ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ

"ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ"

Факультет ИСП

Кафедра ПИ им. Л.П. Фельдмана

Лабораторная работа №2

по курсу: «Профессиональная практика программной инженерии»

по теме: «Работа с ветками»

Выполнил:

ст. гр. ПИ-19в

Белошицкий Р.Е.

Проверили:

Дмитрюк Т.Г.

Филипишин Д.А.

ДОНЕЦК – 2023

Для первого модуля программы была создана ветка «workers» с помощью команды «git branch workers», после чего командой «git checkout workers» было выполнено переключение на неё. На новой ветке было сделано 5 фиксаций, их лог можно посмотреть с помощью команды «git log --pretty=format:\"%h %ad | %s%d [%an]\" --graph -date=short»(см. рис. 1).

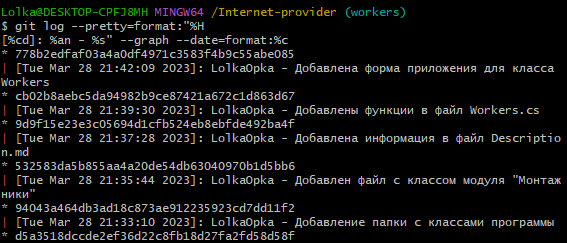


Рисунок 1 – Лог коммитов на ветке workers

Далее было создано ещё 2 ветки (по одной ветке для каждого модуля), на каждой ветке было сделано по 5 фиксаций (см. рис. 2-6).

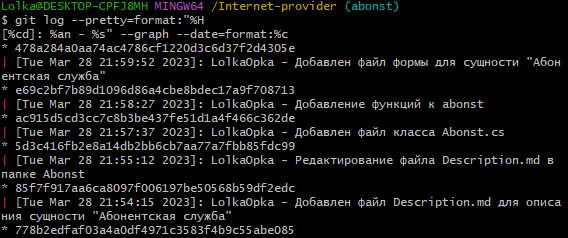


Рисунок 2– Лог коммитов на ветке abonst

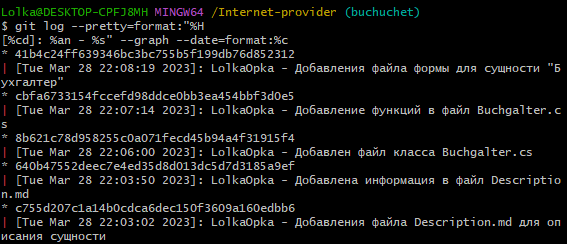


Рисунок 3 – Лог коммитов на ветке buchuchet

Каждая ветка была запушена на удалённый репозиторий на github с помощью команды «git push -u Internet-provider <имя\_ветки>». На рис. 7 можно посмотреть список веток репозитория на github.

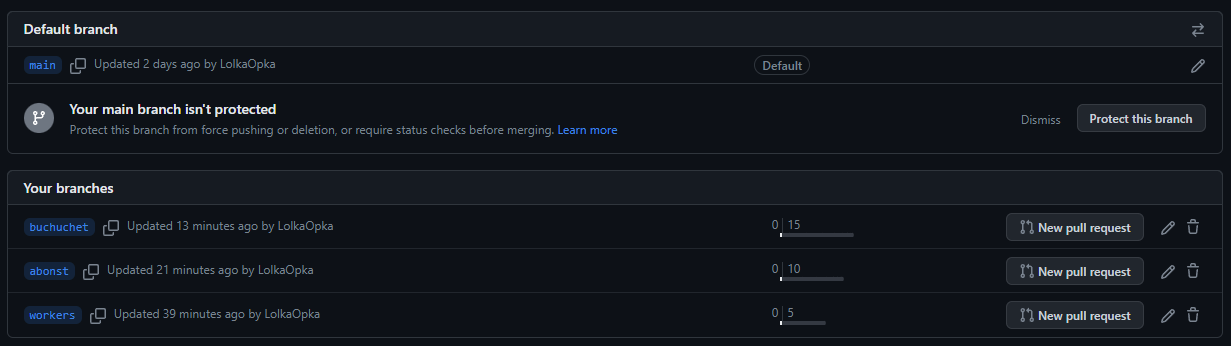


Рисунок 4 – Список веток

С помощью команды «git merge» 5 из 6 дополнительных веток были слиты с веткой «main».

Для того чтобы вызвать конфликт слияния, создадим в ветке main в папке Workers файл «Workers.cs» с таким же содержимым, но с другим значение переменной id и закоммитим его, после чего попытаемся выполнить слияние веток (см. рис. 5 - 11).]

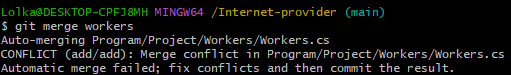


Рисунок 5 – Сообщение о конфликте

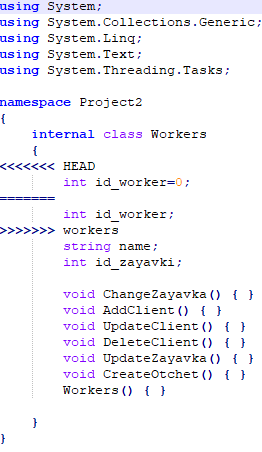


Рисунок 6 – Состояние конфликтного файла

В конфликтном файле с помощью разделителя «===» представлены варианты конфликтной строки из разных веток, система GIT предлагает нам выбрать один из вариантов и зафиксировать его, чтобы совершить слияние веток. После фиксации, если попытаться выполнить слияние ещё раз, система сообщит о том, что данные ветки уже содержаться в текущей ветке.

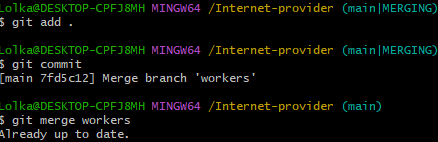


Рисунок 7 – Разрешение конфликта слияния

После слияния всех веток дерево проекта в ветке main теперь выглядит так, как показано на рисунке 8.

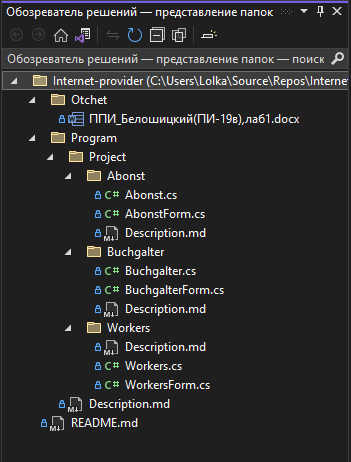


Рисунок 8 – Дерево проекта

Теперь посмотрим лог ревизии с помощью команды «git log --pretty=format:"%H[%cd]: %an - %s" --graph --date=format:%c»

и удалим все вторичные ветки с помощью команды «git branch -d <имя ветки>», команда с параметром «-d» может удалить только те ветки, которые были полностью слиты с другой веткой, чтобы удалить абсолютно любую ветку, можно воспользоваться параметром «-D». Для того чтобы удалить вторичные ветки и из удалённого репозитория была использована команда «git push --delete Internet Provider <имя\_удалённой\_ветки>».

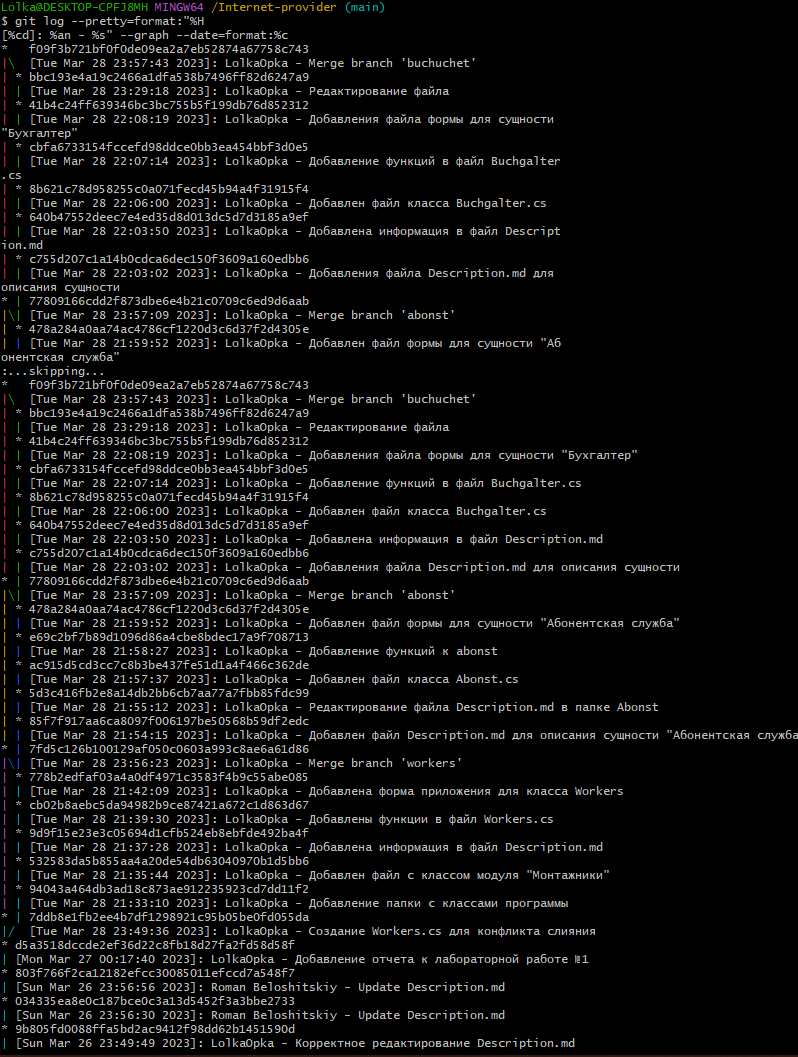


Рисунок 9 -Лог ревизии