ФЕДЕРАЛЬНОЕ АГЕНСТВО СВЯЗИ

Ордена Трудового Красного Знамени федеральное государственное

бюджетное учреждение высшего образования

«Московский технический университет связи и информатики»

Кафедра информатики

Лабораторная работа №1

на тему: «Основы работы с системой контроля версий»

Выполнила студентка

группы БФИ1902

Кареева В.В.

Проверил:

Москва, 2020 г.

Оглавление

[1. Цель работы 3](#_Toc52488099)

[2. Задание 3](#_Toc52488100)

[3. Ход лабораторной работы 3](#_Toc52488101)

[**Вывод** 9](#_Toc52488102)

[**Список использованных источников** 10](#_Toc52488103)

1. Цель работы

Цель данной лабораторной работы — изучить базовые функции системы контроля версий на примере Git.

2. Задание

Задание на лабораторную работу — определить проект (состоящий из нескольких файлов), который будет добавлен под версионный контроль, и провести ряд операций в СКВ, включающих:

1. Создание репозитория

2. Внесение ряда изменений в файлы и выполнение коммитов

3. Создание веток и выполнение ряда изменений в проекте в созданных ветках

4. Слияние ветки с веткой мастер без конфликтов

5. Слияние ветки с веткой мастер с решением возникающих конфликтов

6. Откат измений до одного из предыдущих коммитов

# 3. Ход лабораторной работы

Для начала проведем настройку Git. Введем имя пользователя и email. Затем создадим пустой репозиторий в папке «OOP» при помощи команды «git init».

На текущий момент наша папка пуста. Добавим в нее текстовый файл «one.txt». Содержимое текстового файла представлено на Рисунке 1.

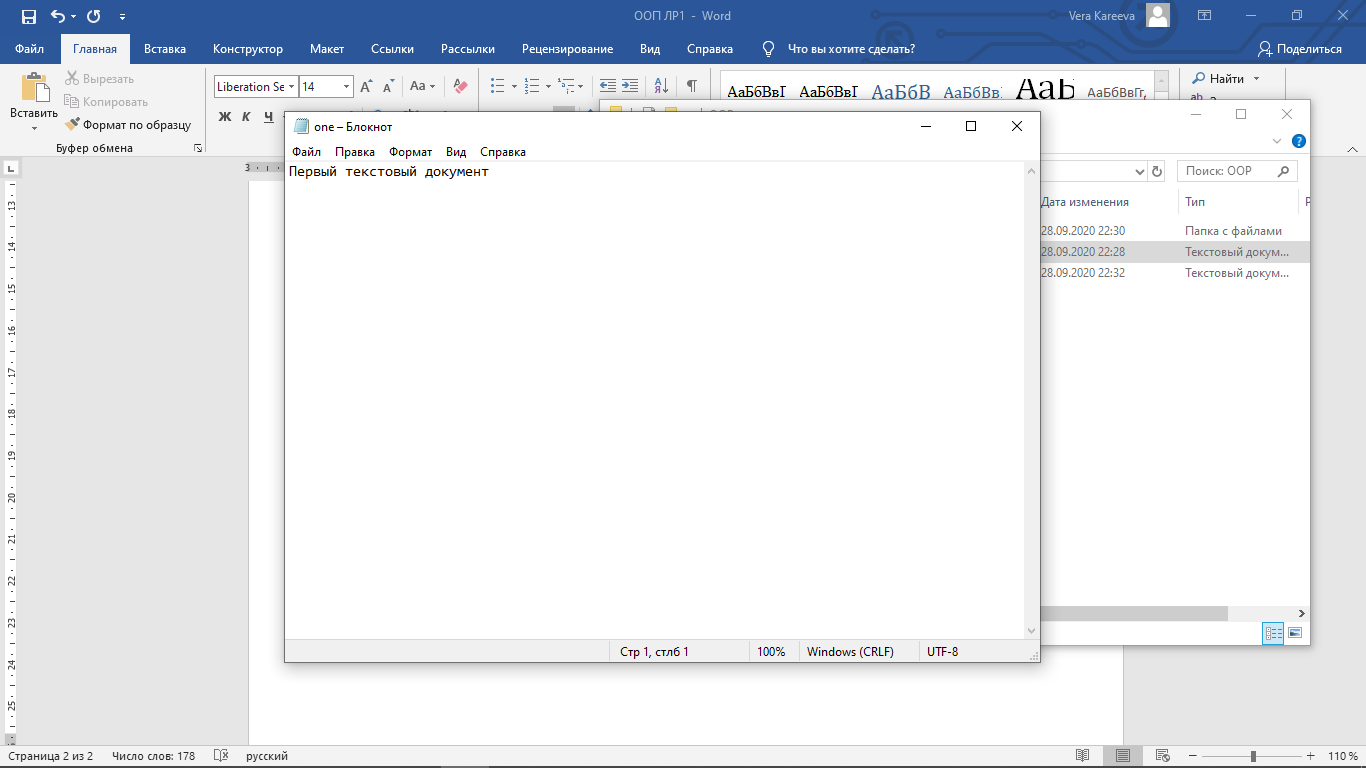


Рисунок 1 - Содержимое первого файла

Проверить состояние проекта можно в любой момент при помощи команды «git status». Чтобы добавить созданный текстовый файл в локальный репозиторий для начала необходимо его добавить в staging area при помощи команды «git add one.txt». После этого производим коммит в локальный репозиторий при помощи команды «git commit -m "Добавлен текстовый файл"». На Рисунке 2 представлена консоль и введенные в нее команды.

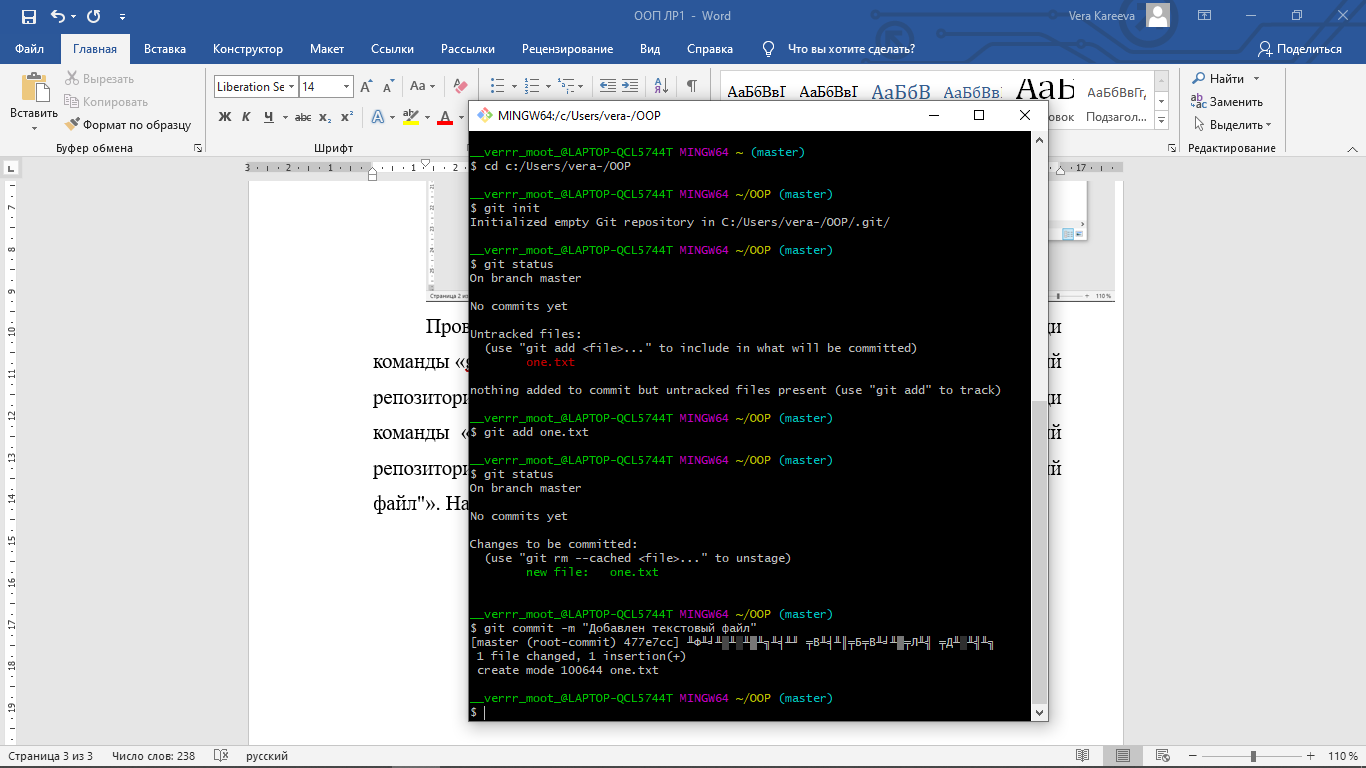


Рисунок 2 - Добавление текстового файла в репозиторий через консоль

Теперь добавим в нашу папку ещё один текстовый файл под названием «two.txt». Его содержимое представлено на Рисунке 3.

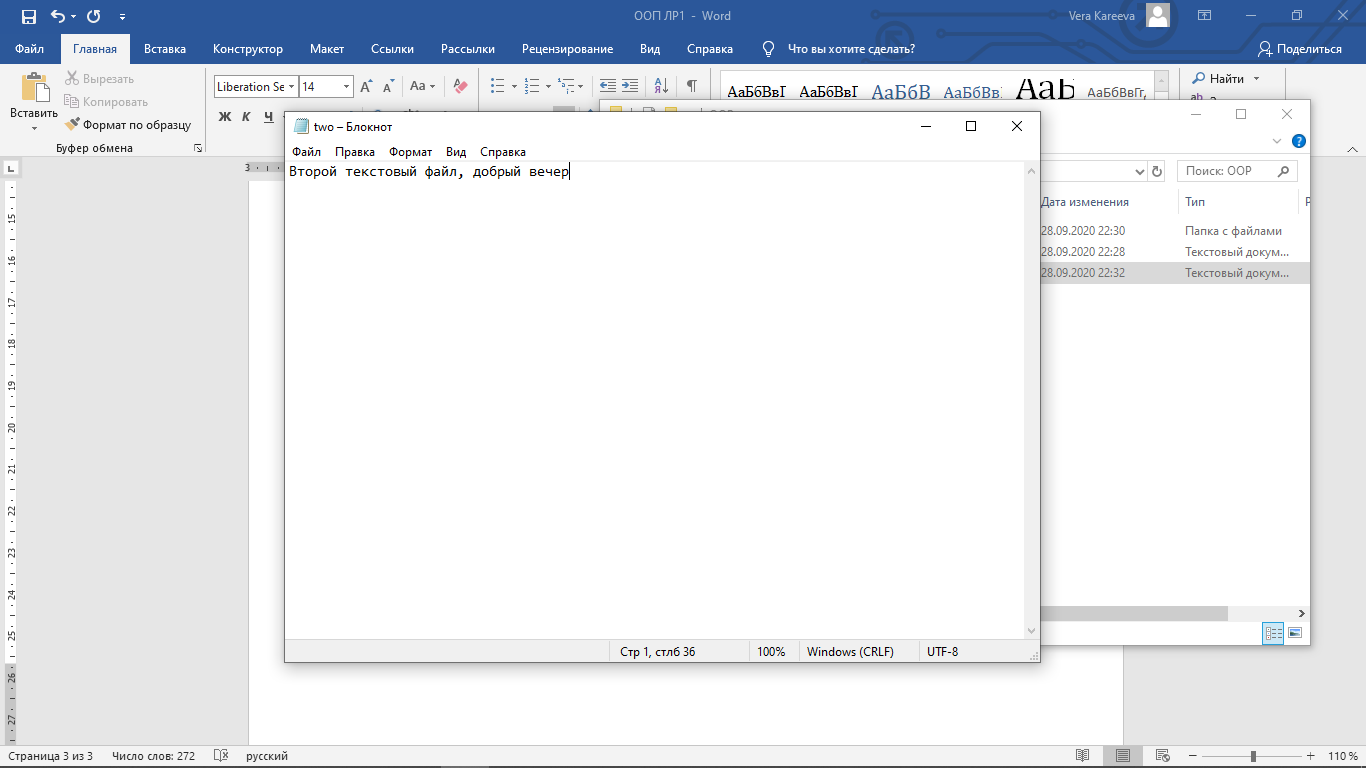


Рисунок 3 - Содержимое второго файла

Вновь используя команды «git add two.txt» и «git commit -m "Добавление нового текстового файла"», добавляем второй текстовый файл в локальный репозиторий. Просмотреть историю коммитов можно при помощи команды «git log». На Рисунке 4 представлено добавление второго текстового файла в репозиторий.

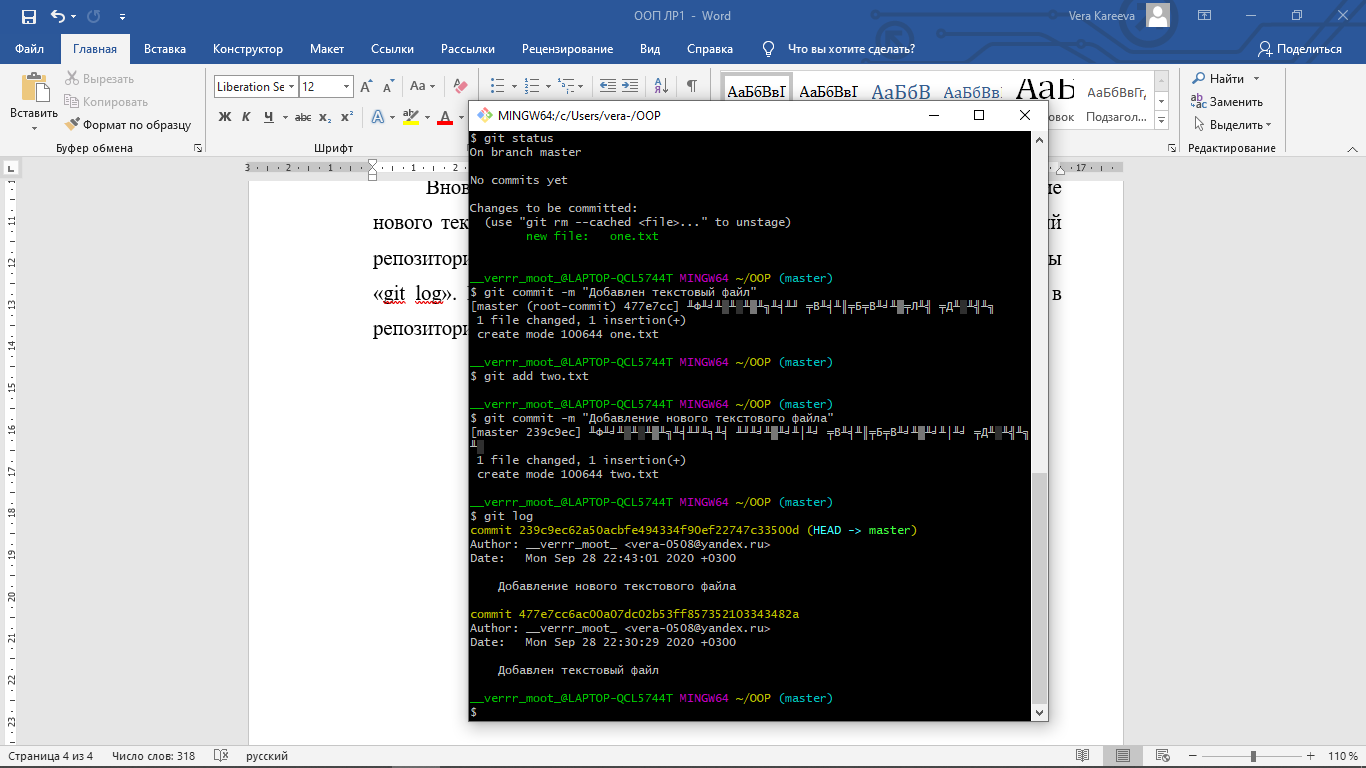


Рисунок 4 - Добавление второго текстового файла в репозиторий через консоль

Далее по заданию нам нужно создать и перейти в новую ветку. Для этого используем команду «git checkout -b Vetka». Находясь в новой ветке, изменим содержимое текстового файла «one.txt» и произведем коммит. На Рисунке 5 представлено создание новой ветки и взаимодействие с ней.

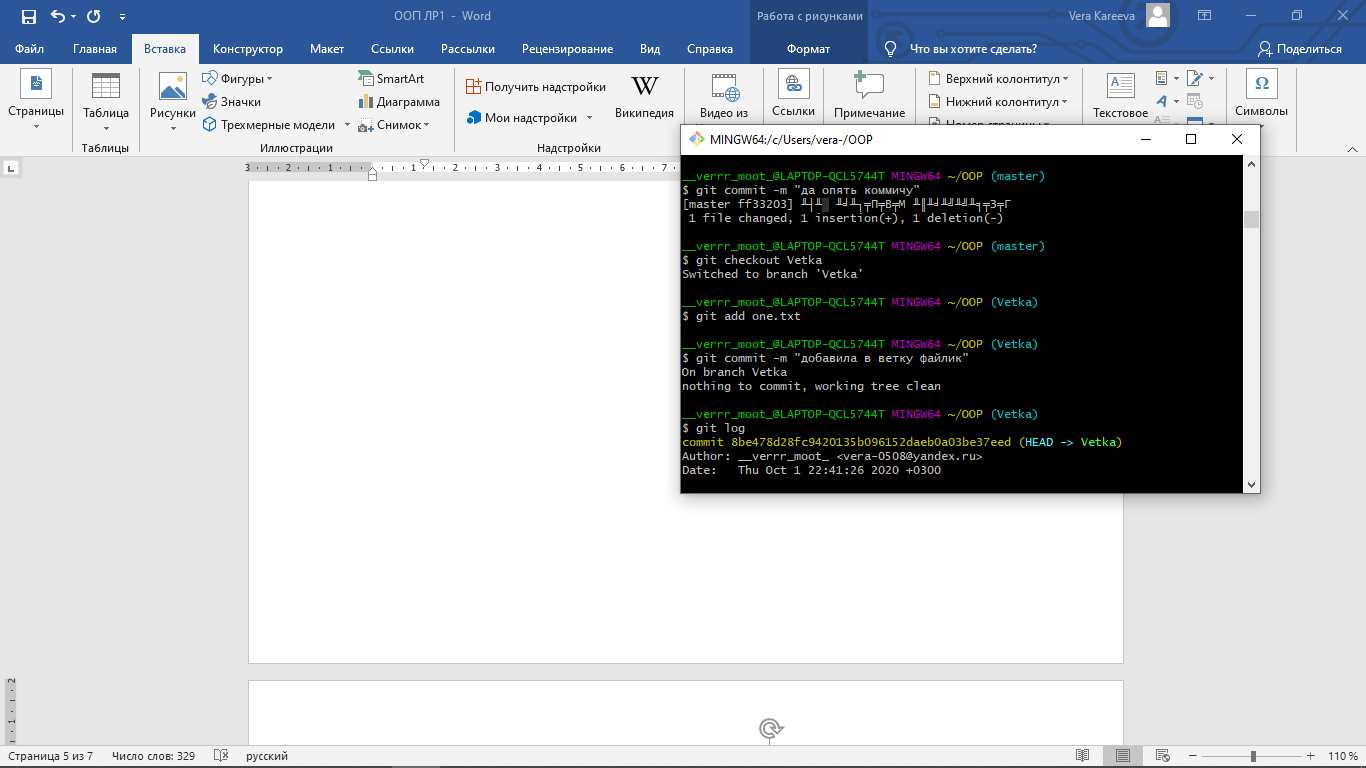


Рисунок 5 - Новая ветка

После этого сделаем слияние ветки «Vetka» с веткой мастер при помощи команды «git merge Vetka». На Рисунке 6 представлено слияние веток.



Рисунок 6 - Слияние веток

Если по какой-то причине содержимое файлов различается между ветками, Git остановит слияние и предупредит об ошибке (Рисунок 7).

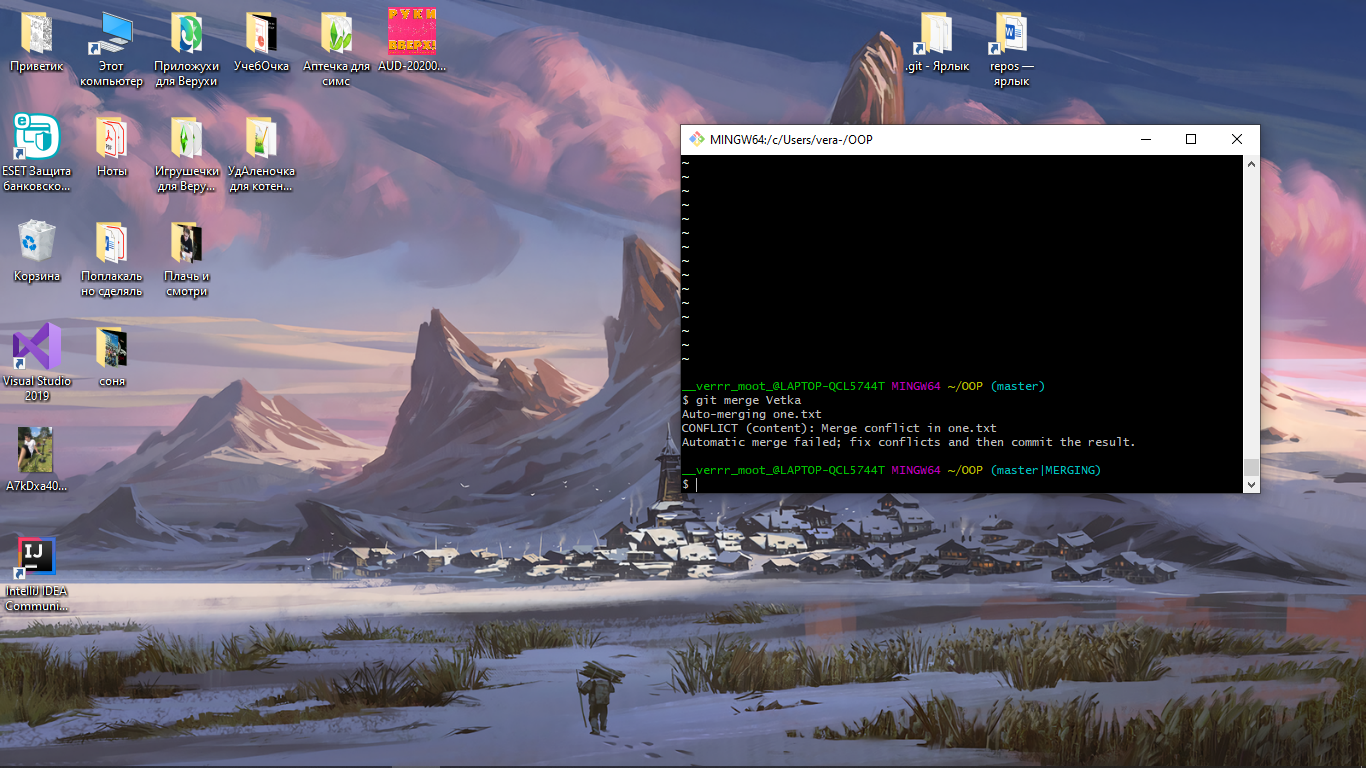


Рисунок 7 - Конфликт при слиянии веток

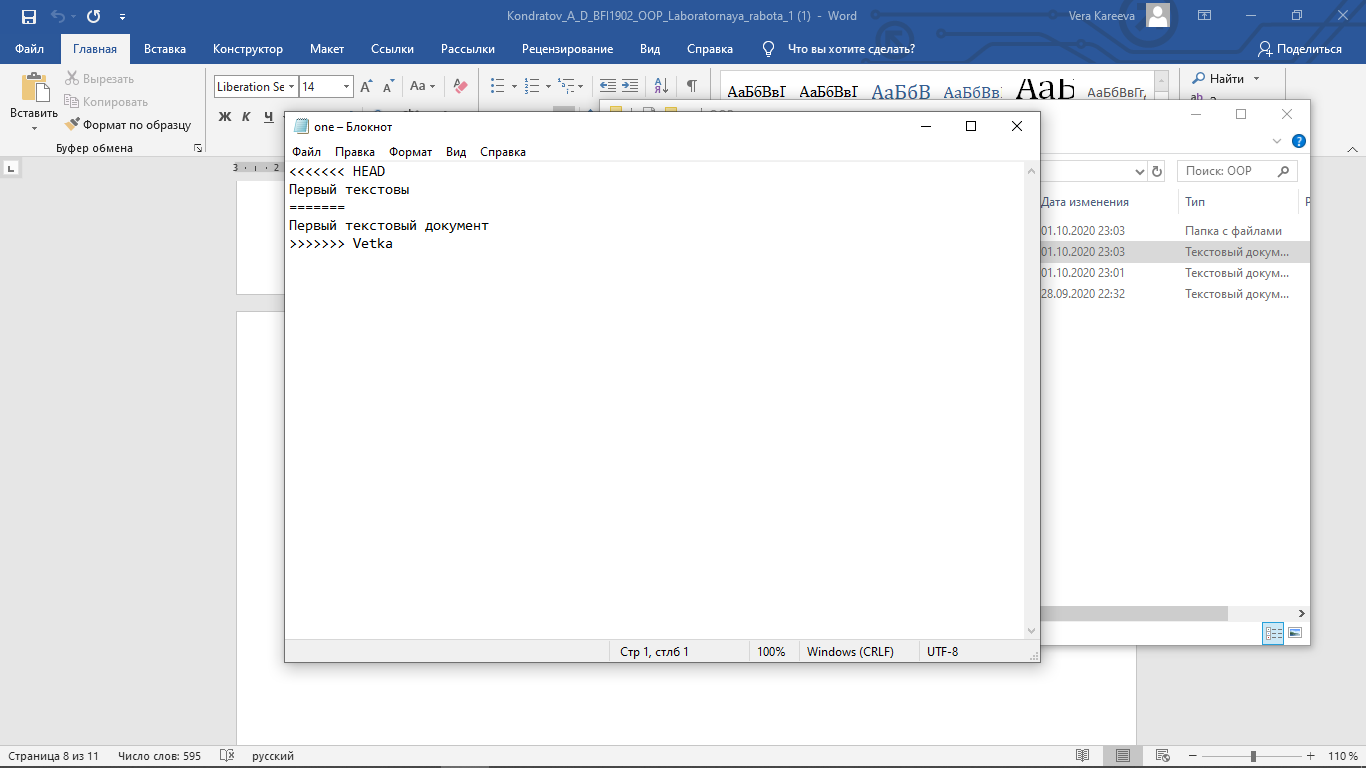
Также изменится и содержимое конфликтных файлов, чтобы указать на эти различия. Вид такого файла представлен на Рисунке 8.

Рисунок 8 - Содержимое конфликтного файла

Для того, чтобы разрешить конфликт, нужно изменить содержимое текстового файла так, чтобы текст между ветками был идентичен. Тогда повторное слияние пройдет без конфликтов.

Предположим, что у нас возникла необходимость сделать откат до одного из предыдущих коммитов. Пускай это будет коммит, когда мы добавили первый текстовый файл. Сделать это можно при помощи команды «git reset —hard». Удостоверится в откате можно при вводе команды «git log». На Рисунке 9 представлен откат до предыдущего коммита.

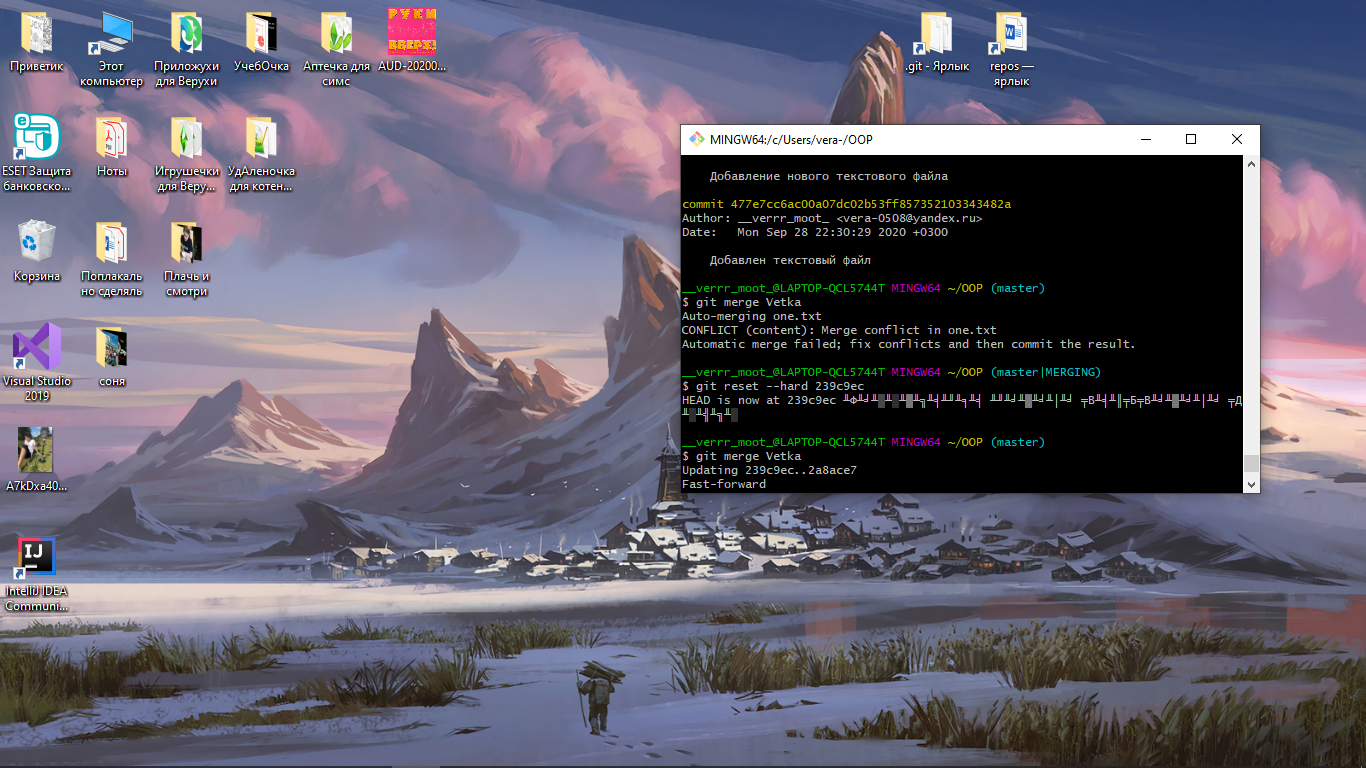


Рисунок 9 - Откат и успешное слияние веток после конфликта

**Вывод**

Таким образом, на примере Git мы научились пользоваться базовыми функциями системы контроля версий.

**Список использованных источников**

1) ГОСТ 7.32-2017 Система стандартов по информации, библиотечному и издательскому делу. Отчёт о научно-исследовательской работе. Структура и правила оформления

2) ГОСТ 7.1-2003 Библиографическая запись. Библиографическое описание. Общие требования и правила составления