст. гр. ИСТ-120

О.Э.Титов

Принял:

пр. каф. ИСПИ

В.В. Данилов

Владимир. 2022

1. Цель работы

Научиться пользоваться системой контроля версий при работе с программным проектом и выполнять базовые операции в ней на примере GIT.

1. Выполнение работы

Задание 1

Создайте локальный репозиторий своего проекта с несколькими коммитами и ветками. Залейте проект на удалённый репозиторий.

**Шаг 1.** Создадим локальный репозиторий своего проекта.

Для этого выполним следующие команды в командной строке:

* cd Desktop – переходим на рабочий стол;
* md Lab5 – создаём папку для нашего репозитория;
* cd Lab5 – заходим в созданную папку;
* git init – создаём локальный репозиторий;
* git status – получаем информацию о репозитории;

Результат выполнения шага 1 представлен на рисунке 1.

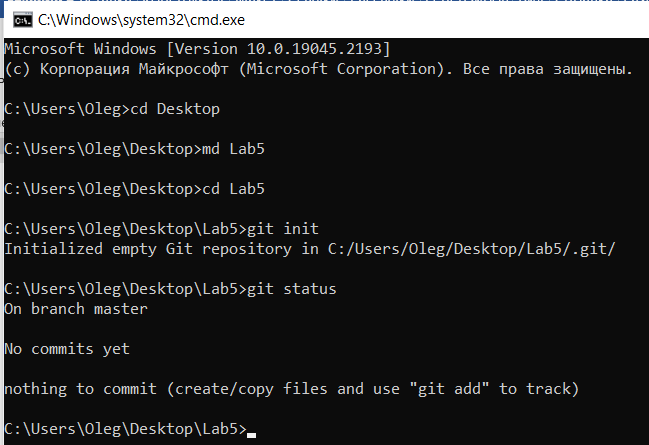


Рисунок 1 - Результат выполнения шага 1.

**Шаг 2.** Добавим несколько файлов в наш репозиторий и сделаем коммиты.

Для этого выполним следующие команды в командной строке:

* git status – получаем информацию о репозитории;
* git add -A – индексируем все файлы в репозитории;
* git commit -m “Some text” – закоммитим изменения в файлах;
* git log – просмотрим историю коммитов;

Результат выполнения шага 2 представлен на рисунке 2.

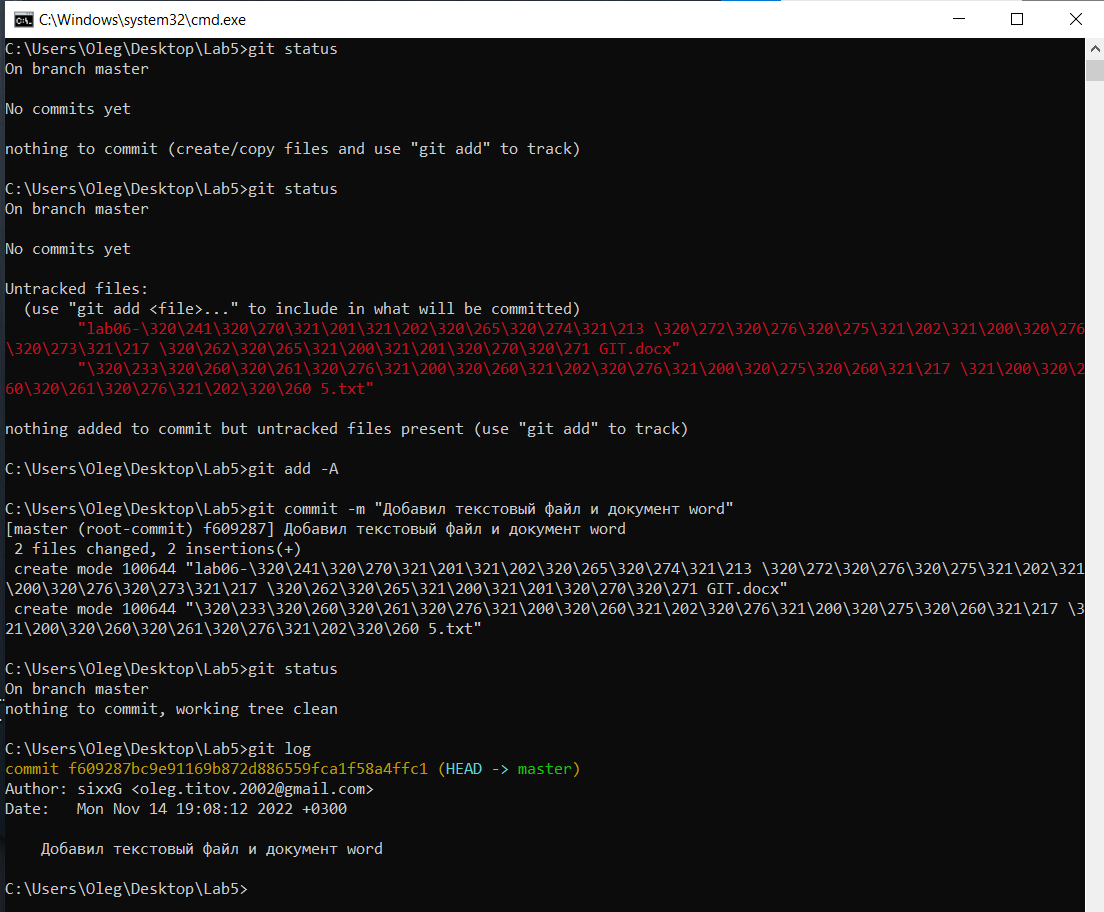


Рисунок 2 - Результат выполнения шага 2.

**Шаг 3.** Создадим новую ветку и сделаем новый коммит в ней.

Для этого выполним следующие команды в командной строке:

* git branch NewBranch – создаём новую ветку;
* git branch -a – просмотр всех веток в репозитории;
* git checkout NewBranch – выберем новую ветку для внесения коммитов;
* Повторим команды из шага 2;

Результат выполнения шага 3 представлен на рисунке 3.

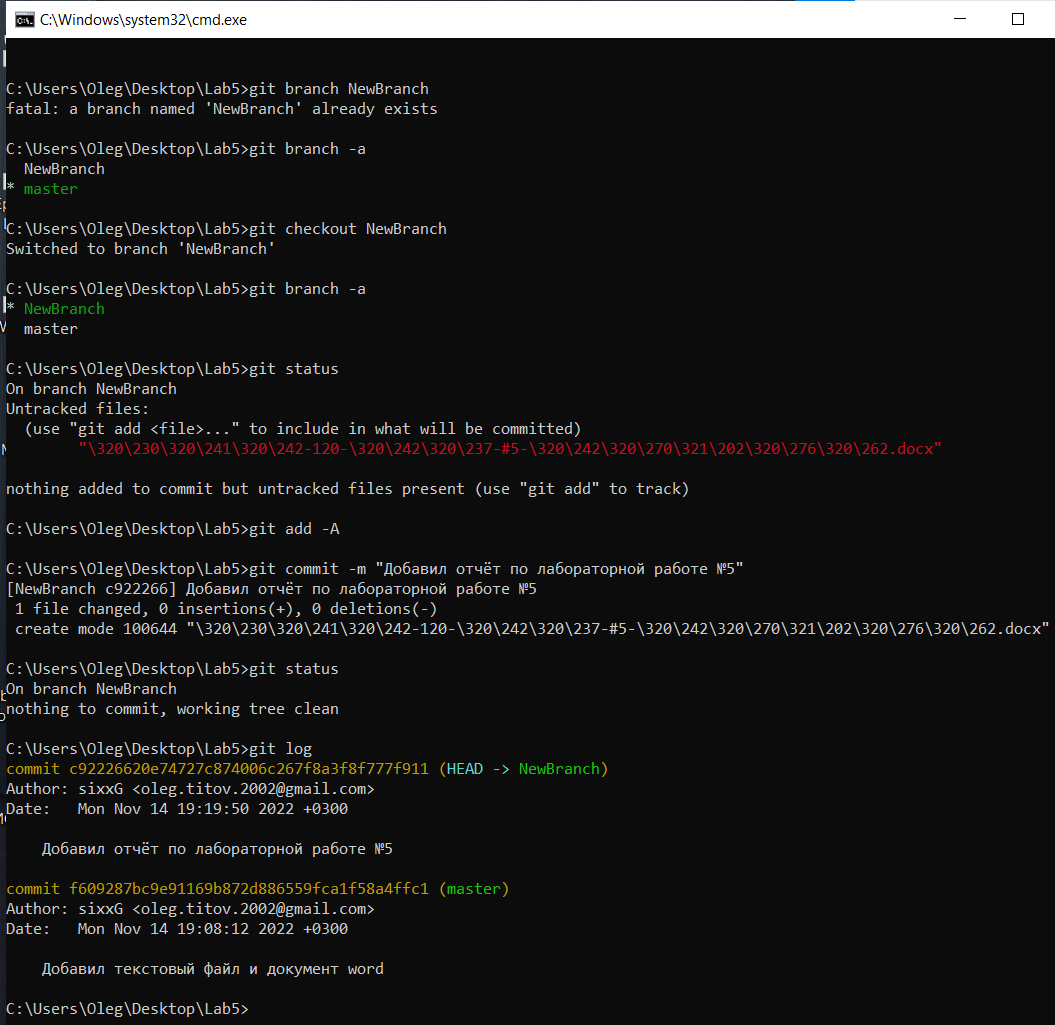


Рисунок 3 - Результат выполнения шага 3.

**Шаг 4.** Объединим все ветки.

Для этого выполним следующие команды в командной строке:

* git merge master – сольём новую ветку с веткой master;
* git checkout master – выберем ветку master в качестве активной;
* git merge NewBranch – сольём ветку master с новой веткой;
* git branch -a – просмотрим активную ветку;
* git status – просмотрит информацию о репозитории;

Результат выполнения шага 4 представлен на рисунке 4.

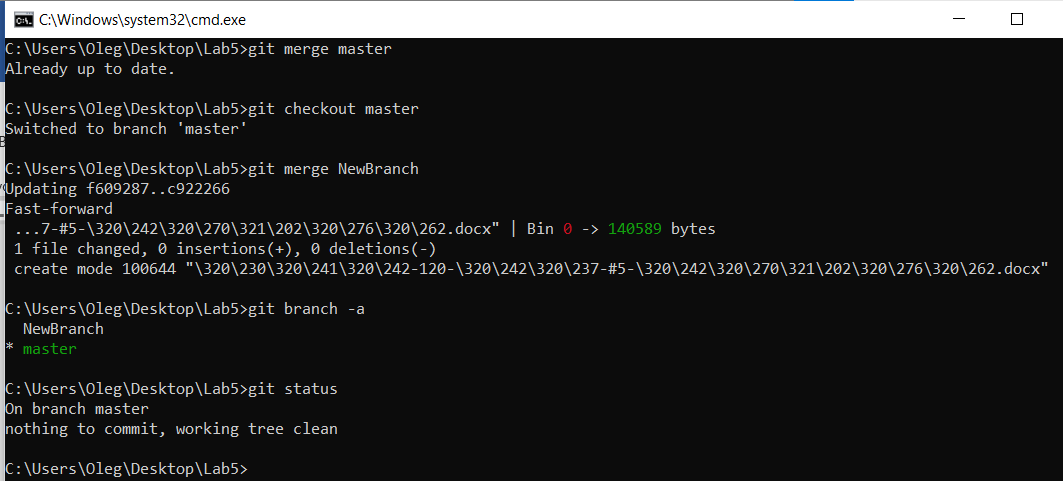


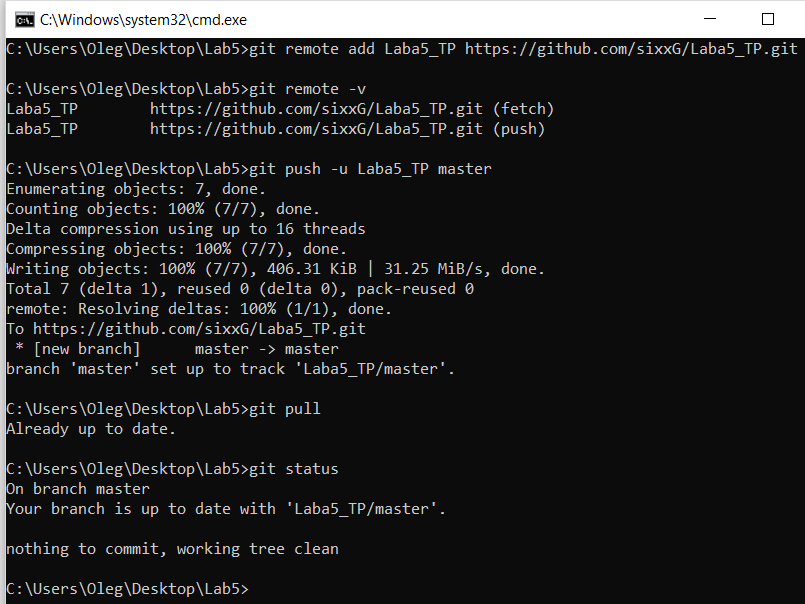
Рисунок 4 - Результат выполнения шага 4.

**Шаг 5.** Зальём созданный проект на удалённый репозиторий.

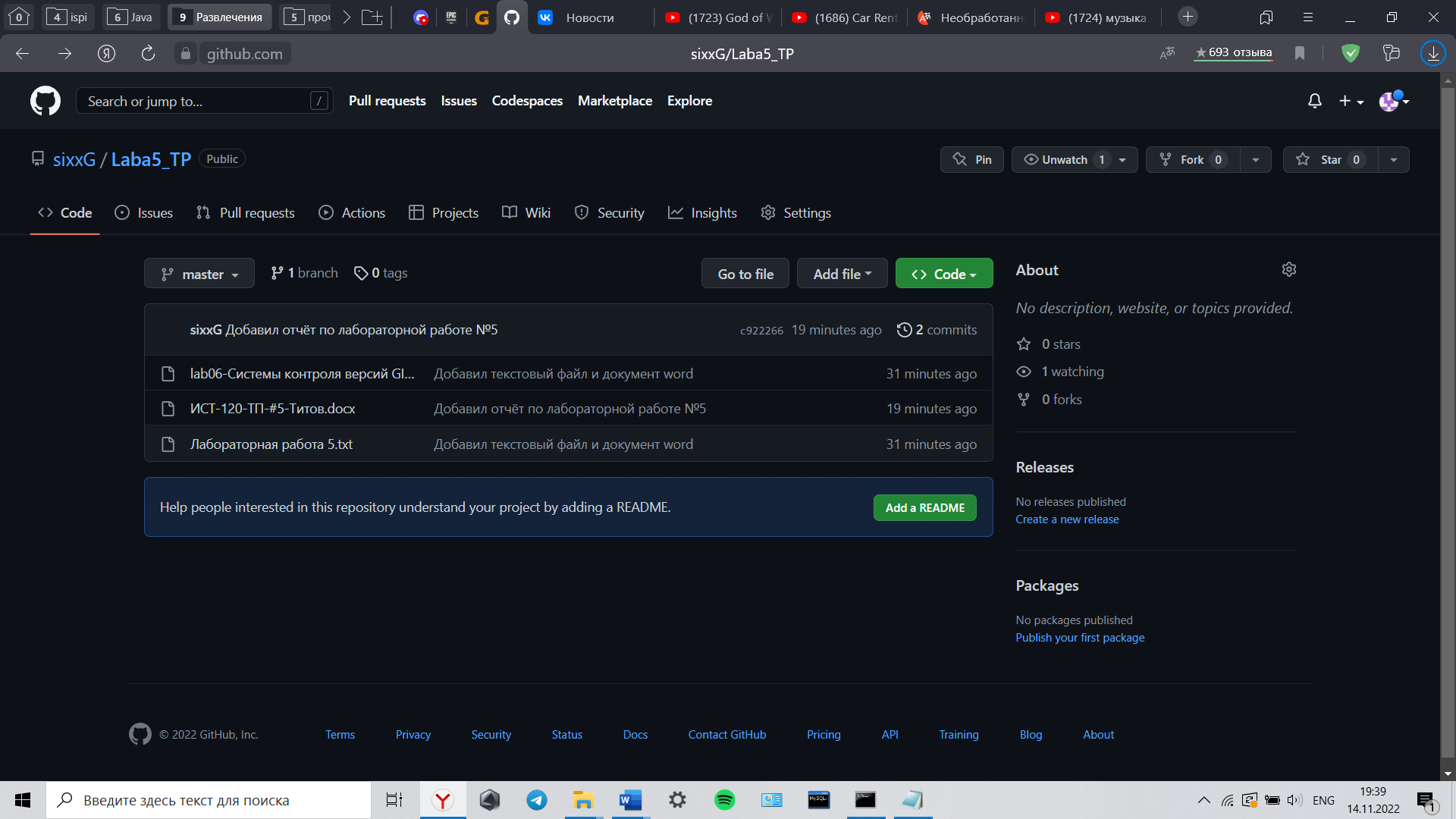
Для этого зарегистрируемся / авторизируемся на GitHub, создадим новый репозиторий и выполним следующие команды в командной строке:

* git remote add Laba5\_TP <https://github.com/sixxG/Laba5_TP.git> – добавим удалённый репозиторий на GitHub;
* git remote -v – просмотр списка удалённых репозиториев;
* git push -u Laba5\_TP master – зальём проект с локального репозитория на удалённый.
* git pull – обновим репозиторий до последней версии;
* git status – просмотрит информацию о репозитории;

Результат выполнения шага 5 представлен на рисунке 5.



а



б

Рисунок 5 - Результат выполнения шага 5. Название – *а – результат выполнения команд в командной строке; б- Созданный удалённый репозиторий.*

Ссылка на созданный репозиторий: <https://github.com/sixxG/Laba5_TP>.

1. Выводы

В процессе выполнения лабораторной работы научились пользоваться системой контроля версий при работе с программным проектом и выполнять базовые операции в ней на примере GIT.