**Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации**

Ордена Трудового Красного Знамени федеральное государственное образовательное учреждение высшего профессионального образования

**МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И ИНФОРМАТИКИ**

Кафедра математической кибернетики и информационных технологий

Лабораторная работа №2

по дисциплине «Кроссплатформенные технологии программирования»

Выполнил: студент группы БСТ2001

Багдасарян Артём Сергеевич

Москва 2022

Оглавление

[**1 Цель и задание** 3](#_Toc95221329)

[**2 Ход выполнения работы** 3](#_Toc95221330)

[**2.1 Настройка игнорирования** 3](#_Toc95221331)

[**2.2 Вывод лог коммитов** 4](#_Toc95221332)

[**2.3 Переименовать ветку** 4](#_Toc95221333)

[**2.4 Слияние веток и удаление** 5](#_Toc95221334)

[**2.5 Удалённый репозиторий** 5](#_Toc95221335)

[**3 Выводы** 6](#_Toc95221336)

[**Список использованных источников** 7](#_Toc95221337)

# **1 Цель и задание**

**Цель:** изучение подходов к командной разработке в системе контроля версий git

**Задание:** В созданном в предыдущей работе либо вновь созданном репозитории:

1. Настроить игнорирование git’ом файлов определенного расширения с использованием glob-шаблонов
2. Вывести лог коммитов в измененном пользовательском формате
3. Переименовать ветку
4. Провести в ветке работу и слить с веткой master, удалить слитую ветку
5. Настроить удалённый репозиторий: получить изменения из репозитория и отправить изменения.
6. Организовать совместную работу над проектом в соответствии с выбранной идеологией

# **2 Ход выполнения работы**

## **2.1 Настройка игнорирования**

В папке создаём файл с расширением .gitignore. После открываем его с помощью текстового редактора Блокнот и вносим в него расширения, которые необходимо игнорировать.

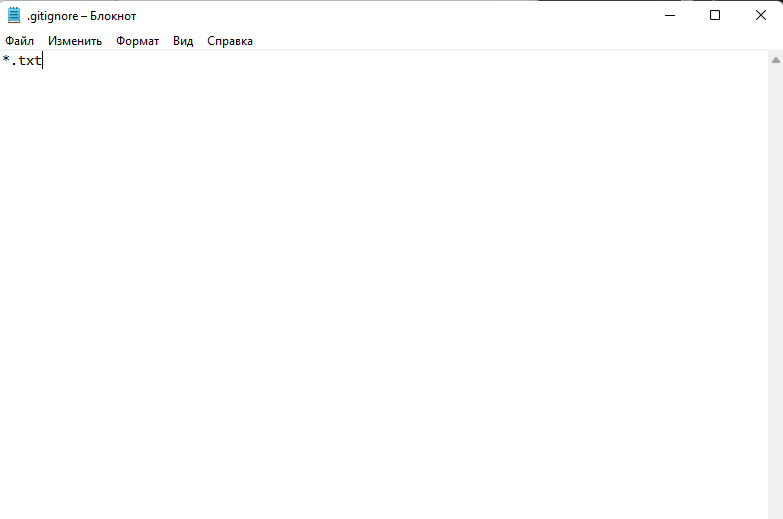


Рисунок 1 – Внесение расширений для игнорирования

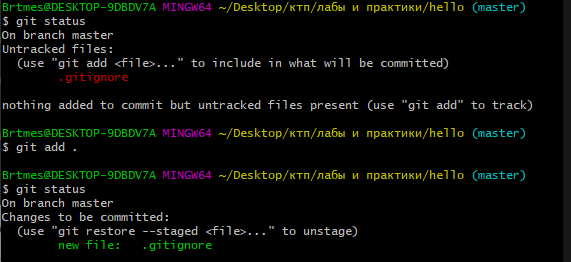


Рисунок 2 – Добавление файла .gitignore в систему контроля Git

После чего буду игнорироваться файлы с расширением .txt

## **2.2 Вывод лог коммитов**

git log – вывод истории коммитов

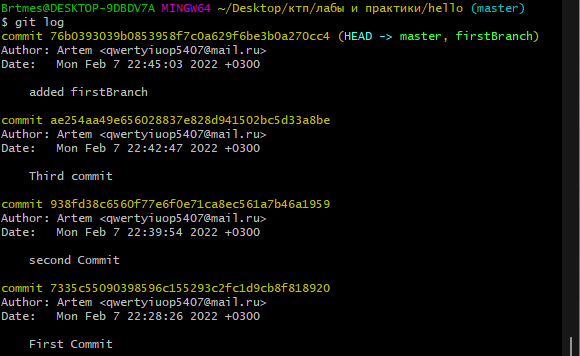


Рисунок 3 – Вывод лог коммитов

## **2.3 Переименовать ветку**

Необходимо перейти на ветку, которую будем переименовывать и с помощью команды, представленной на рисунке ниже, переименовать ветку.

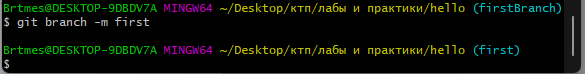


Рисунок 4 – Изменение названия ветки

## **2.4 Слияние веток и удаление**

На рисунке представлены команды для слияния и удаления ветки

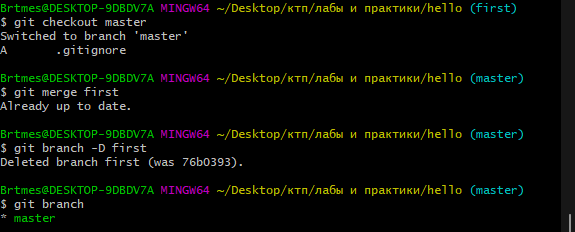


Рисунок 5 – Слияние и удаление ветки

## **2.5 Удалённый репозиторий**

На GitHub создаём репозиторий, после чего необходимо прописать в Git следующие команды



Рисунок 6 – Репозиторий, в котором будут храниться файлы

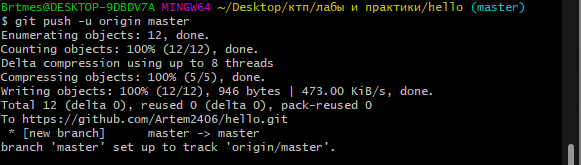


Рисунок 7 – Первоначальная загрузка на удалённый репозиторий

Для получения изменений используем команду git pull, после чего файлы автоматически изменятся

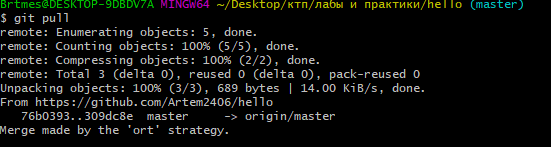


Рисунок 8 – Получение обновлений

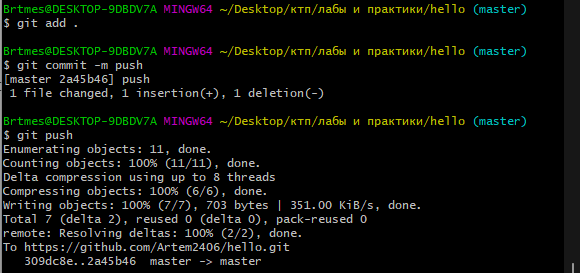


Рисунок 9 – Отправка изменений

# **3 Выводы**

Выполнив данные задачи были изучены подходы к командной разработке в системе контроля версий git

# **Список использованных источников**

1. Камаев В.А., Костерин В.В. Технологии программирования. М.: Высшая школа, 2006.
2. Жоголев Е.А.Технология программирования. – М.: Научный мир, 2004.