

**МИНОБРНАУКИ РОССИИ**

**федеральное государственное автономное образовательное учреждение**

**высшего образования**

**«Московский государственный технологический университет «СТАНКИН»**

**(ФГАОУ ВО «МГТУ «СТАНКИН»)**

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | | Институт цифровых интеллектуальных систем | Кафедра  компьютерных систем управления | |

Дисциплина «Основы системного программного обеспечения»

**Отчет по лабораторной работе №1**

|  |  |  |
| --- | --- | --- |
| **Выполнил**  **студент гр. АДБ-22-06:** | \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_  *(дата) (подпись)* | **Сидоров А.В.** |
| **Проверил**  **к.т.н., доцент** | \_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_  *(дата) (подпись)* | **Ковалев И.А.** |

Оглавление

[1. Цель и задачи работы 3](#_Toc197049399)

[2. Ход работы 3](#_Toc197049400)

[2.1 Регистрация и первичная настройка GitHub 3](#_Toc197049401)

[2.2 Создание удалённого репозитория 3](#_Toc197049402)

[2.3 Установка и глобальная настройка Git 4](#_Toc197049403)

[2.4 Создание папки и файла 5](#_Toc197049404)

[2.5 Инициализция. Добавление и фиксация изменений 6](#_Toc197049405)

[2.6 Связывание локального и удалённого репозиториев 7](#_Toc197049406)

[2.7 Работа с ветками 9](#_Toc197049407)

[2.8 Откаты изменений 13](#_Toc197049408)

[2.9 Игнорирование файлов 15](#_Toc197049409)

[2.10 Совместная работа (Fork → Clone → Pull Request) 17](#_Toc197049410)

[4. Результаты 17](#_Toc197049411)

[5. Выводы 17](#_Toc197049412)

# 1. Цель и задачи работы

1. Закрепить навыки базовой работы с Git в локальном и удалённом репозитории.
2. Освоить типовой рабочий процесс: add → commit → push и pull → merge.
3. Научиться создавать ветки, решать конфликты, откатываться к предыдущим версиям.
4. Освоить совместную работу через Fork/Clone и файл .gitignore для исключения лишних данных.

# 2. Ход работы

## 2.1 Регистрация и первичная настройка GitHub

1. Перешёл на https://github.com и создал бесплатный публичный аккаунт.

2. В Settings → Repositories → Default branch сменил название основной ветки с main на master.

## 2.2 Создание удалённого репозитория

1. Нажал New repository → ввёл имя firstApp, описание и оставил тип Public.

2. Поставил галочку «Add README». В результате GitHub создал репозиторий.

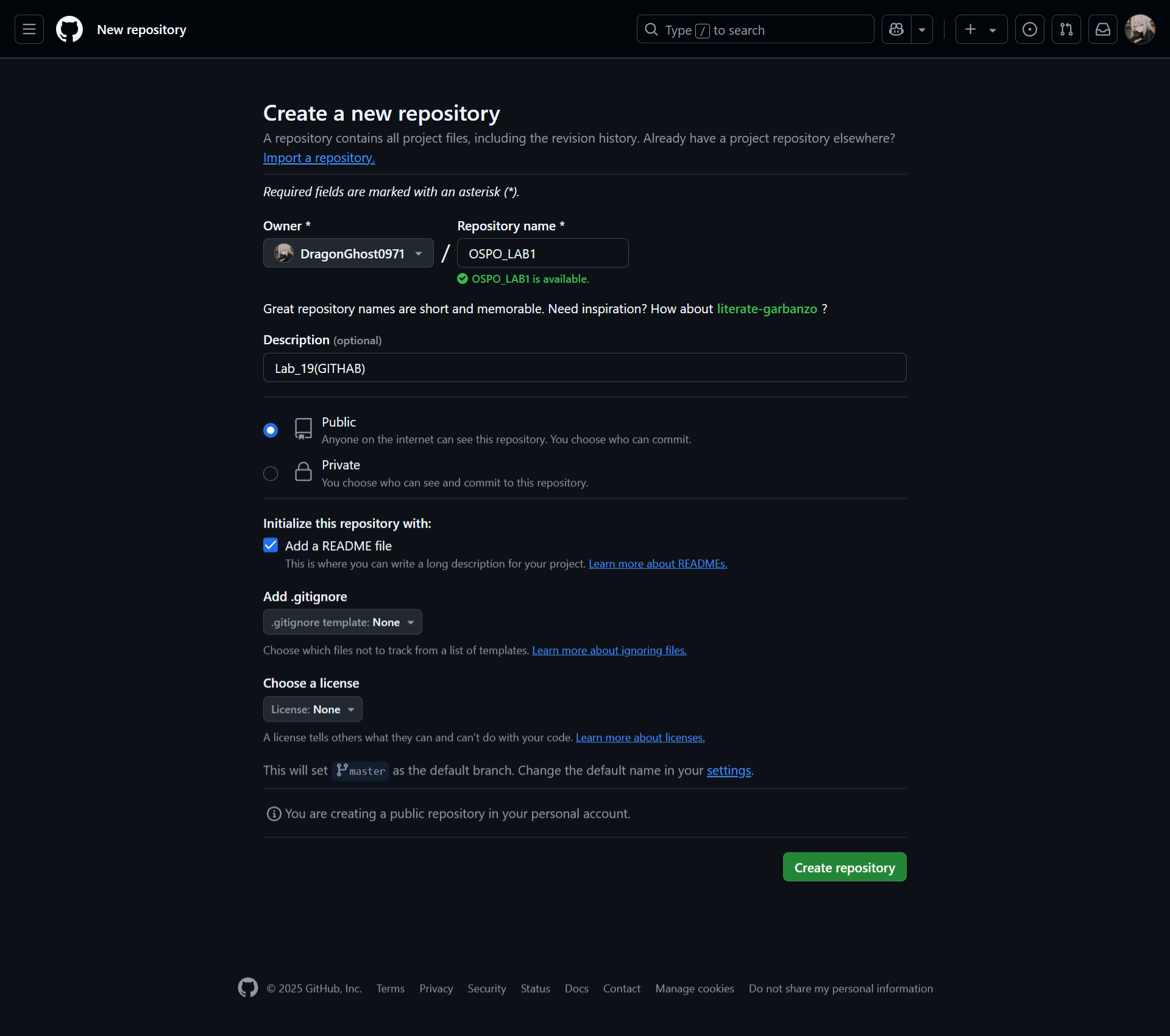


Рис. 1 – Создание репозитория

## 2.3 Установка и глобальная настройка Git

git --version  
git config --global user.name "Name"  
git config --global user.email "mail@mail.ru"

После ввода команд данные сохраняются в ~/.gitconfig; каждое дальнейшее изменение будет подписано правильным именем и почтой.

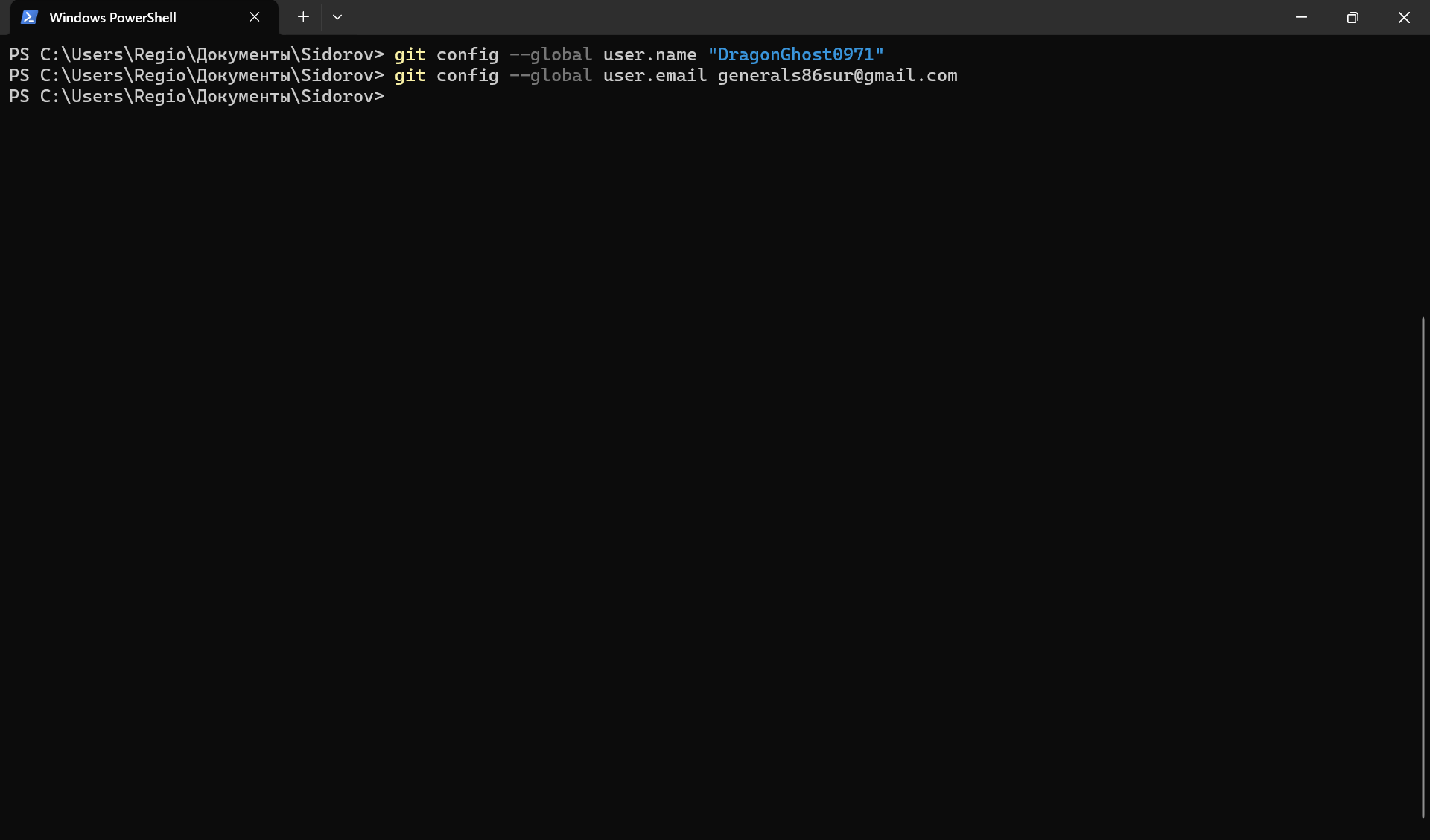


Рис. 2 - Глобальная настройка git

## 2.4 Создание папки и файла

Создал папку со своей фамилией и файла с помощью команд:

Mkdir Имя\_папки  
Echo > имя\_файла.txt

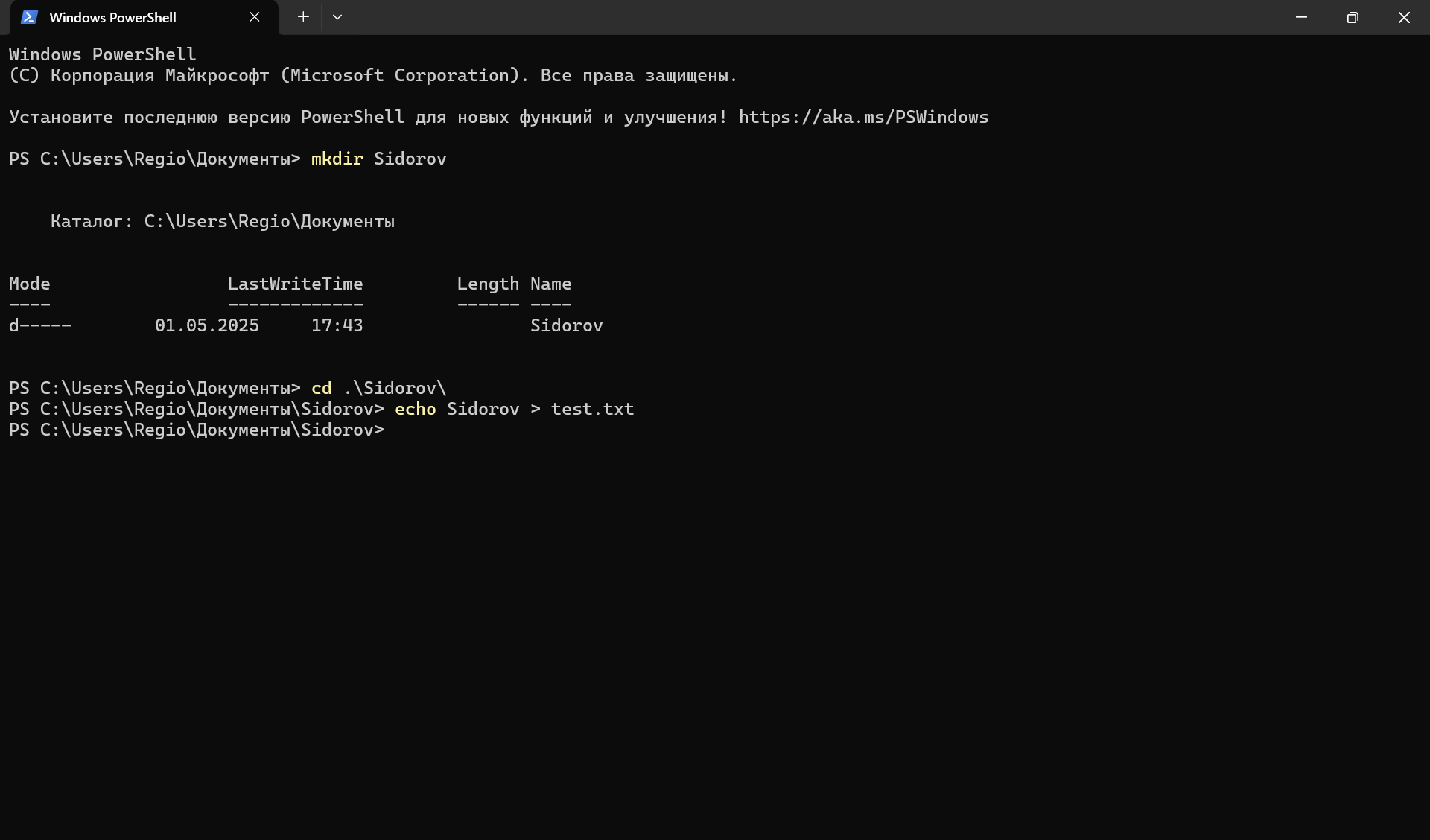


Рис. 3 - Создание папки и текстового файла

## 2.5 Инициализция. Добавление и фиксация изменений

git init  
git add .  
git commit -m "first commit"

Git создал снимок (commit) с уникальным SHA-1.

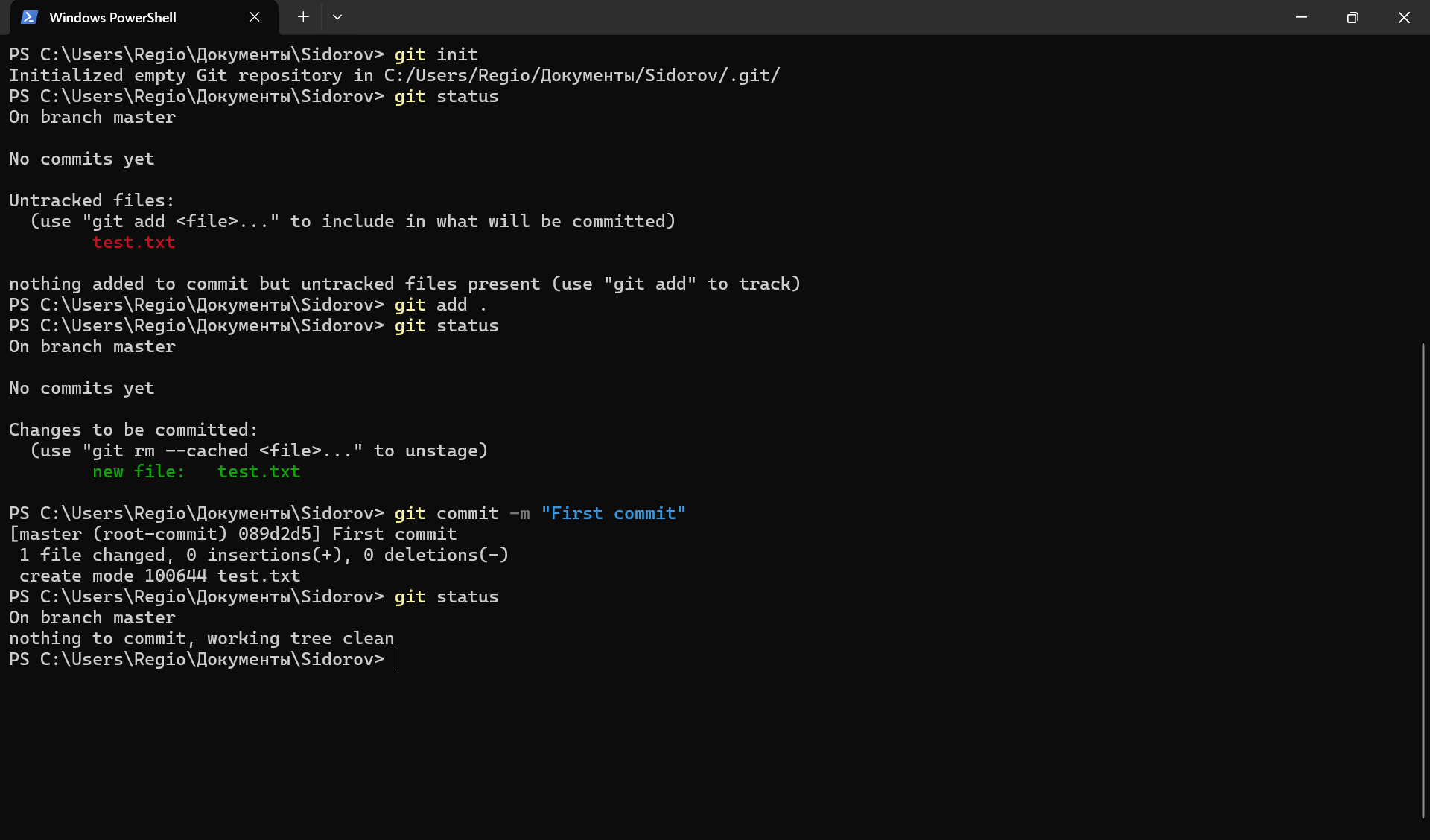


Рис. 4 - Инициализация. Добавление и фиксация иземенений

## 2.6 Связывание локального и удалённого репозиториев

git remote add origin url  
git pull origin master --allow-unrelated-histories

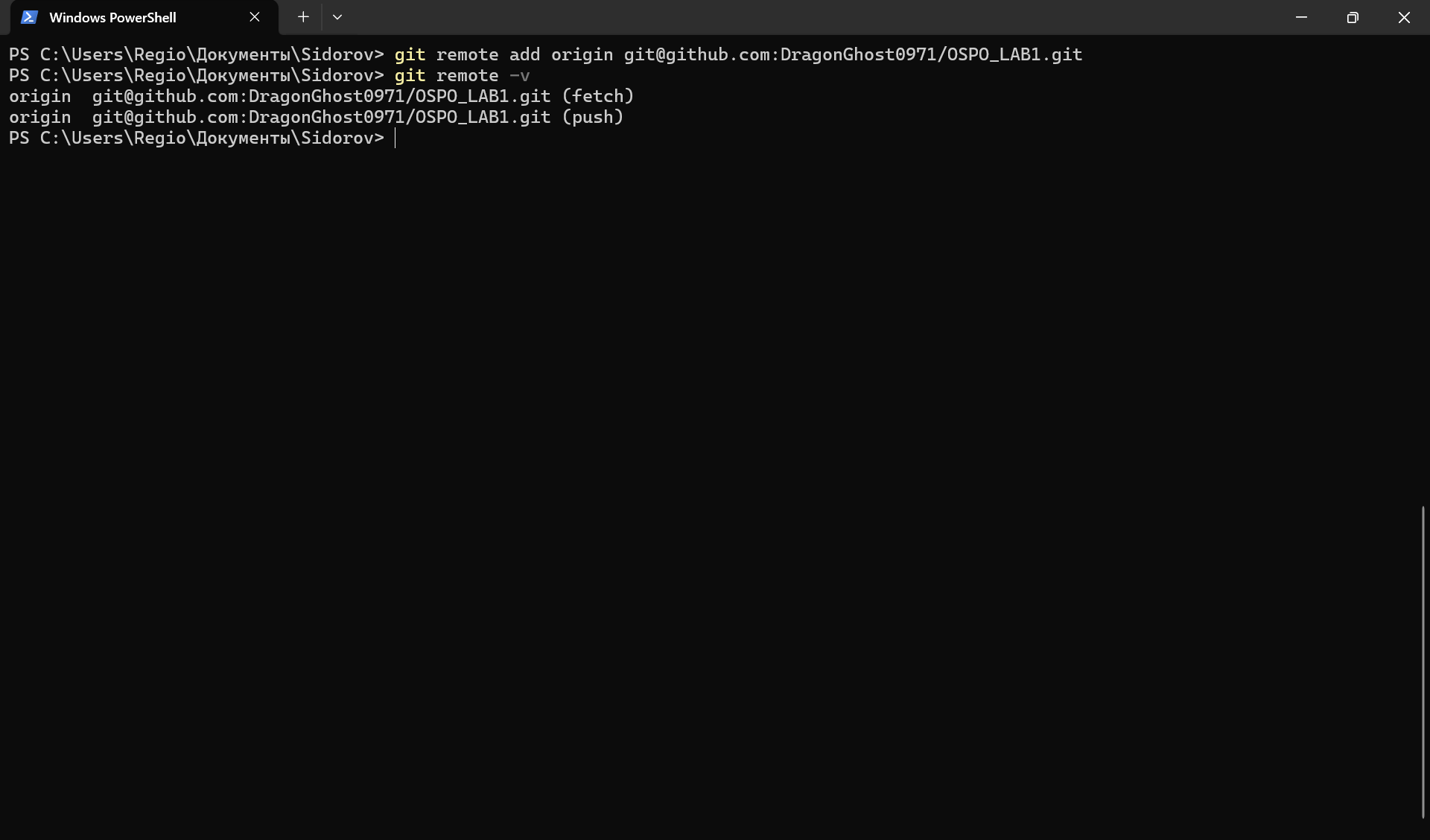


Рис. 5 - связывание локального с удалённым репозиторием

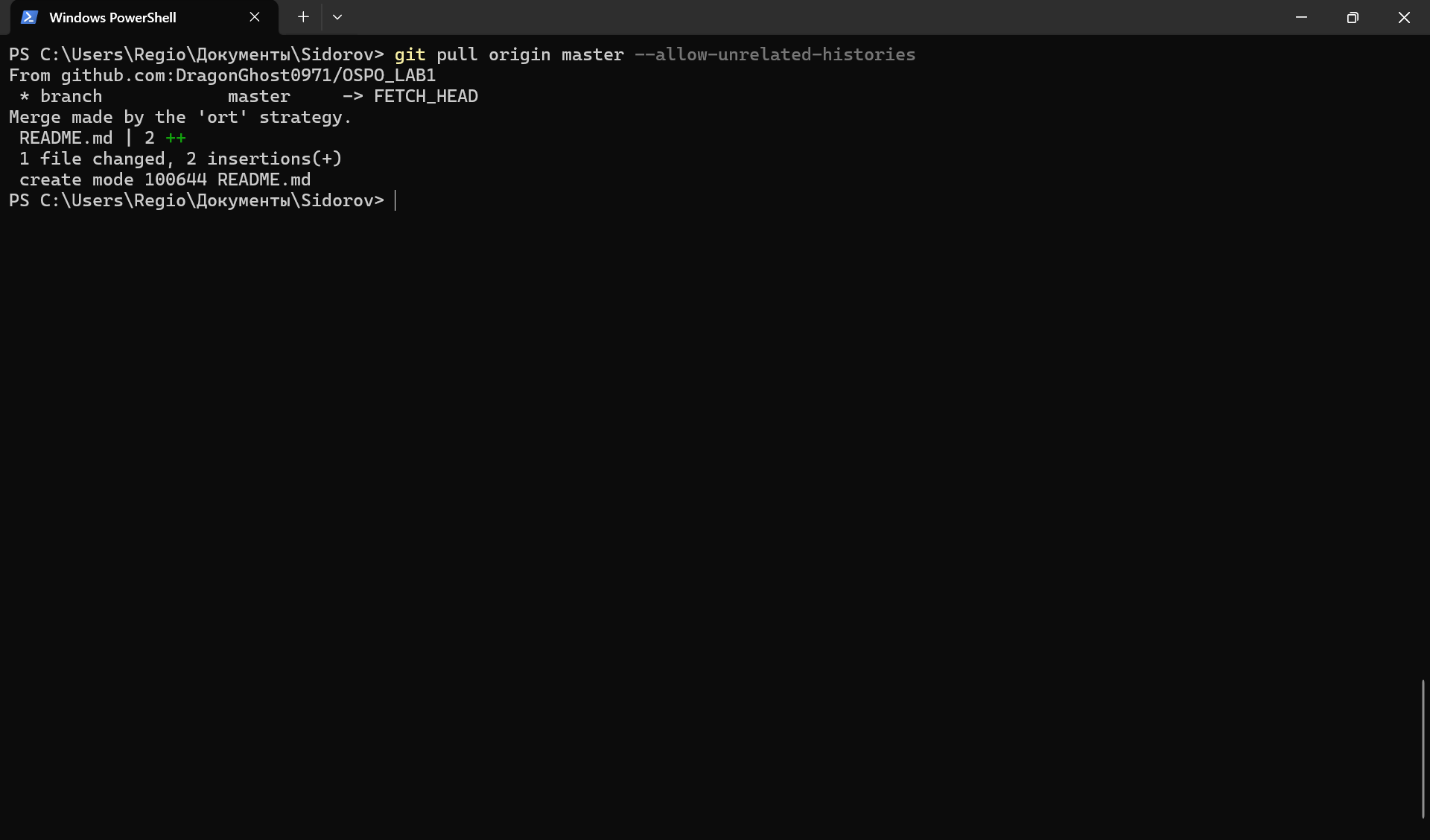


Рис. 6 - Подтягивание удалённого репозитория к локальному

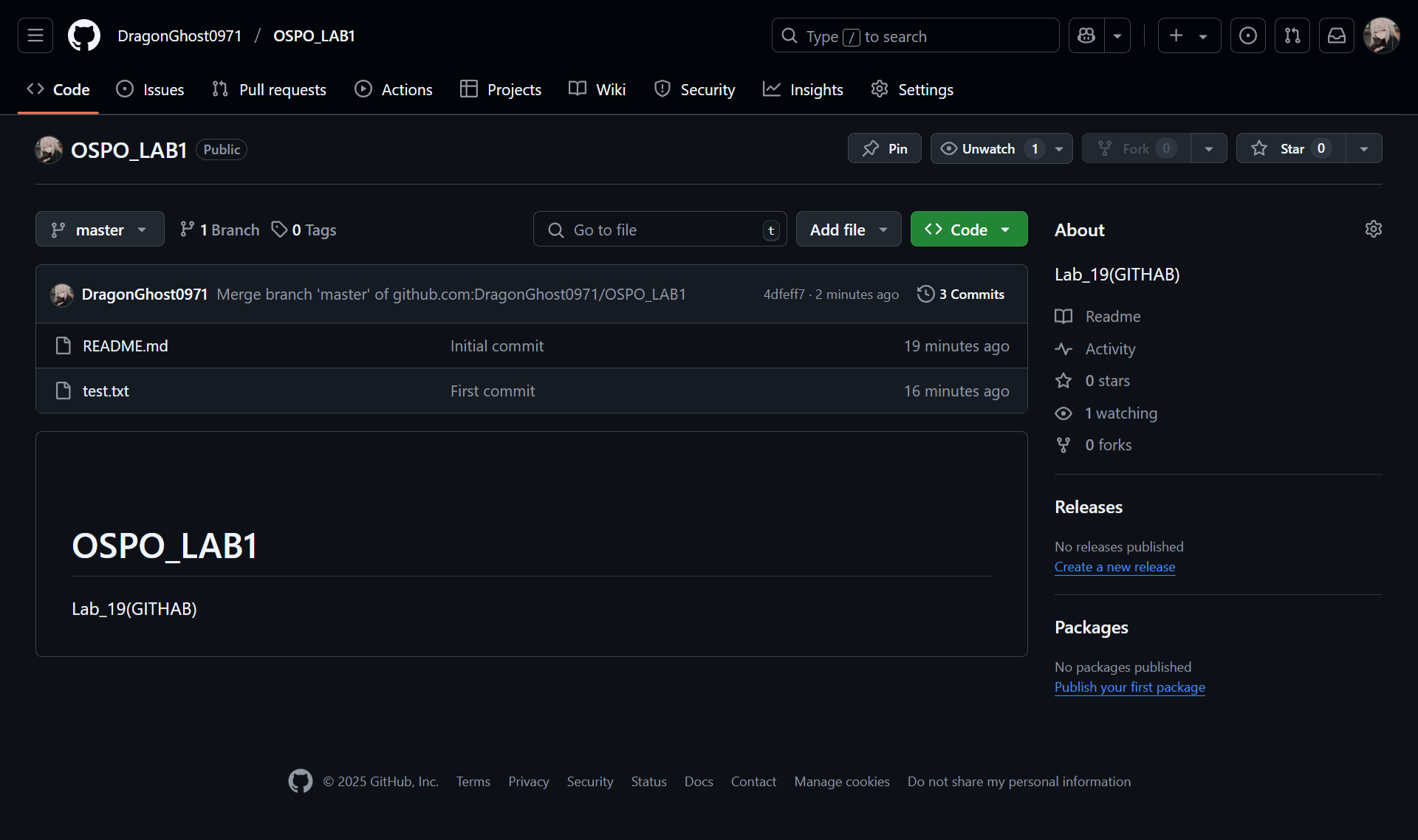


Рис. 7 - Удалённый репозиторий

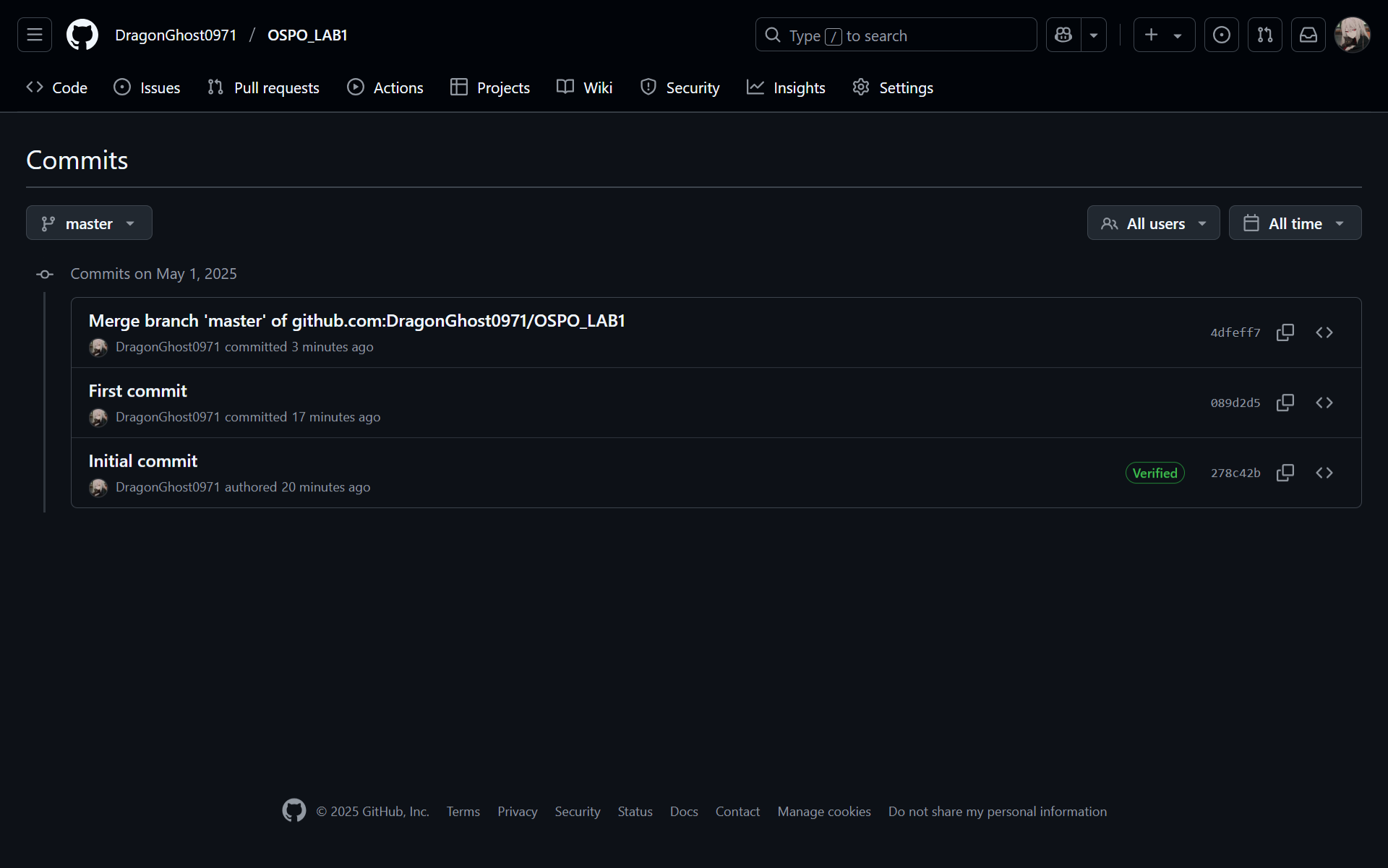


Рис. 8 - Коммиты удалённого репозитория

## 2.7 Работа с ветками

Создание новой ветки:

git branch second  
git checkout second  
git add .  
git commit -m "add test\_new.txt"  
git push -u origin second

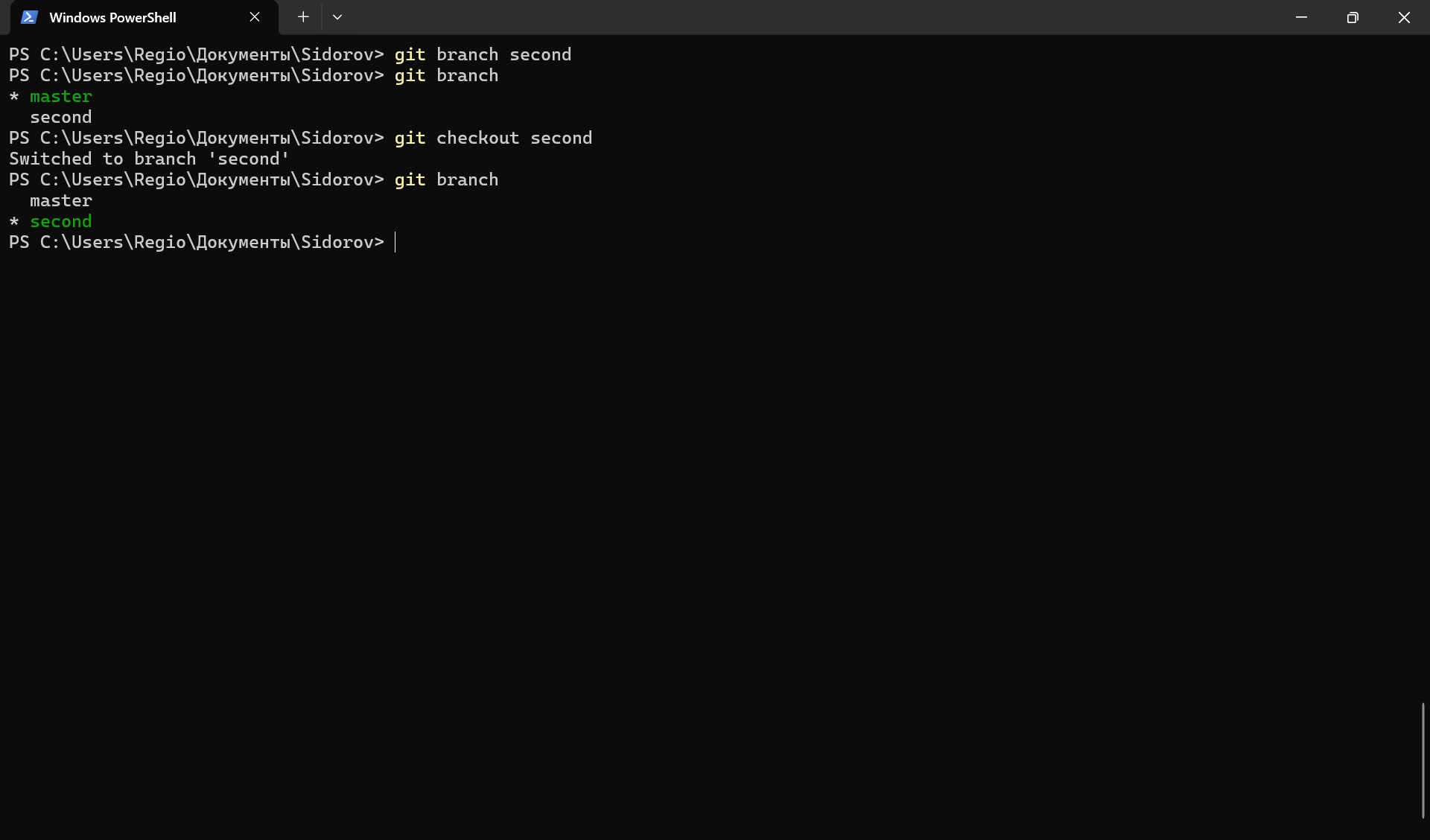


Рис. 9- Создание ветки second

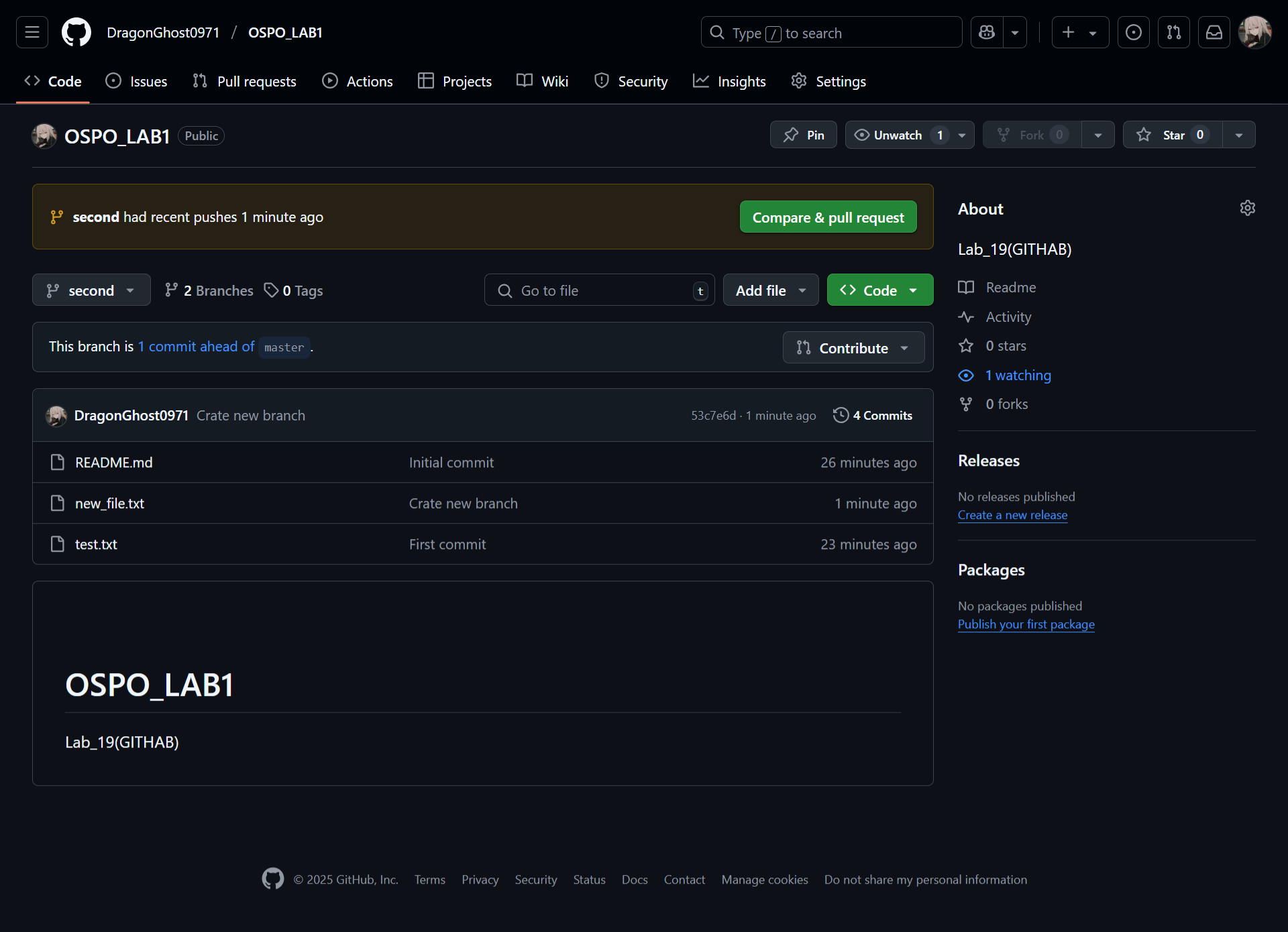


Рис. 10 - ветка second на удалённом репозитории

Слияние и удаление ветки:

git checkout master  
git merge second  
git push origin master  
git branch -d second  
git push origin --delete second



Рис. 11 - Слияние second с master веткой

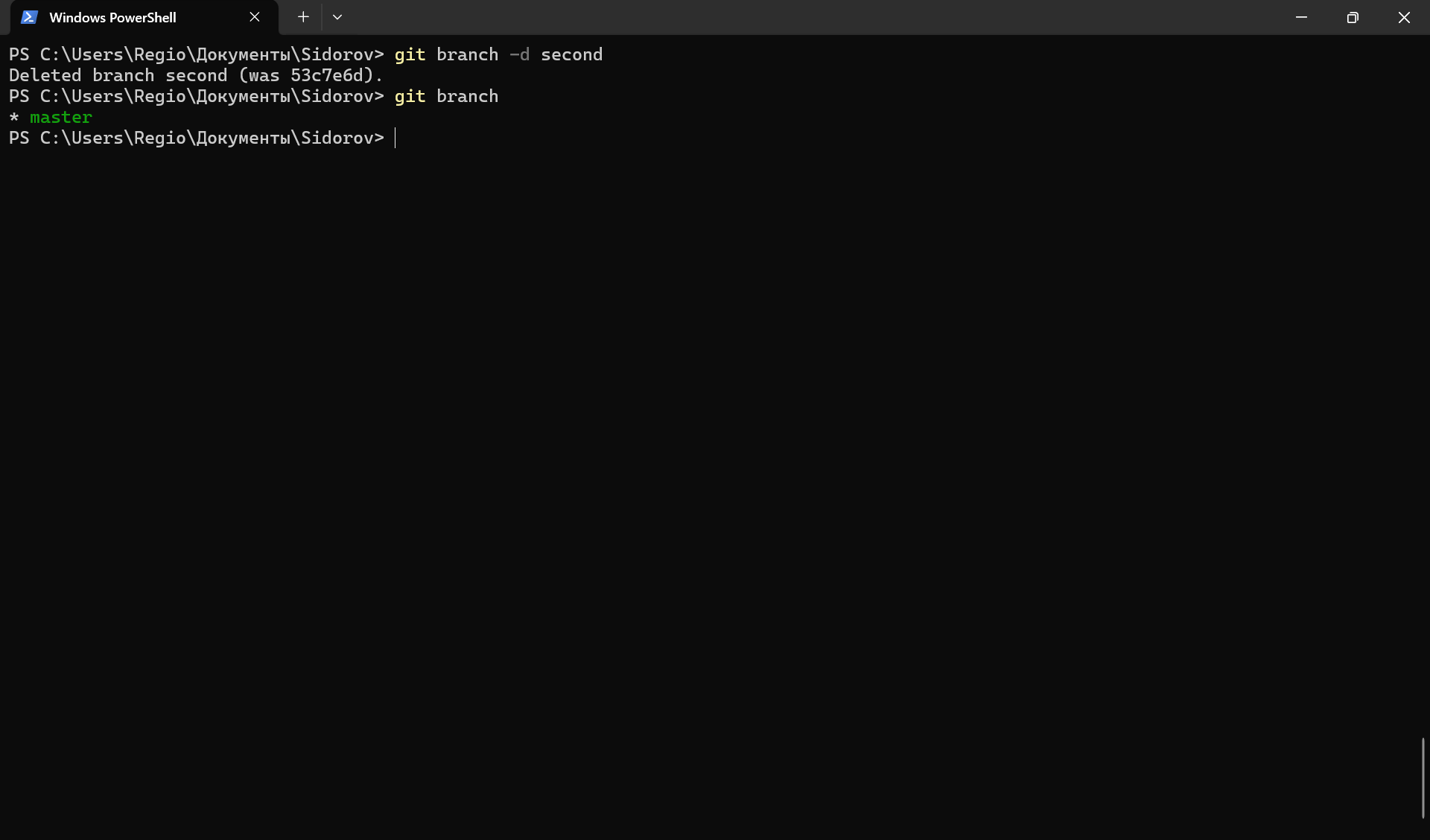
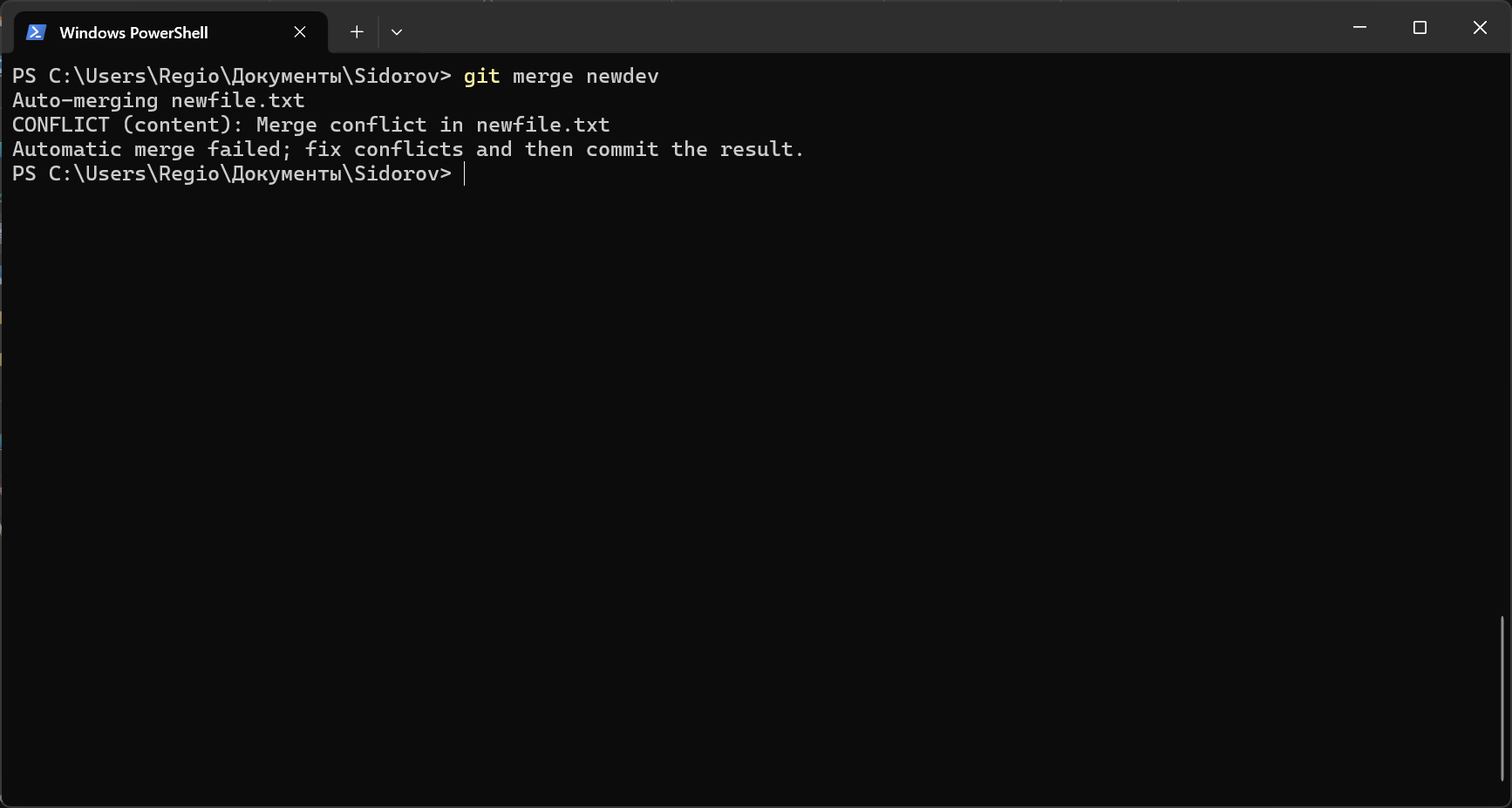
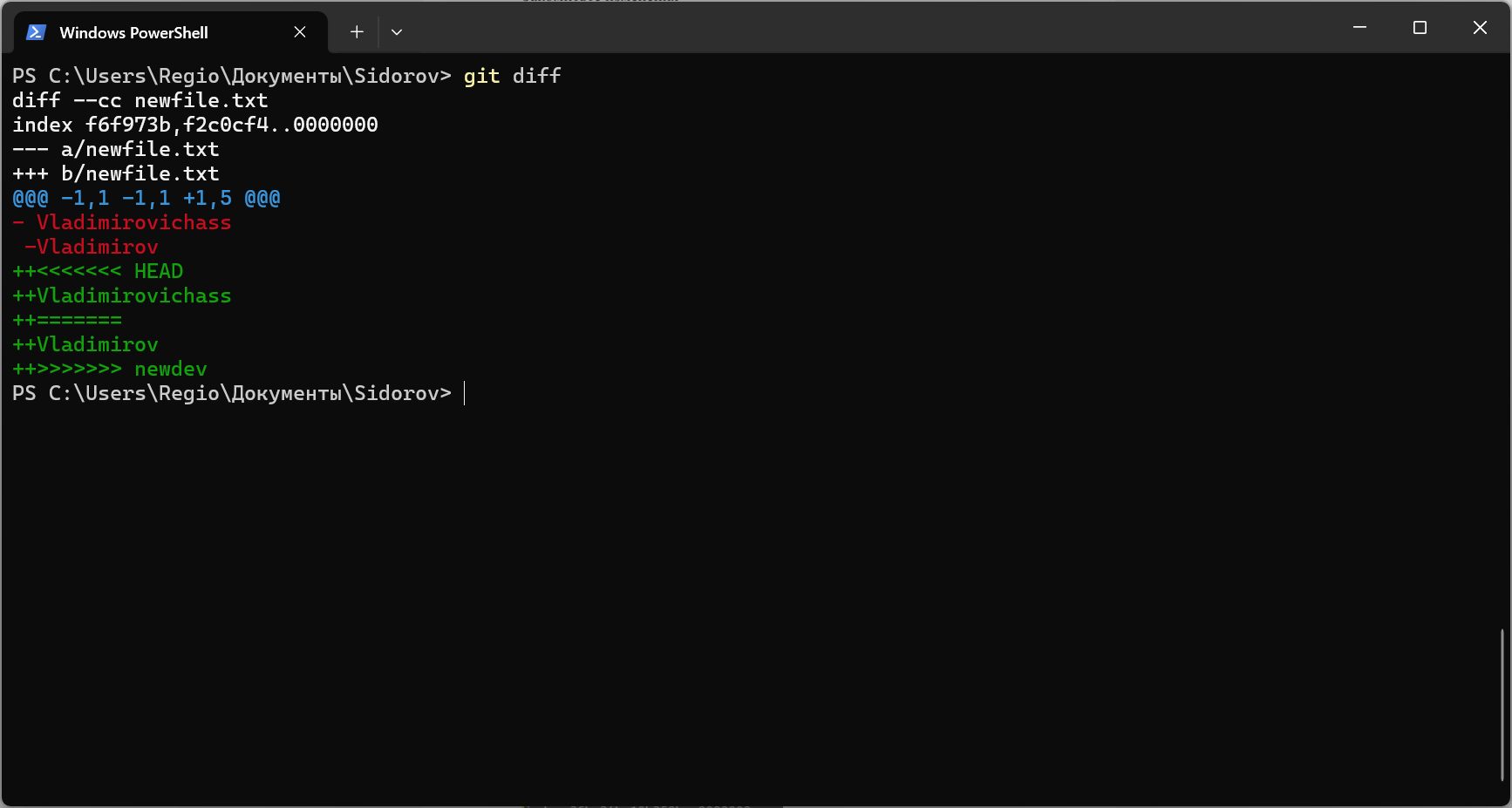


Рис. 12 - Удаление secon ветки

Конфликт с веткой newdev и его разрешение:

<<<<<<< HEAD  
Ivanovichii  
=======  
Ivano  
>>>>>>> newdev





## 2.8 Откаты изменений

git log  
git checkout hash  
git checkout master  
  
git reset HEAD

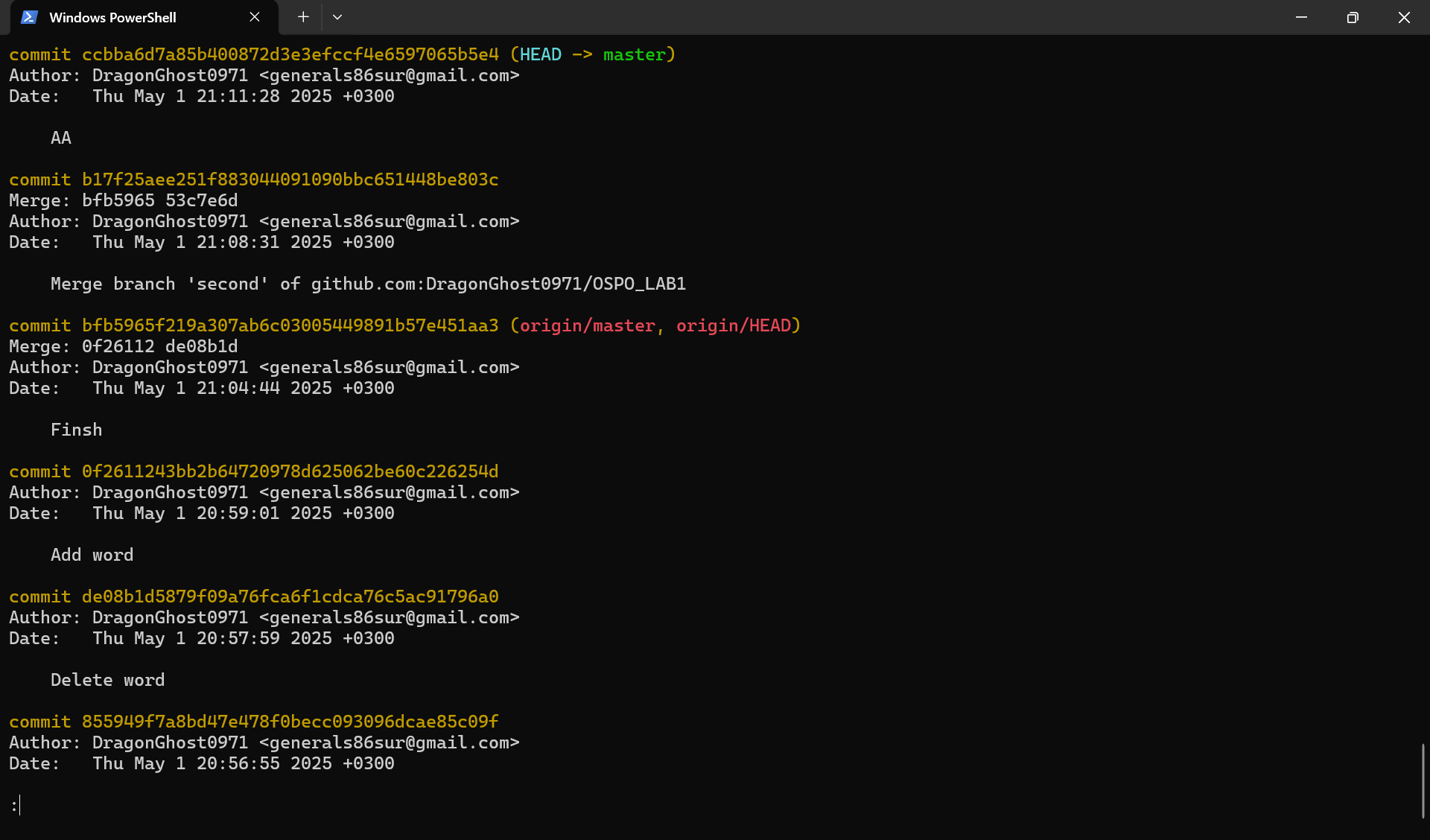


Рис. 13 - Просмотр коммитов

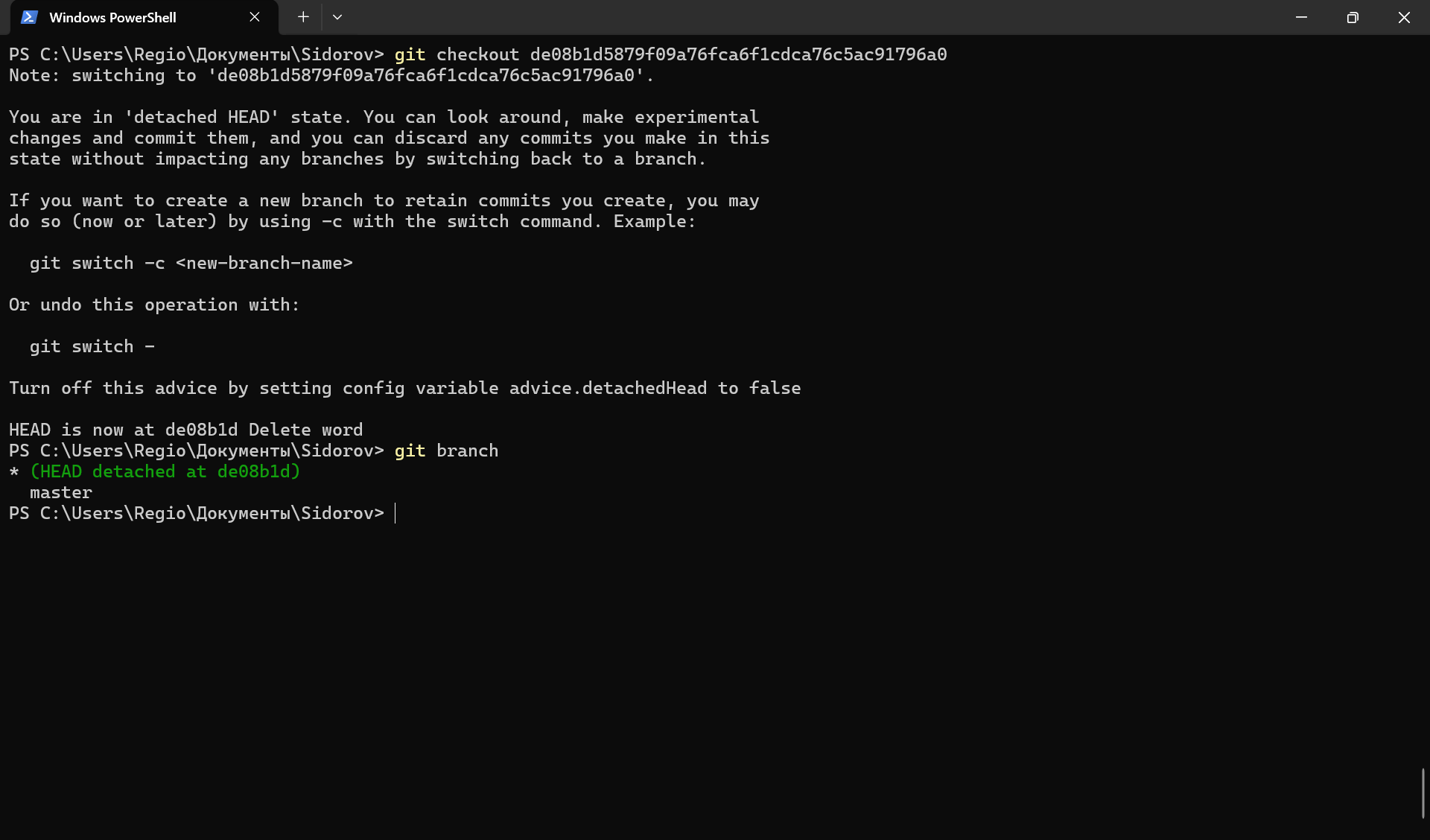


Рис. 14 - Перейти к коммиту по hash

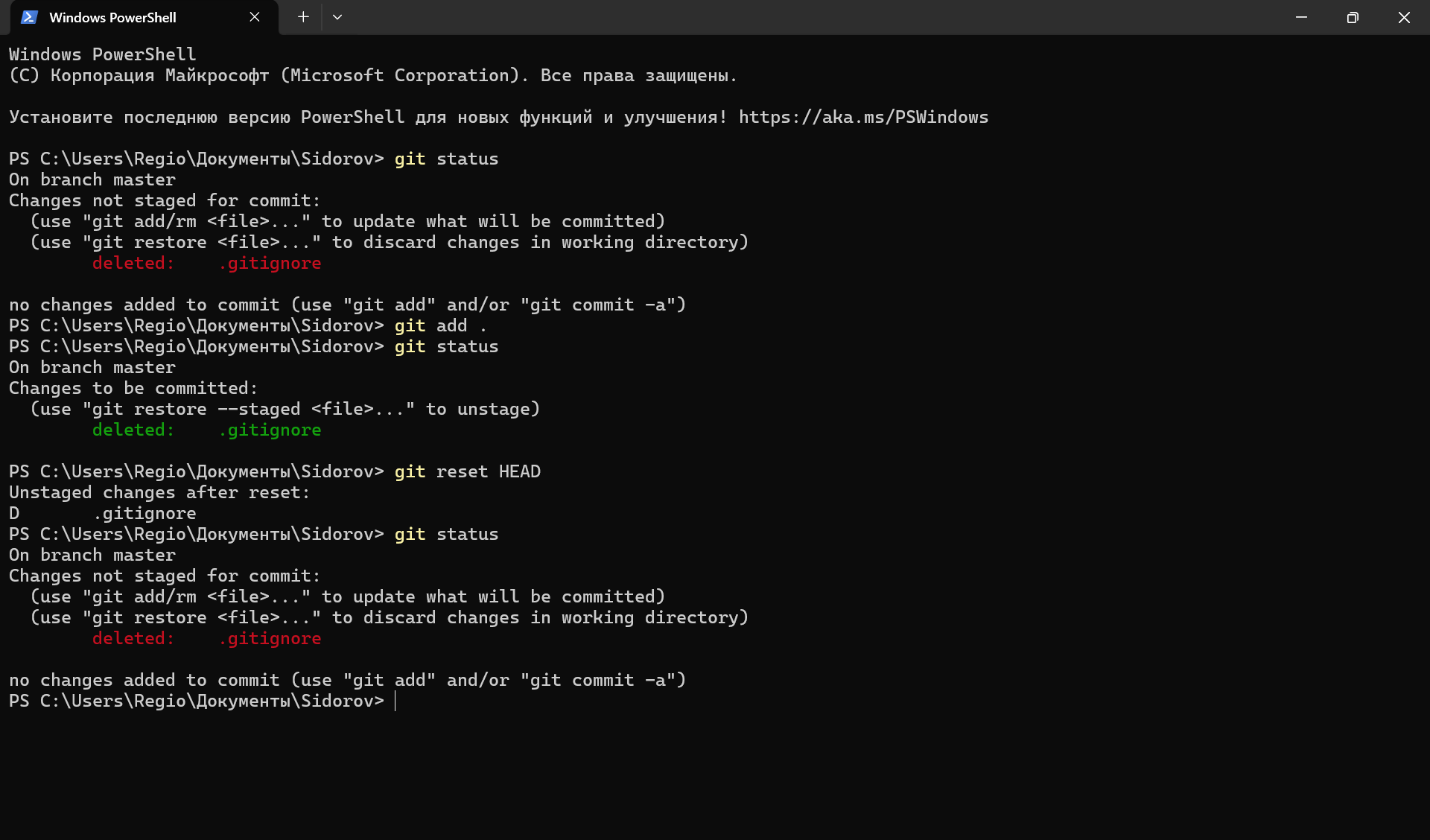


Рис. 15 - Снятие файлов с закрепления

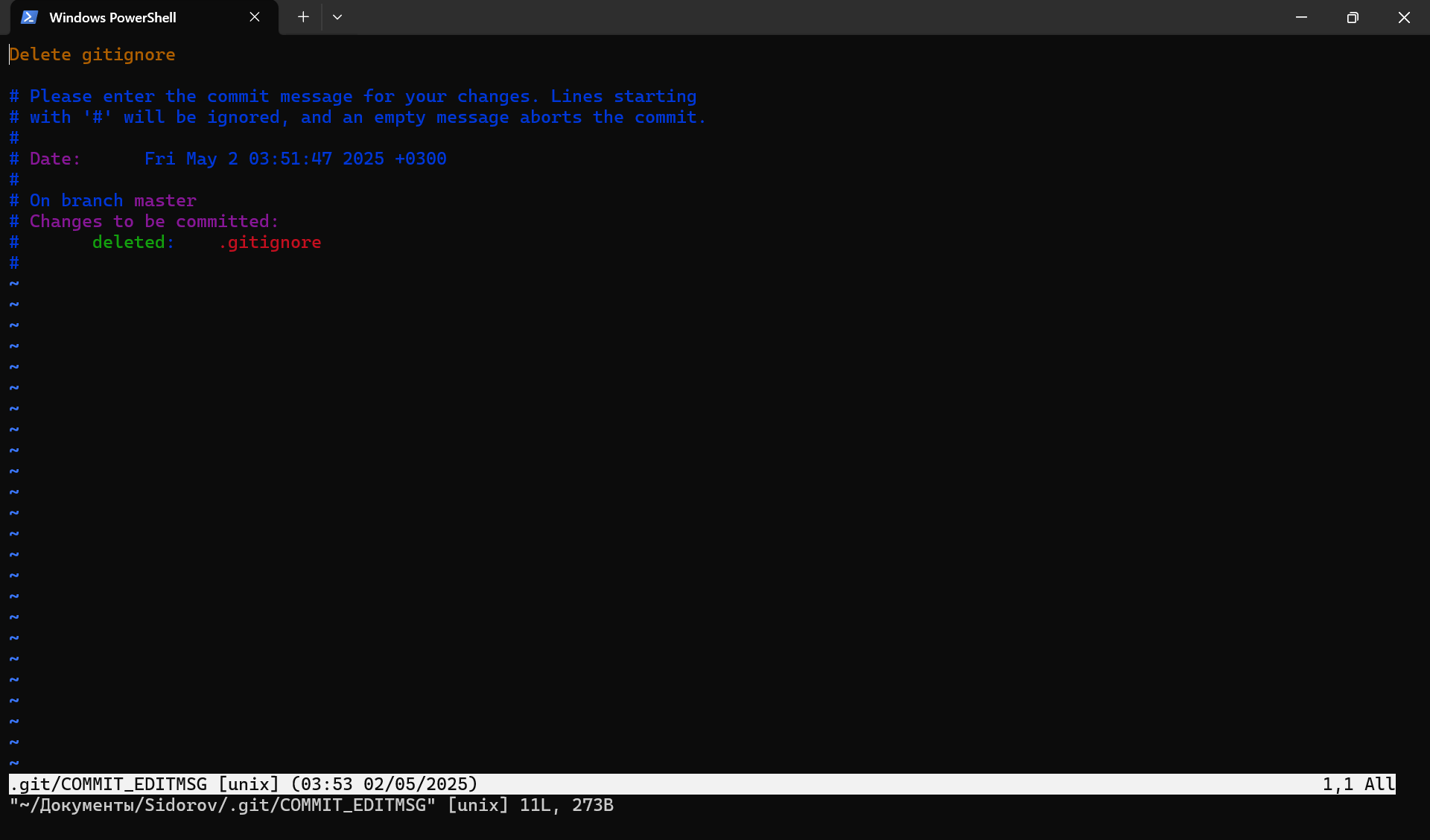


Рис. 16 - Git commit --amend

## 2.9 Игнорирование файлов

Создаем файл “.gitignore” в корневой директории и записываем в него файлы которые необходимо игнорировать

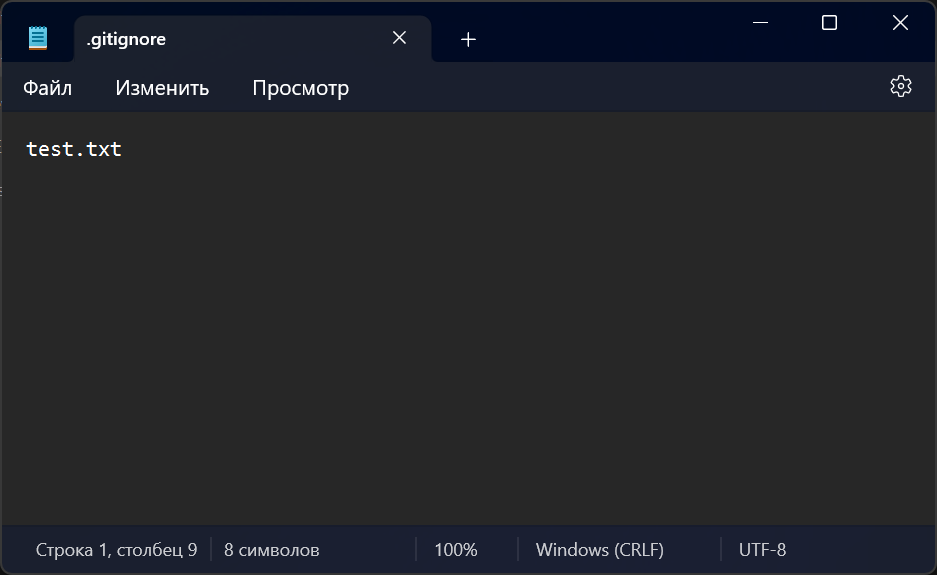


Рис. 17 - Файлы которые необходимо игнорировать

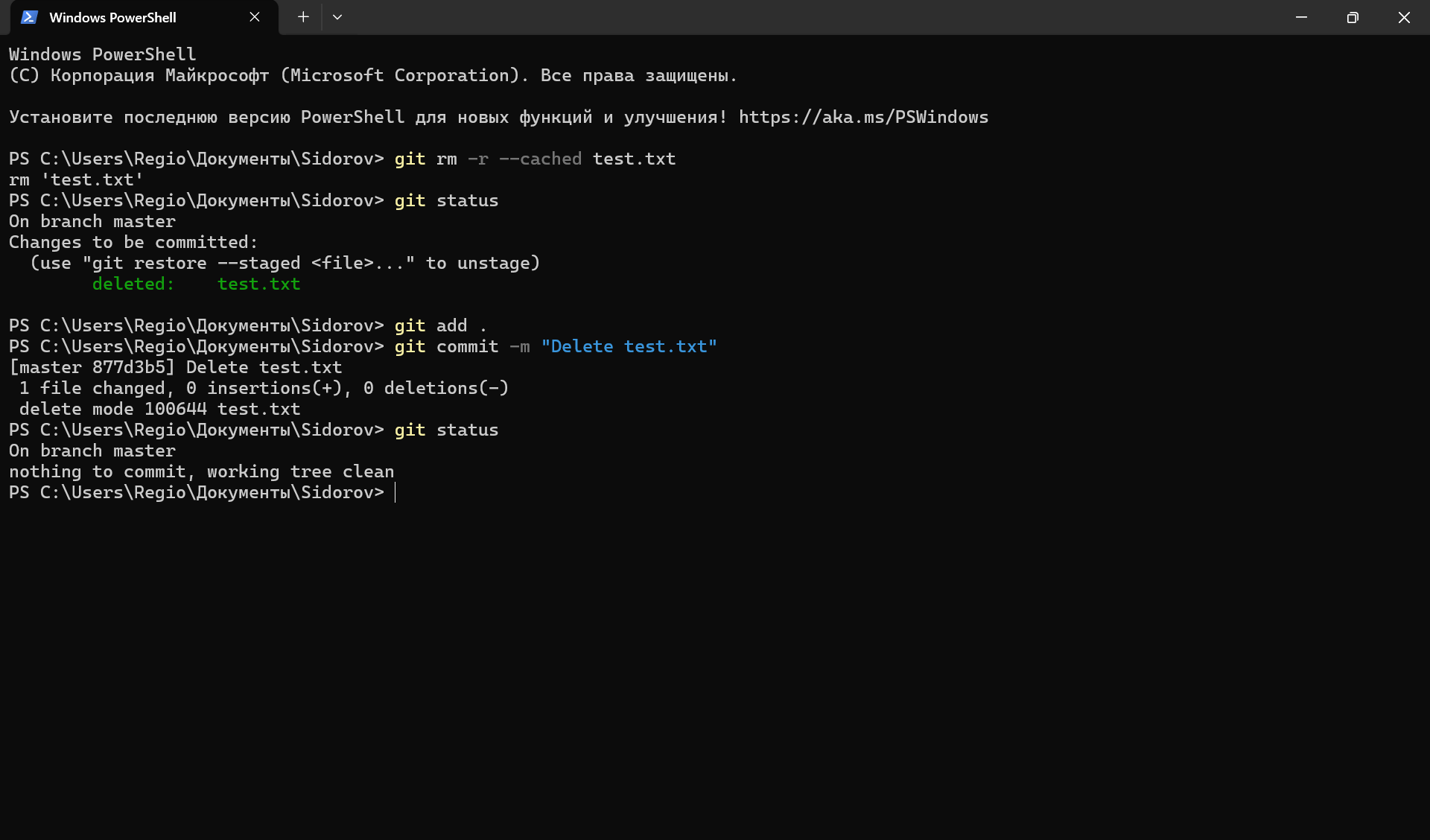


Рис. 18 - Предварительно нужно удалить этот файл из отслеживаемых

git add .   
git commit -m "add .gitignore"  
git push

## 2.10 Совместная работа (Fork → Clone → Pull Request)

1. Напарник сделал Fork моего репозитория.

2. Я выдал ему права Write в Settings → Collaborators.

3. Каждый из нас выполнил git clone своего URL и внёс изменения.

4. После git push напарник оформил Pull Request; я провёл ревью и выполнил Merge.

# 4. Результаты

* Создано и синхронизировано два репозитория (локальный + GitHub).
* Освоены базовые и продвинутые команды Git (init, add, commit, push, pull, branch, merge, revert, reset, checkout).
* Отработаны типичные случаи конфликтов и их ручное разрешение.
* Организована совместная разработка с использованием fork, clone и pull request.
* Реализовано исключение лишних файлов через .gitignore.

# 5. Выводы

В ходе лабораторной работы были получены практические навыки полного жизненного цикла проекта под управлением Git. Освоены как индивидуальные операции (фиксация, откат, ветвление), так и элементы командной разработки (fork, merge-conflict resolution).