Министерство образования и науки России

ФГБОУ ВО "Севастопольский государственный университет"

Кафедра ИС

Отчет

по лабораторной работе № 3

**РАБОТА С ВЕТКАМИ в GIT**

Выполнил:

ст. гр. ИC-42

Лисянский А. И.

Проверил:

Севастополь

2015

**Цель работы:**

изучить особенности ветвления в Git.

**Ход работы**

1. Создайте ветку
2. Проверьте изменения (команда status)
3. Добавьте новый файл *Выполните:* touch <имя файла2. >
4. Зафиксируйте изменения. Теперь в Вашем проекте есть две ветки. *Выполните:* git hist --all или git log   
   Добавление опции --graph в git log вызывает построение дерева коммитов с помощью простых ASCII символов.
5. Выполните переключения к ветке master.
6. Сделайте изменения в ветке master.
7. Закомментируйте текущие изменения.
8. Вернитесь к созданной ранее ветке.
9. Просмотрите отличие текущих веток git hist –all   
   Метка --all гарантированно означает, что видны все ветки. По умолчанию показывается только текущая ветка.
10. Объедините ветки.
11. Вернитесь к созданной ветке в точку перед слиянием с master (необходимо найти последний коммит перед слиянием).
12. Проверьте ветку.
13. Поищите лог новой ветки.
14. Вернитесь в master и создайте конфликт.
15. Разрешите конфликт.
16. Сделайте коммит.
17. Объедините ветки.

**Выполнение**

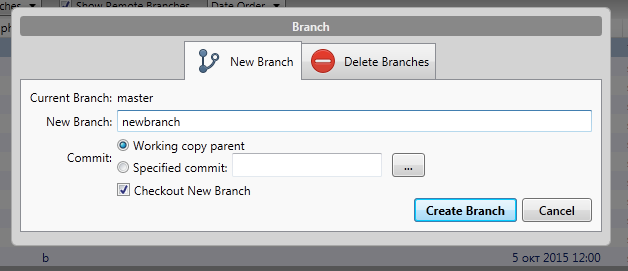


Рисунок 1 – Создание новой ветки

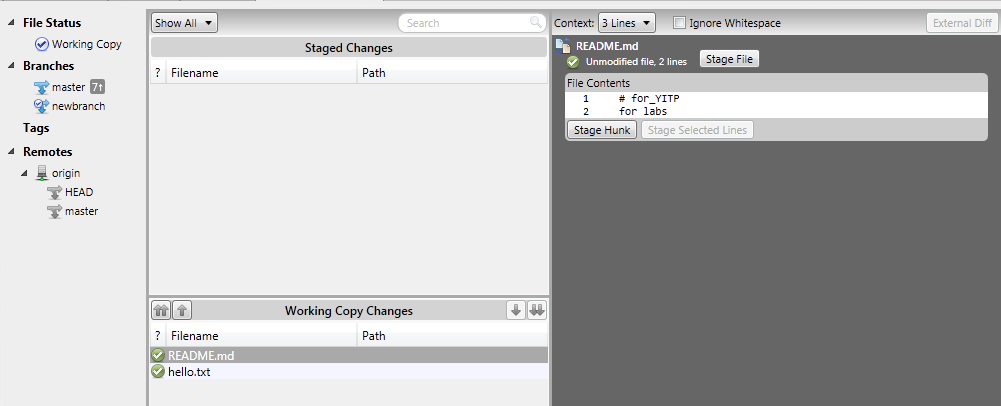


Рисунок 2 – проверка статуса новой ветки

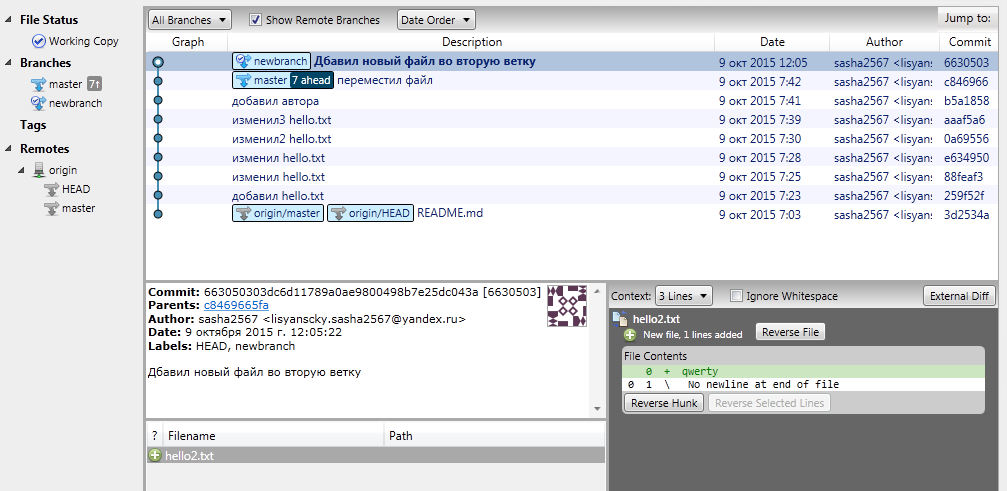


Рисунок 3 – История commitов всех веток

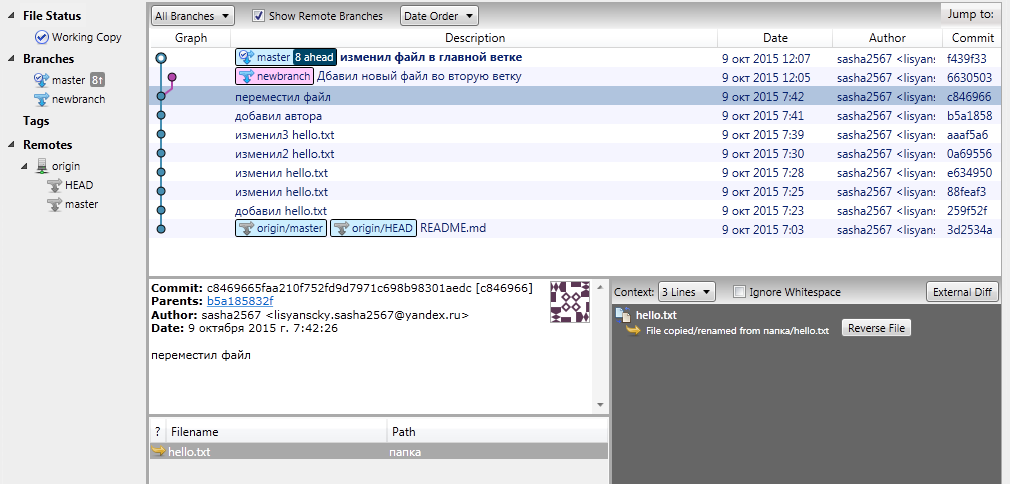


Рисунок 4 – Граф веток проекта с разными изменениями в файлах

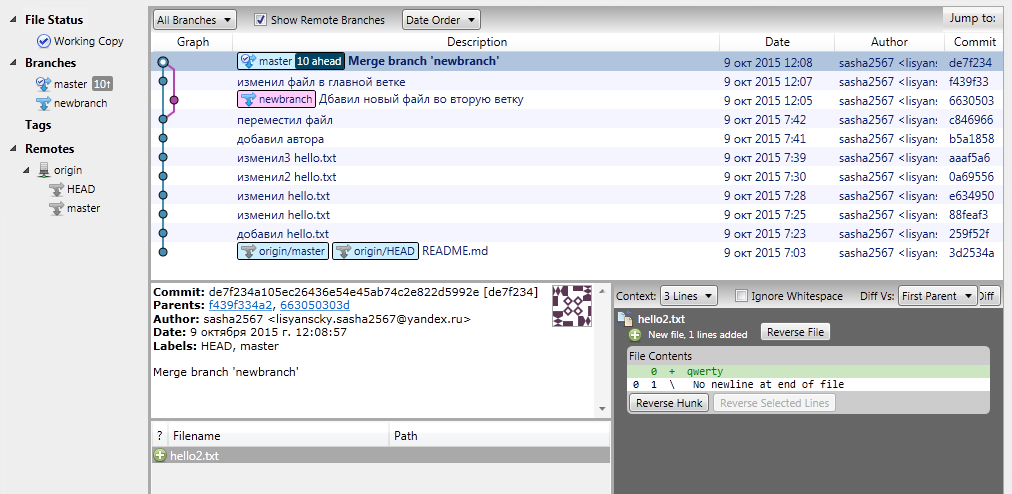


Рисунок 5 – Объединение веток

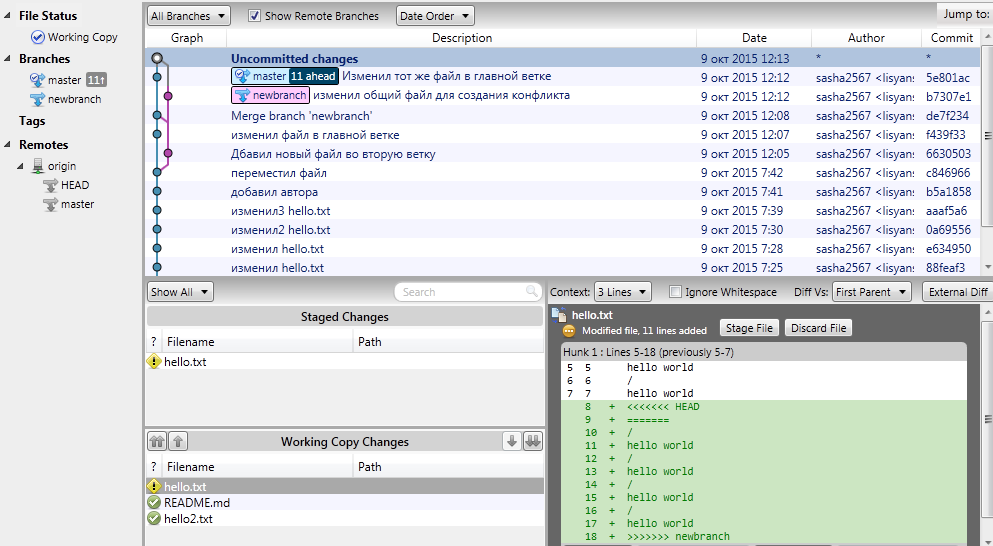


Рисунок 6 – появление конфликта

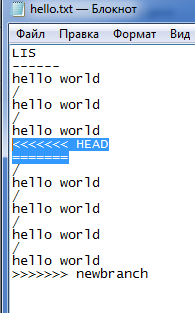


Рисунок 7 – решение конфликта – ручное решение

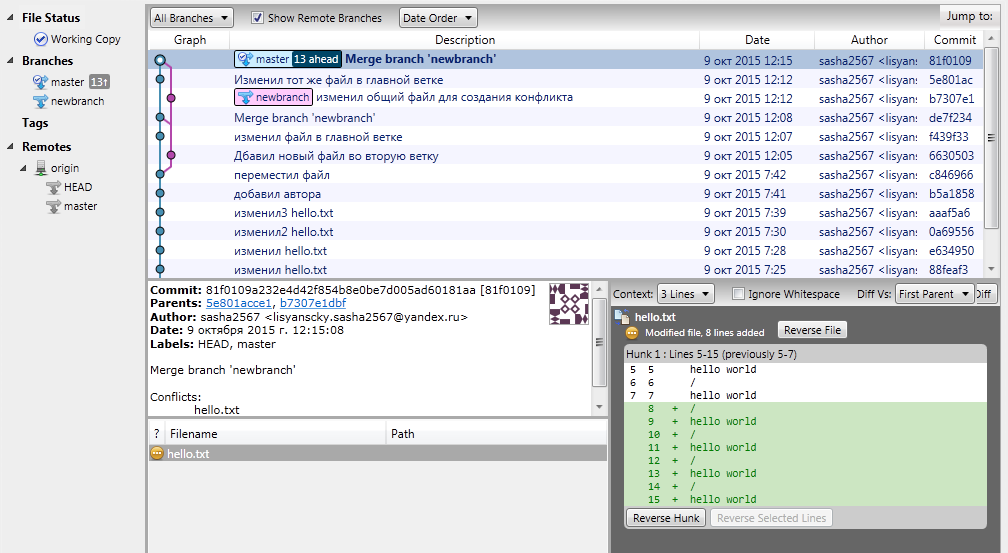


Рисунок 8 – решенный конфликт после слияния веток

**Вывод**

В ходе выполнения лабораторной работы были проделаны рядовые действия с локальным хранилищем, исследованы графические интерпретации команд гита по слиянию веток разработки.