Федеральное агенство связи

Ордена Трудового Красного Знамени федеральное государственное

бюджетное учреждение высшего образования

«Московский технический университет связи и информатики»

Кафедра информатики

Лабораторная работа №0 (Часть 2)

по дисциплине: «Технология разработки программного обеспечения»

на тему: «Основы работы с системой контроля версий»

Выполнил студент

группы БФИ1902

Рахимов Е.К.

Проверила:

Москва, 2020 г.

**Оглавление**

**1. Цель работы 3**

**2. Задание 3**

**3. Ход лабораторной работы 3**

**Вывод 9**

**Список использованных источников 10**

**1. Цель работы**

Цель данной лабораторной работы — изучение подходов к командной разработке в системе контроля версий git

**2. Задание**

Задание на лабораторную работу — В созданном репозитории совершить некоторый ряд действий:

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

Возможна совместная работа студентов над одним проектом для изучения взаимодействия группы разработчиков и возможностей работы с СКВ.

**3. Ход лабораторной работы**

Для работы будет использоваться репозиторий новый репозиторий. Чтобы настроить игнорирование Git'ом файлов определенного расширения для начала необходимо создать файл «.gitignore». Процесс создания файла и добавление его в репозиторий представлен на рис 1.

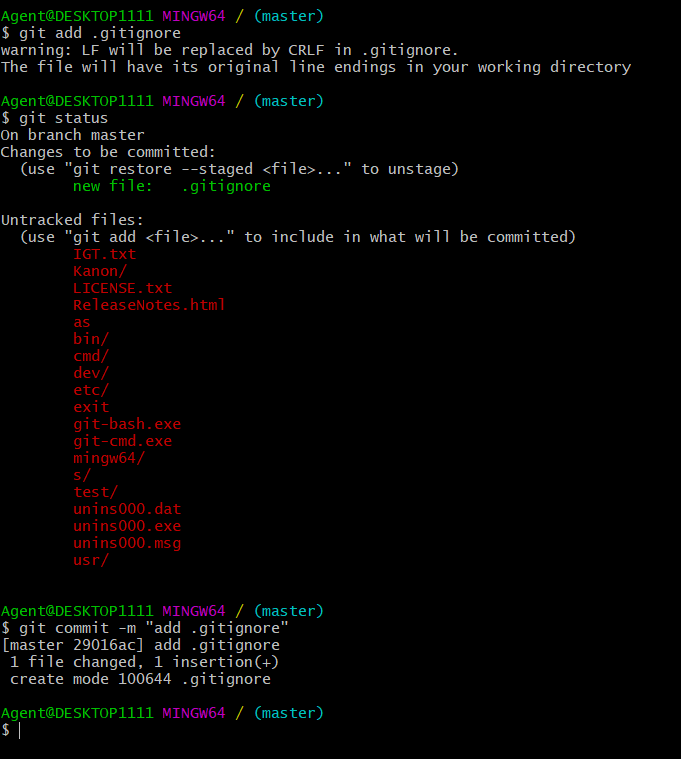


Рисунок 1 — Создание файла «.gitignore»

Внутри этого файла мы теперь можем указать, какой формат файлов будет игнорироваться Git'ом. Для примера возьмем файлы формата «.png» и файл с названием «IGT.txt». Теперь добавление или изменение файлов формата «.png» и изменение файла «IGT.txt» будут игнорироваться Git'ом. Содержимое файла «.gitignore» представлено на рис. 2.

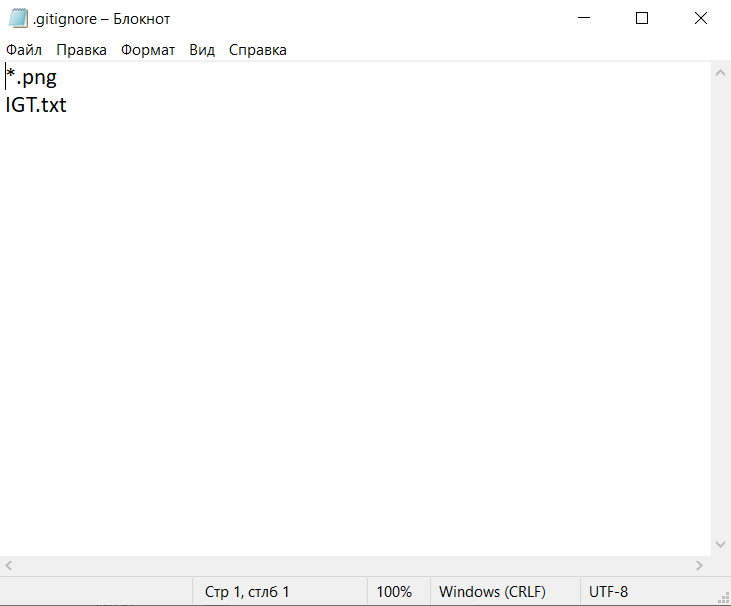


Рисунок 2 — Содержимое файла «.gitignore»

Вывести лог коммитов можно в текстовый файл при помощи команды «git log > log.txt». Содержимое файла «log.txt» представлено на рис. 3.

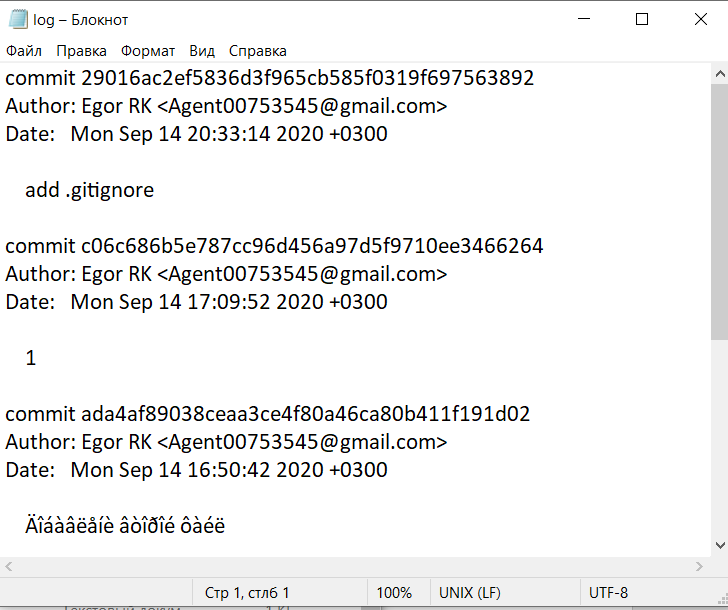


Рисунок 3 — Содержимое файла «log.txt»

Теперь попробуем переименовать созданную нами в прошлой лабораторной работе ветку «Kanon» в «TestKanon». Сделать это можно при помощи команды «git branch -m Kanon TestKanon». Теперь, перейдя в ветку «TestKanon», изменим содержимое файла «sss.txt», произведем коммит и сольем ветку с веткой master. Удалим слитую ветку при помощи команды «git branch -d TestKanon».

После этого создадим удаленный репозиторий под названием «stud» на сайте «[https://github.com](https://github.com/)». Мы можем связать локальный и удаленный репозитории при помощи команды «git remote add origin https://github.com/Имя\_пользователя/studying.git», после чего отправить содержимое локального репозитория на удаленный при помощи команды «git push -u origin master». Содержимое удаленного репозитория после произведенных операций представлено на рис. 4.

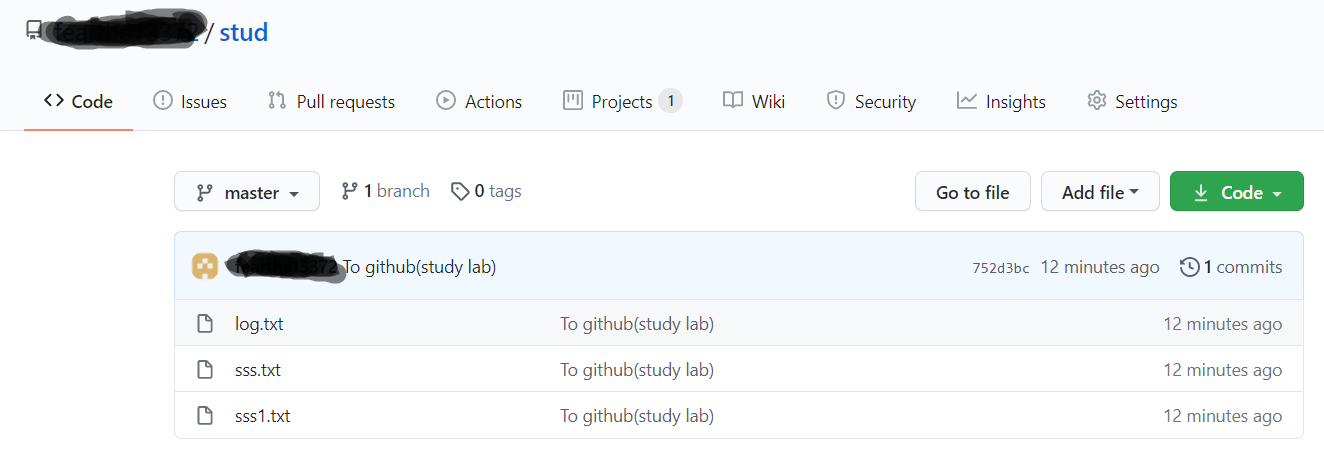


Рисунок 4 — Содержимое удаленного репозитория

**Вывод**

Таким образом, я изучил подходы к командной разработке в системе контроля версий Git.

**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