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

Коваленко А.В.

Проверили:

Дмитрюк Т.Г.

Филипишин Д.А.

ДОНЕЦК – 2023

Для первого модуля программы была создана ветка «faculty» с помощью команды «git branch faculty», после чего командой «git checkout faculty» 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).

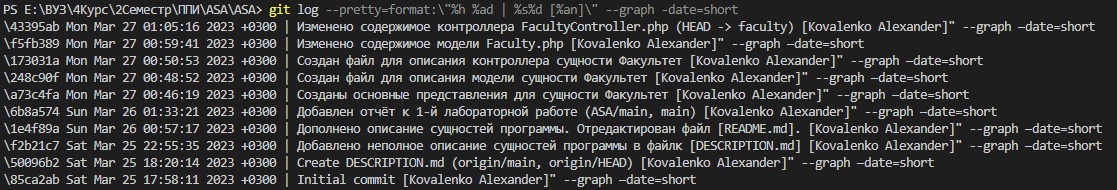


Рисунок 1 — Лог коммитов на ветке faculty

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

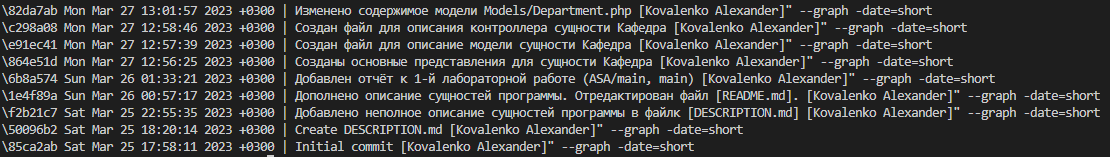


Рисунок 2 — Лог коммитов на ветке department

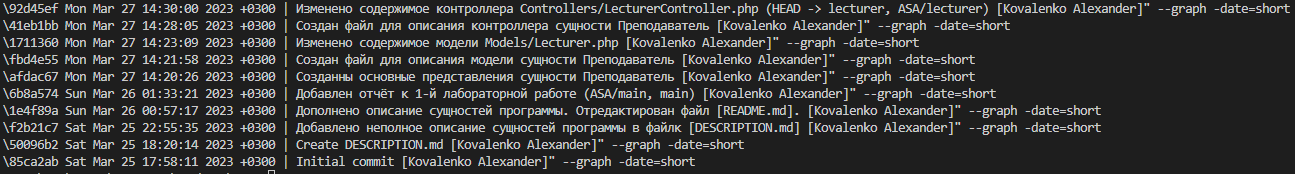


Рисунок 3 — Лог коммитов на ветке lecturer

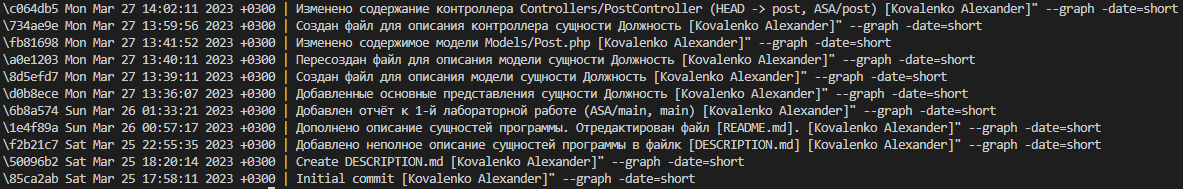


Рисунок 4 — Лог коммитов на ветке post

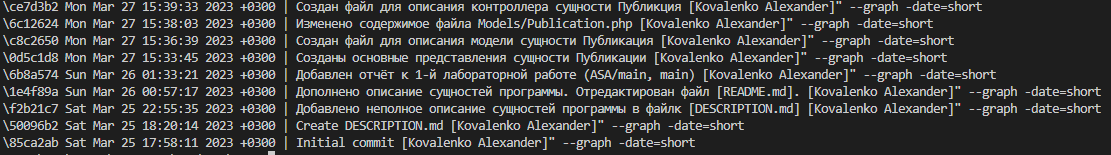


Рисунок 5 — Лог коммитов на ветке publication

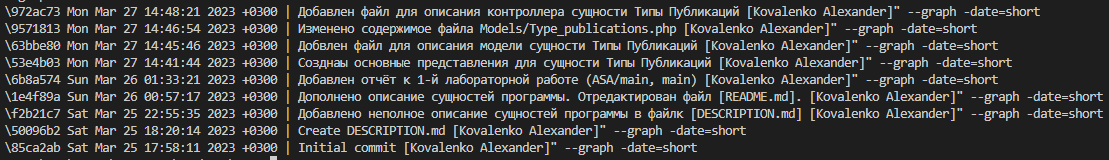


Рисунок 6 — Лог коммитов на ветке type\_publication

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



Рисунок 7 — Список веток

С помощью команды «git merge» 5 из 6 дополнительных веток были слиты с веткой «main» (см. рис. 8).



Рисунок 8 — Схема слияния веток

Для того чтобы вызвать конфликт слияния, создадим в ветке main в папке Models файл «Type\_publications.php» с таким же содержимым, но с другим значение переменной $table и закоммитим его, после чего попытаемся выполнить слияние веток (см. рис. 9 - 11).

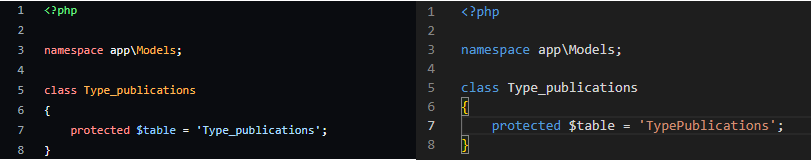


Рисунок 9 — Сравнение содержимого файла «Type\_publications.php» в ветках type\_publications и main

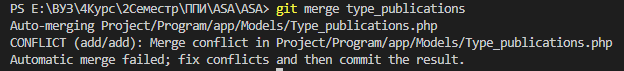


Рисунок 10 — Сообщение о конфликте слияния

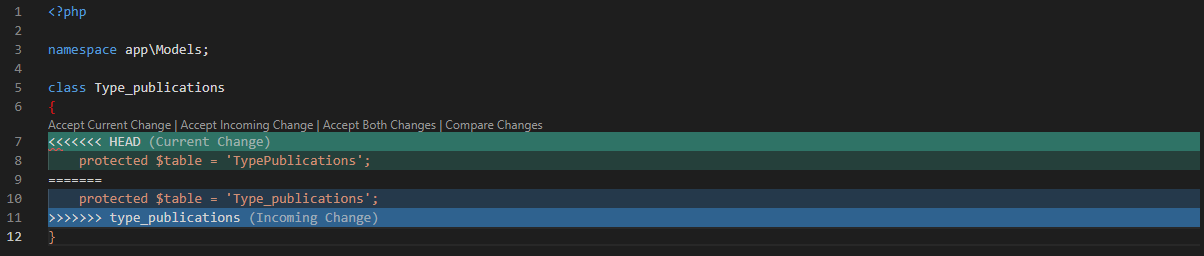


Рисунок 11 — Состояние конфликтного файла

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.

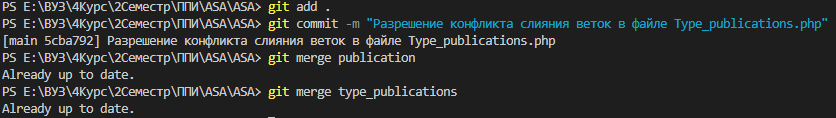


Рисунок 12 — Разрешение конфликта слияния

После слияния всех веток дерево проекта в ветке main теперь выглядит так, как показано на рисунке 13.

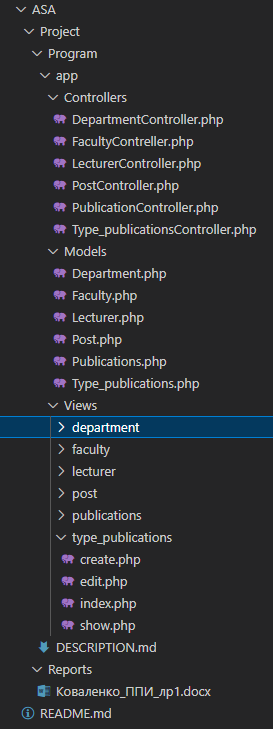


Рисунок 13 — Дерево проекта

Теперь посмотрим лог ревизии с помощью команды «git log --pretty=format:\"%h %ad | %s%d [%an]\" --graph —date=short» 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 ASA <имя\_удалённой\_ветки>».

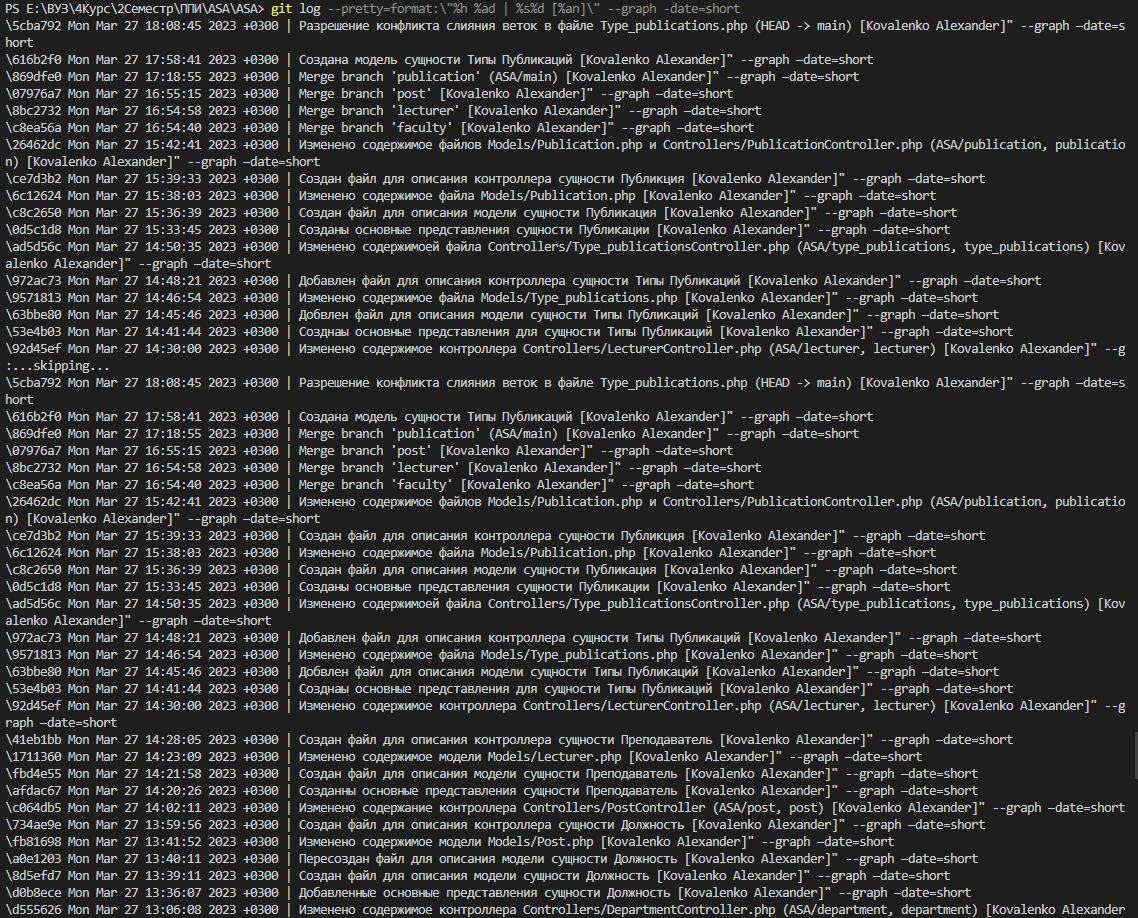


Рисунок 14 — Лог ревизии