Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Тихоокеанский государственный университет»

Высшая школа кибернетики и цифровых технологий

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1

ПО КУРСУ «ОСНОВЫ ПРОГРАММНОЙ ИНЖЕНЕРИИ»

«РАБОТА С GIT И GITHUB»

Выполнил: студент 2 курса группы ПО(б)-31

Разуваев Сергей Александрович

Проверила: ассистент ВШ КЦТ

Балабасова Алиса Дмитриевна

Хабаровск 2025 г.

Цель работы: освоить создание репозитория, добавление файлов, работу с ветками, форками и pull request.

Задание:

1. Создать репозиторий на GitHub, добавить описание проекта.

2. Настроить .gitignore, README.md (у себя локально). Закоммитить и отправить изменения в репозиторий (консольно, не через интерфейс GitHub).

3. Добавить бейдж статуса сборки в README.md.

4. Настроить SSH-ключ для аутентификации с GitHub.

5. Создать новую ветку "feature-branch" (у себя локально).

6. Внести изменения в код, закоммитить. Переключиться на основную ветку, объединить изменения. Разрешить возможные конфликты при слиянии.

7. Сделать форк существующего репозитория или своего же проекта.

8. Внести изменения в код, закоммитить и запушить с помощью ssh в форкнутый репозиторий.

9. Создать Pull Request между своим форком и оригинальным репозиторием (или между разными ветками в своём репозитории). Провести code review изменённого кода перед мержем, используя GitHub review tools.

Создание нового git-репозитория в GitHub (рисунок 1).

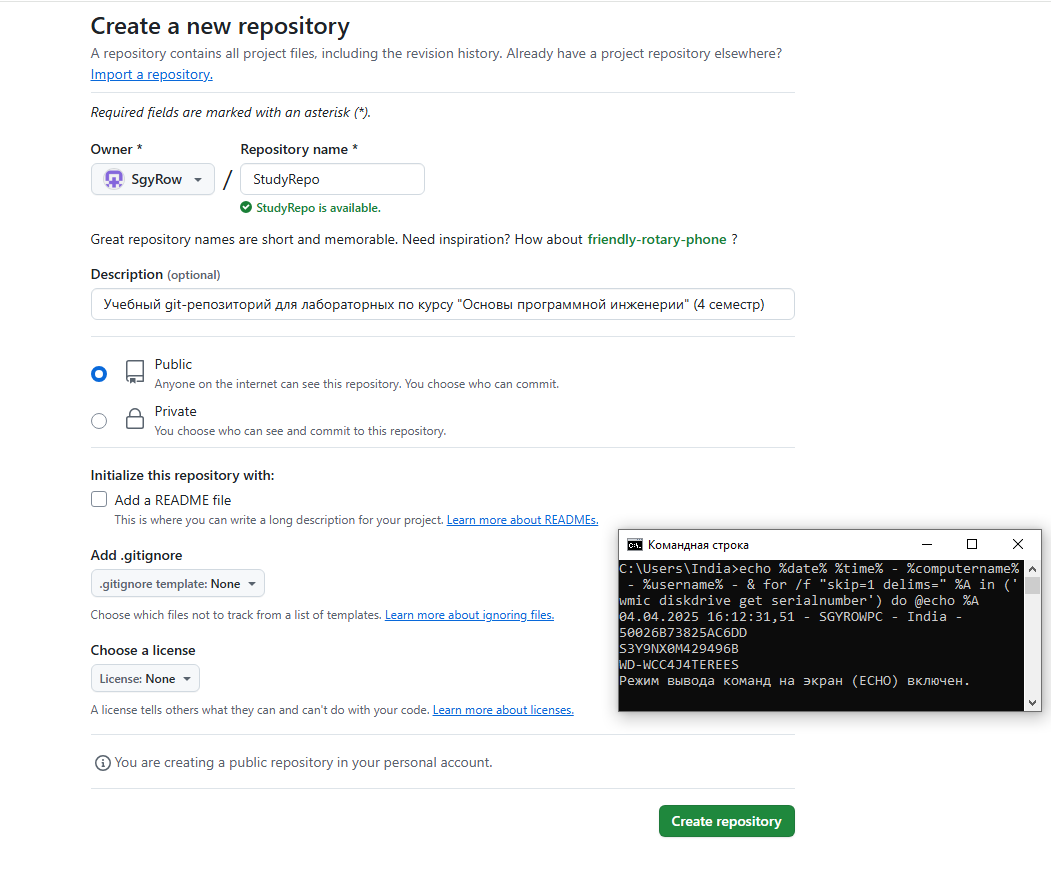


Рисунок 1 – Создание нового репозитория

Созданные и настроенные локально файлы .gitignore и README.md (рисунки 2, 3).

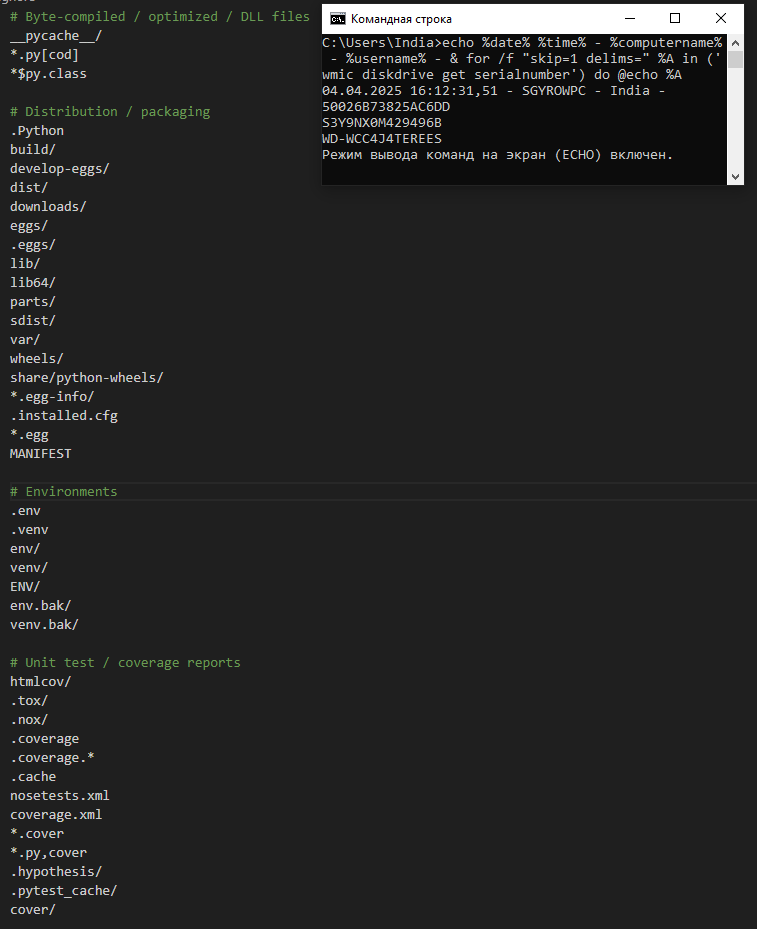


Рисунок 2 – Файл .gitignore

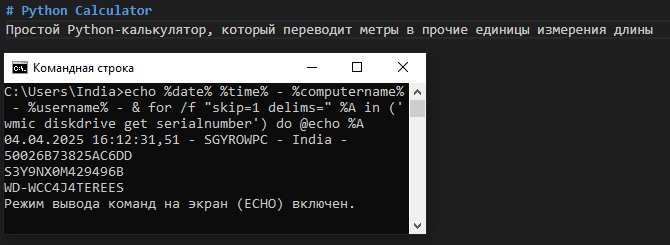


Рисунок 3 – Файл README.md

Коммит созданных файлов в репозиторий через bash (рисунок 4).

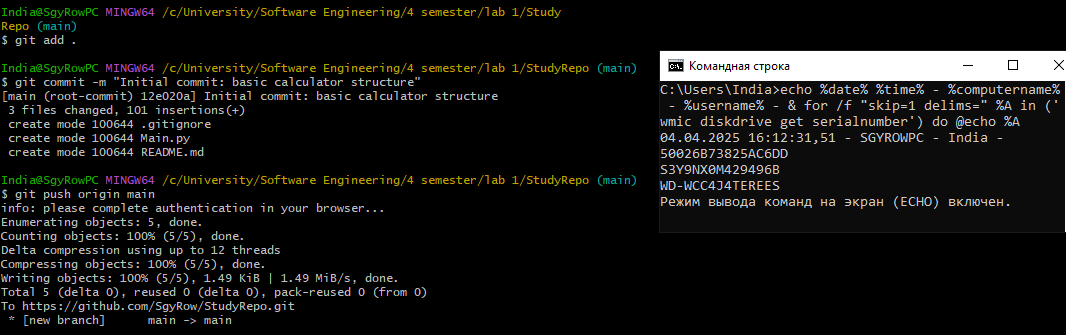


Рисунок 4 – Коммит созданных файлов

Настройка бейджа статуса сборки в README.md (рисунки 5, 6)

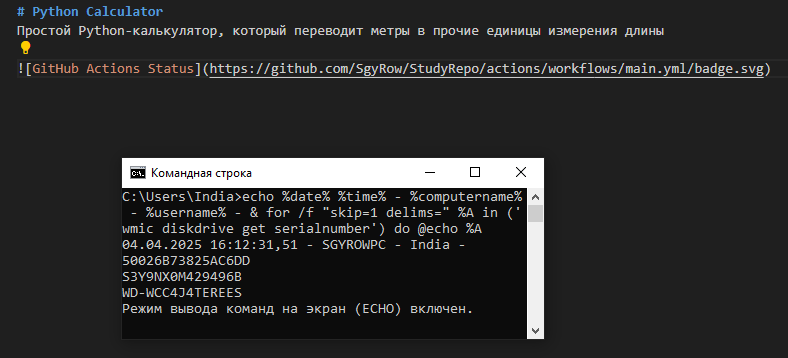


Рисунок 5 – Код бейджа статуса сборки в README.md

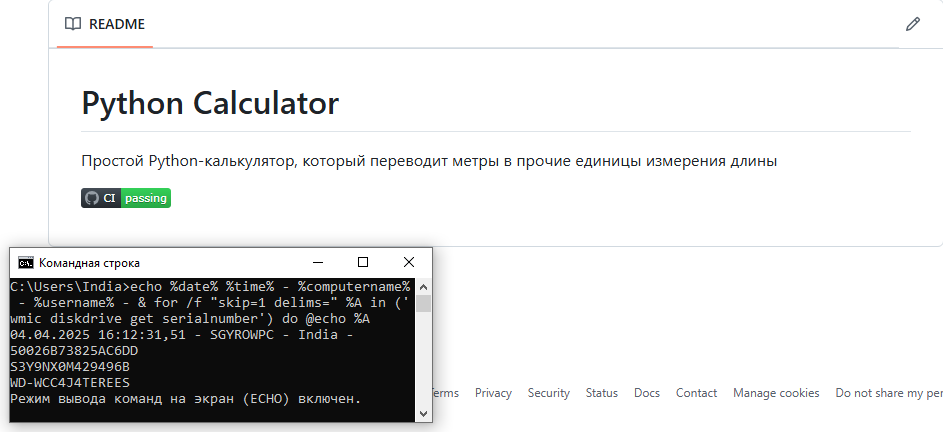


Рисунок 6 – Код бейджа статуса сборки в README.md

Настройка SSH-ключа для аутентификации с GitHub (рисунки 7, 8).

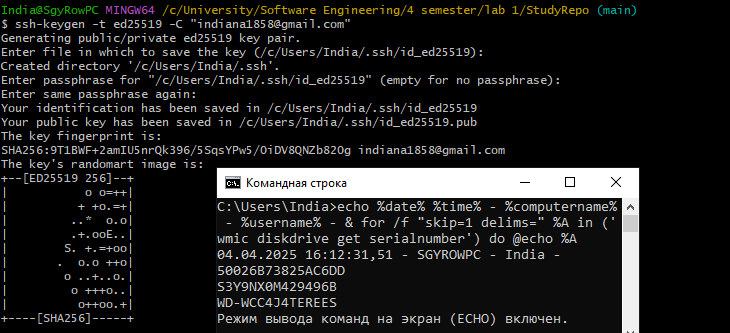


Рисунок 7 – Генерация SSH-ключа для git-репозитория

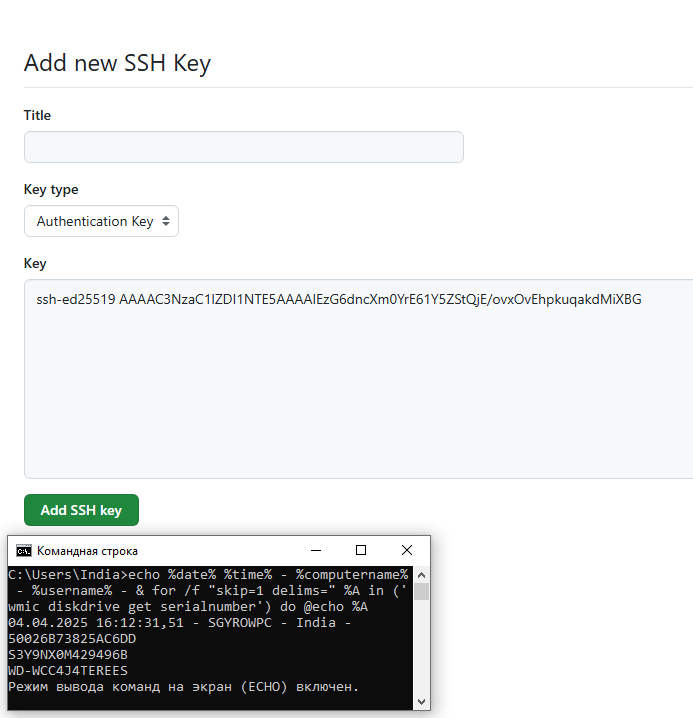


Рисунок 8 – Добавление SSH-ключа для git-репозитория

Создание новой ветки feature-branch (рисунок 9).

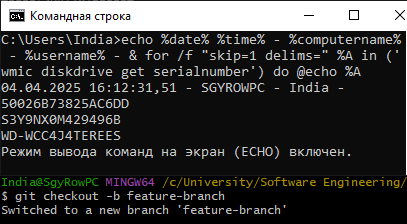


Рисунок 9 – Создание ветки feature-branch

Внесение изменений в код (дополнение Main.py) (рисунок 10).

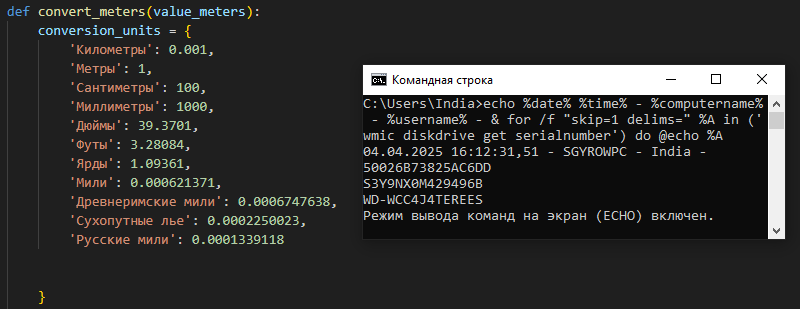


Рисунок 10 – Внесение изменений в код

Коммит в новую ветку feature-branch (рисунок 11).

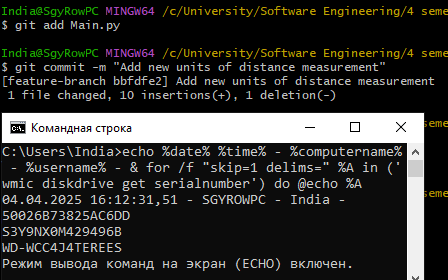


Рисунок 11 – Коммит в новую ветку

Слияние двух веток в одну (рисунок 12).

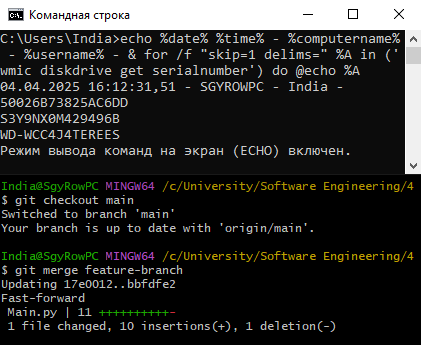


Рисунок 12 – Слияние двух веток

Создание форка репозитория и его копирование на ПК (рисунок 13).

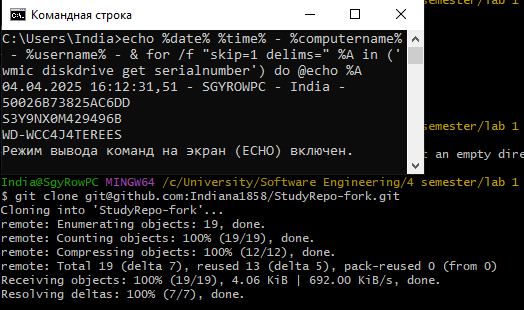


Рисунок 13 – Форк репозитория

Коммит и пуш в форк репозитория (рисунок 14).

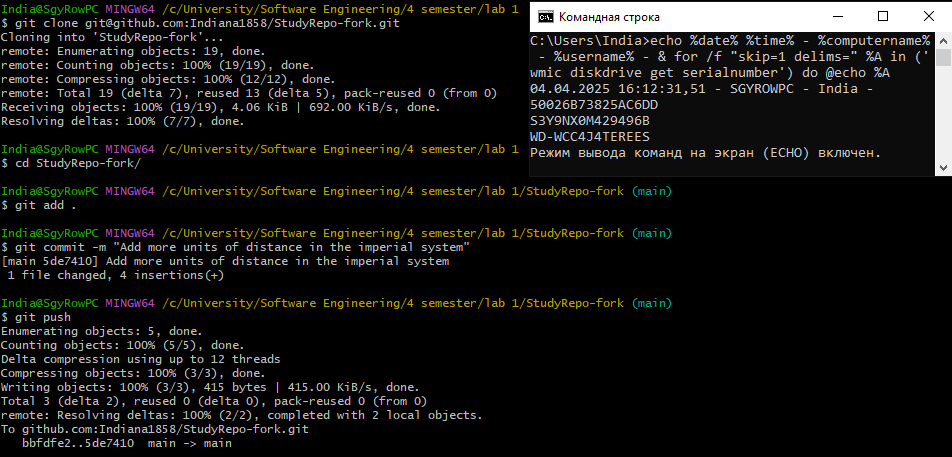


Рисунок 14 – Коммит и пуш в форк

Создание Pull Request-а (рисунок 15).

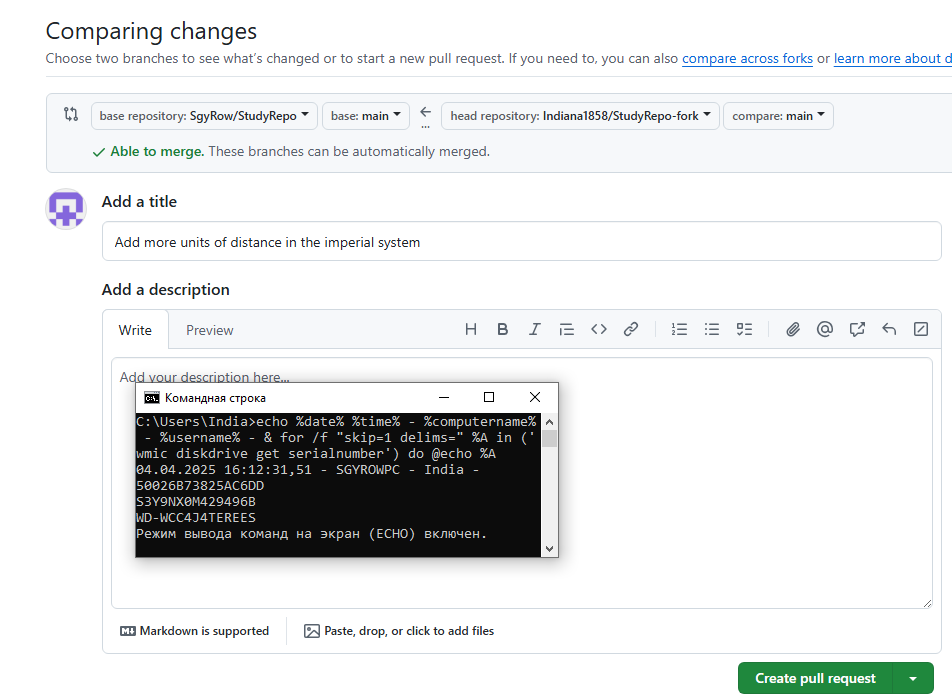


Рисунок 15 – Pull Request

Итоговый внешний вид репозитория (рисунок 16).

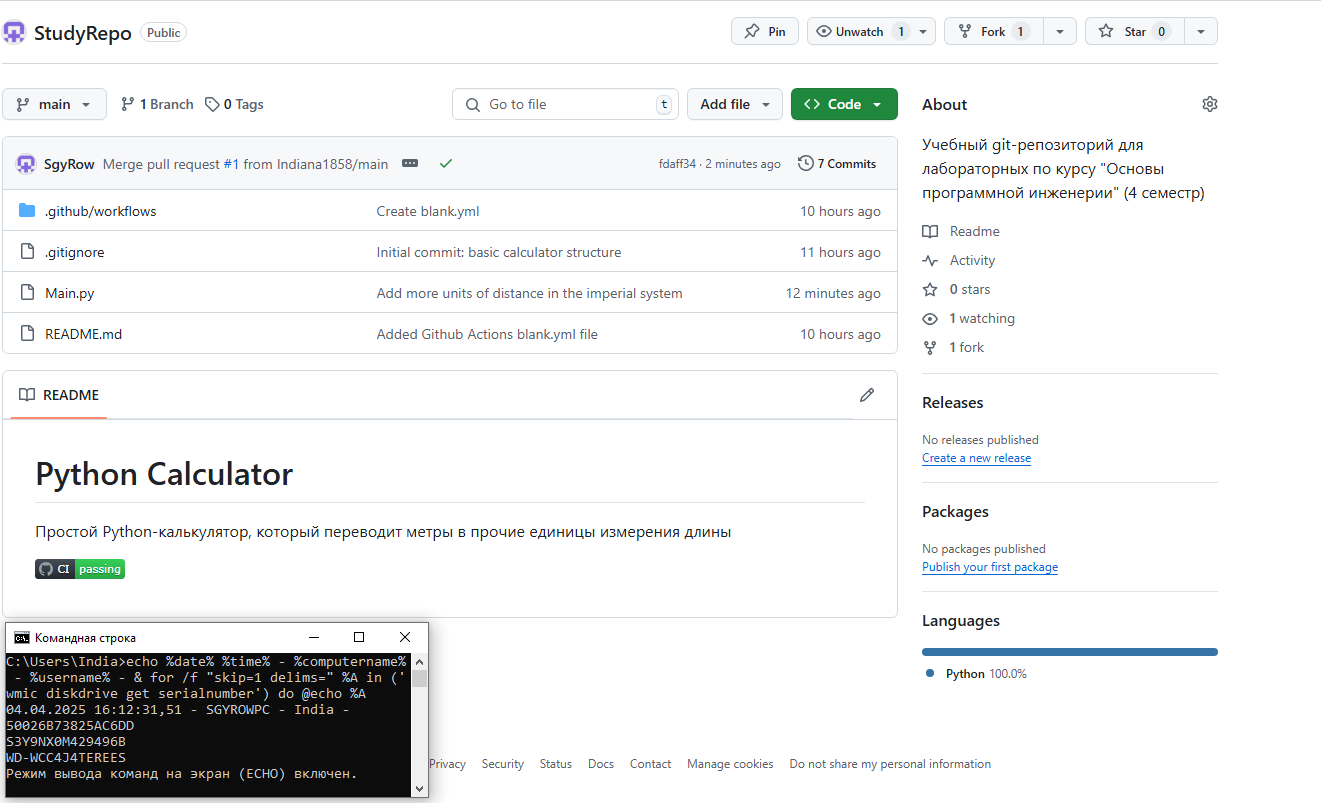


Рисунок 16 – Итоговый репозиторий

Вывод: В ходе выполнения лабораторной работы был создан новый git-репозиторий на GitHub; была настроена работа системы контроля версий Git.

После создания в репозитории были настроены файлы .gitignore и README.md, все изменения через Git Bash отправлены в репозиторий. Была произведена начальная настройка GitHub Actions, в README.md добавлен бейдж статуса сборки. Выполнено создание новых веток, коммиты и пуши изменений в программном коде: произведено слияние двух веток в одну. Настроен SSH-ключ для аутентификации с GitHub: ключ был сгенерирован, а затем добавлен в профиль аккаунта GitHub.

Был создан форк собственного репозитория, произведён коммит и пуш при помощи раннее настроенного SSH-ключа. После этого произведён Pull Request между изначальным и форкнутым репозиториями, разрешены все возможные конфликты, выполнен мерж слияния.