МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное   
образовательное учреждение высшего образования  
«Самарский национальный исследовательский университет   
имени академика С.П. Королева»

(Самарский университет)

Институт информатики и кибернетики

Кафедра ПИАШ

**Отчет по лабораторной работе № 1**

Дисциплина: «Моделирование информационных систем»

Тема: **«Работа с Git и GitHub»**

Выполнили: Башкатов С.А., Казанцев А.Б.

Группа: 3103-090401D

Самара 2025

**ЗАДАНИЕ**

1. Исходные данные

Вариант задания (предоставляется преподавателем).

2. Общий план выполнения работы

1. Создание удалённого репозитория на GitHub

2. Клонирование репозитория

3. Работа с ветками (Branching)

4. Работа с Pull Requests (PR)

5. Работа с Issues

6. Работа с Projects

7. Работа с GitHub Wiki

8. Разрешение конфликтов при слиянии (Merge Conflicts)

9. Создание релизов и тегов

3. Содержание отчета

Отчет по работе может быть оформлен в произвольном виде, и должен содержать элементы внешнего описания.

1. Цель работы

2. Для чего нужен Git?

3. Ход выполнения работы

**ХОД РАБОТЫ**

**1. Цель работы:** Установка и работа в системе контроля версий Git Bash, установка и регистрация на GitHub. Создание репозитория, клонирование. Команды GitНub’а. Cовместная работа, Pull Requests, Issues, Projects и дополнительные возможности GitHub.

**2. Для чего нужен Git?**

Git – это система контроля версий (СКВ). Существует несколько подобных систем, однако Git – на настоящий момент наиболее используемая СКВ.

Git – набор скриптов, который умеет управлять изменениями. Он следит за файлами, ведет их историю, умеет ими манипулировать, откатывать, сливать и т.д.

Задача Git – вести полную историю изменений в некоей папке на сервере или локальном компьютере (**репозиторий**). К изменениям относятся добавление и удаление файлов, модификация их содержимого (нужно также учитывать, что Git не следит за пустыми папками, в папке нужно создать хотя бы один файл).

В историю также записывается, кто и когда сделал изменения.

Git – это распределенная система контроля версий, в ней нет центрального репозитория. Репозиторий – это просто папка с вашими файлами, в которой есть ещё некая служебная информация для Gitа. Их может быть очень много. Репозитории могут обмениваться изменениями между собой.

**3. Ход выполнения работы.**

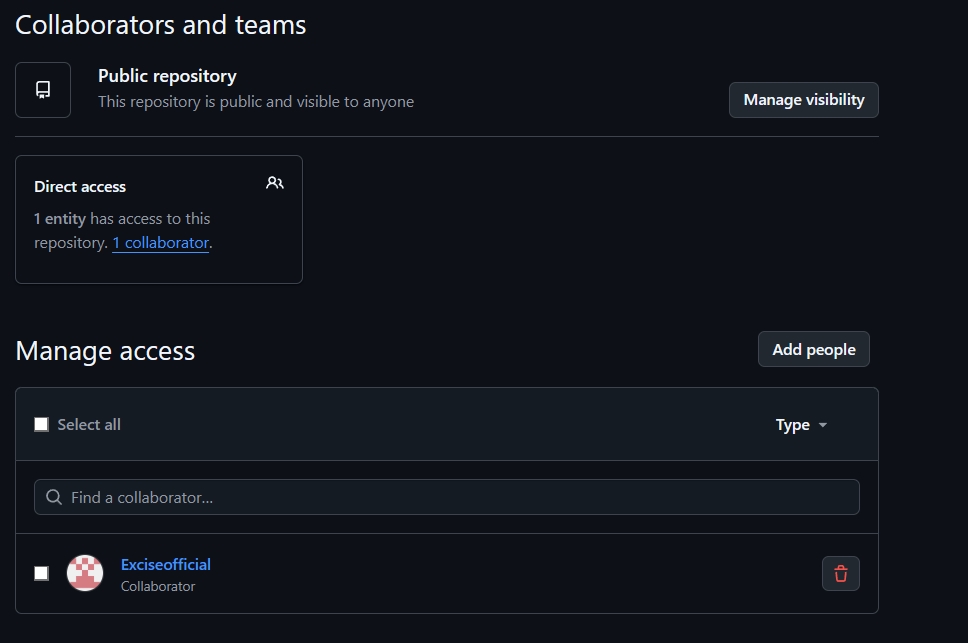
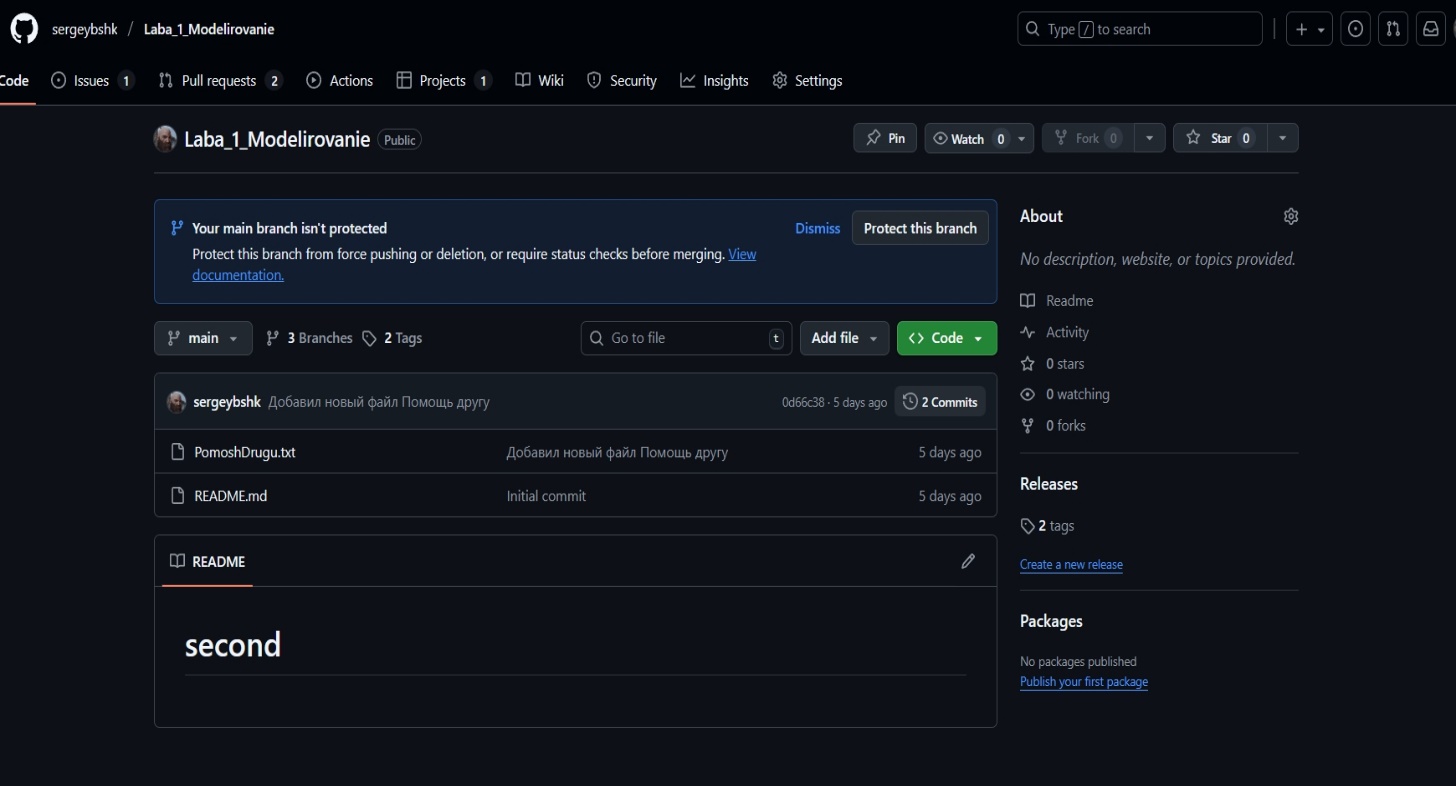
**Шаг 1.** Создание удалённого репозитория на GitHub

Рисунок 2 - Участники проекта

Рисунок 1 - Создание удаленного репозитория

**Шаг 2.** Клонирование репозитория

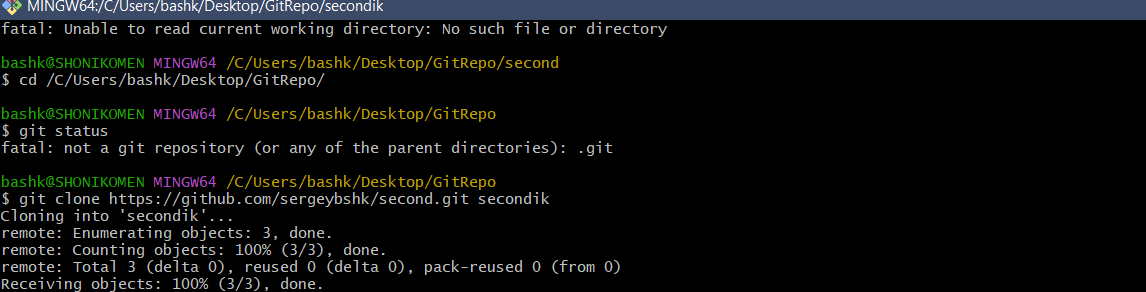
****

Рисунок 3 - Клонирование репозитория на локальный диск

**Шаг 3.** Работа с ветками

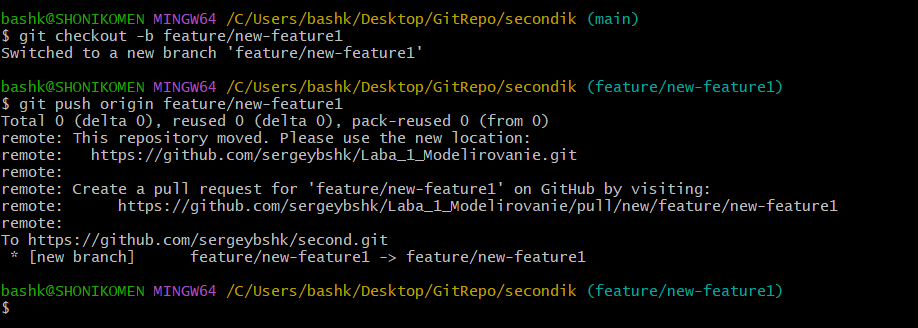
****

Рисунок 5 - Ветки созданы

Рисунок 4 - Создание ветки и отправка ее в удаленный репозиторий

**Шаг 4.** Работа с Pull Requests (PR)

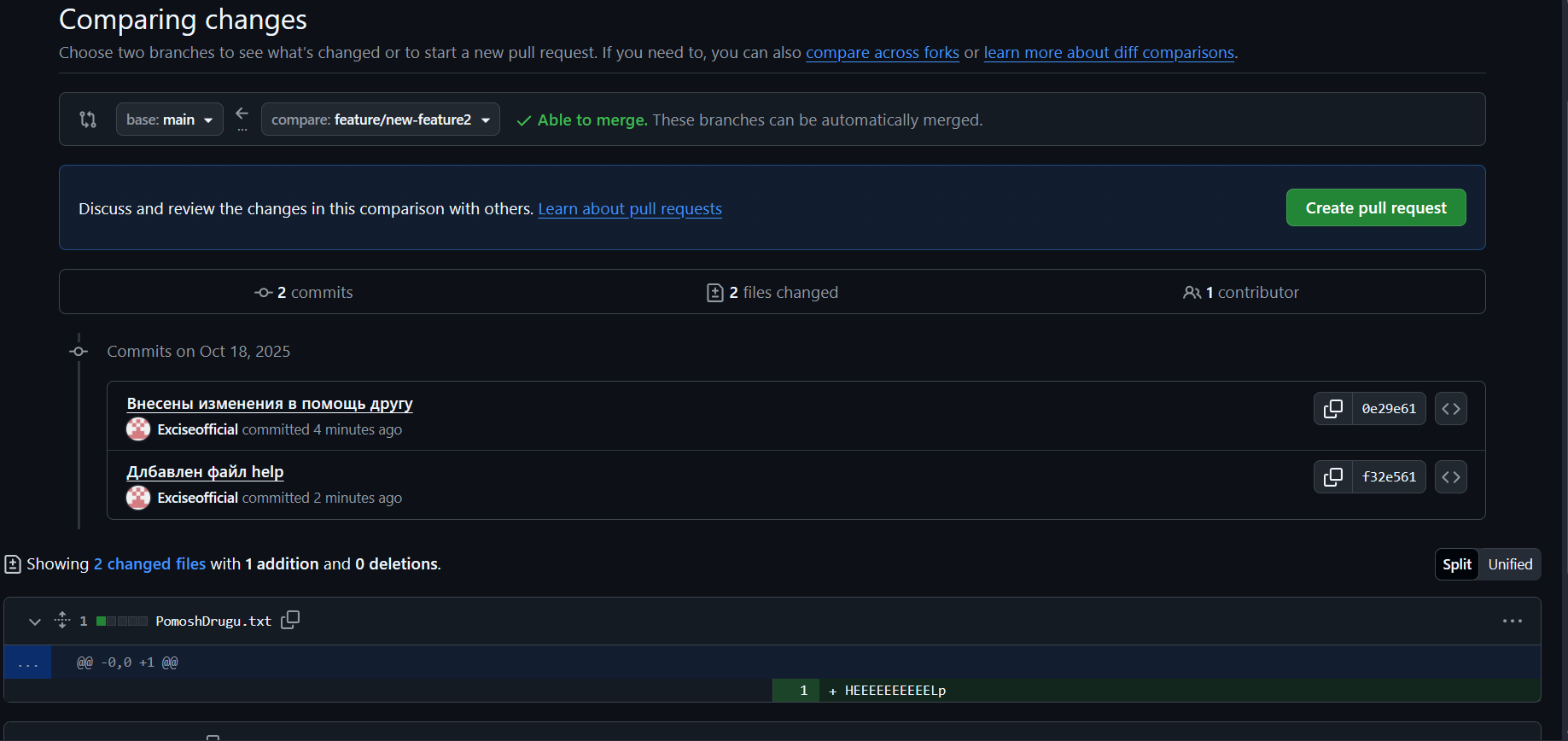
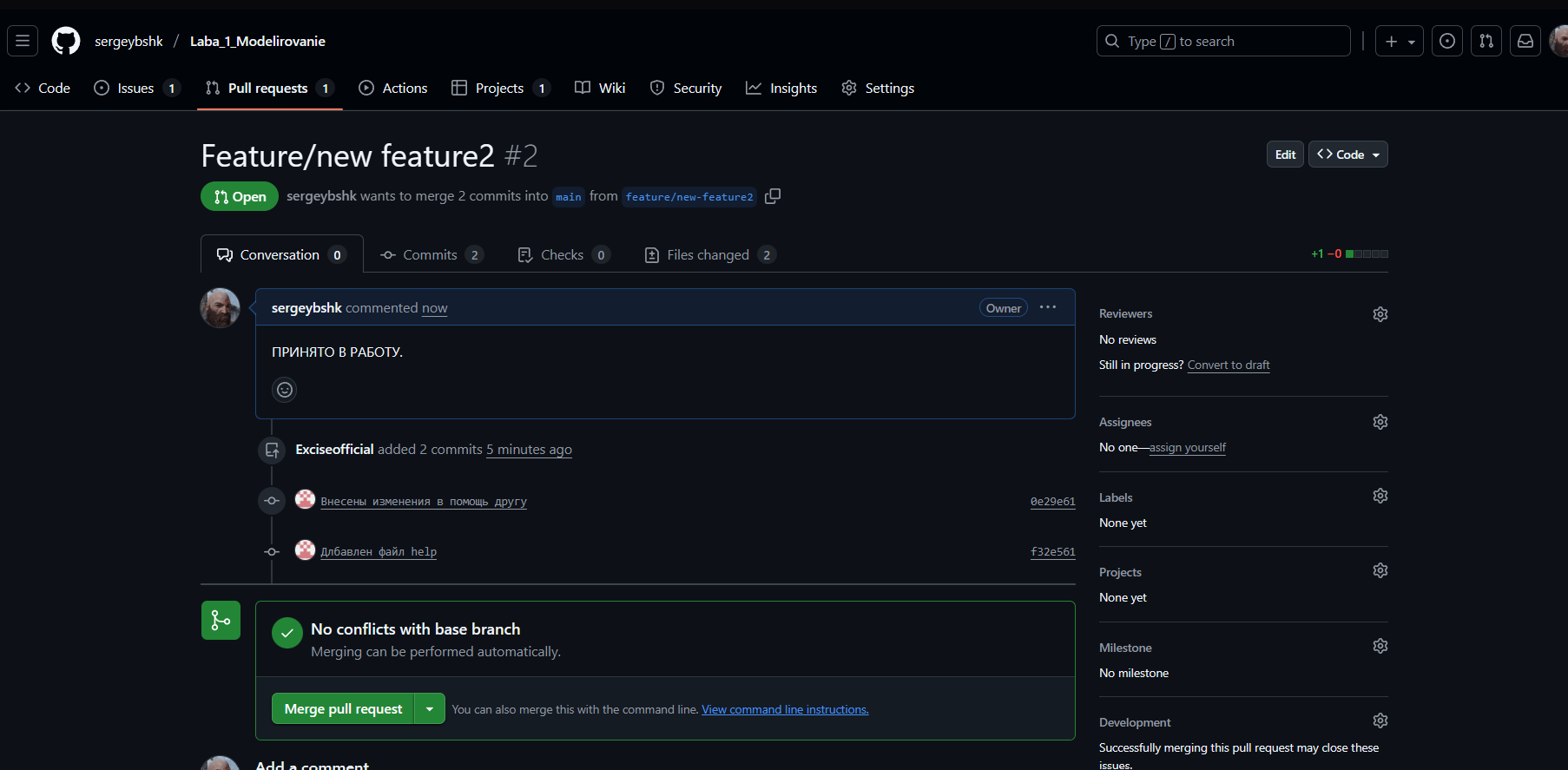
****

Рисунок 7 - PR создан

Рисунок 6 - Работа с PR

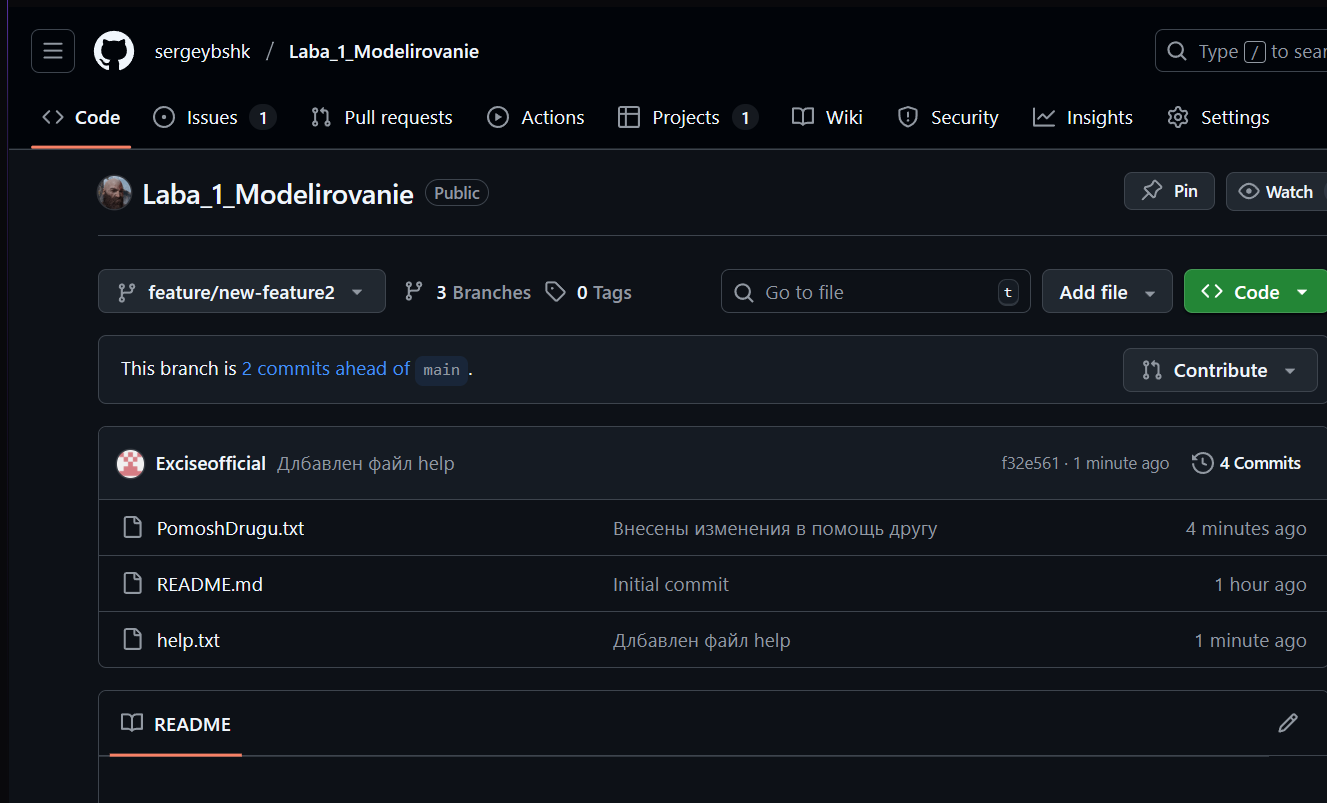
****

Рисунок 8 - В проект добавлены изменения

**Шаг 5.** Работа с Issues

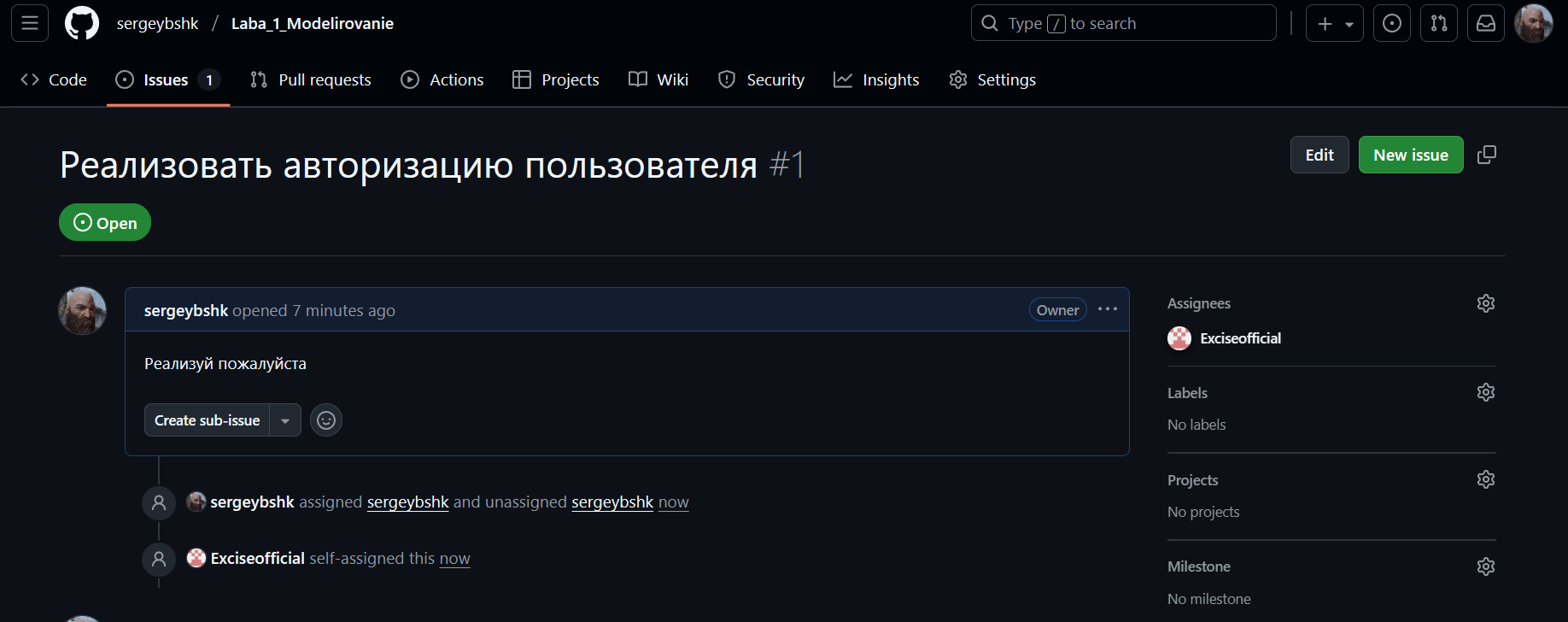


Рисунок 9 – Выполнение шага 5

**Шаг 6.** Работа с Projects

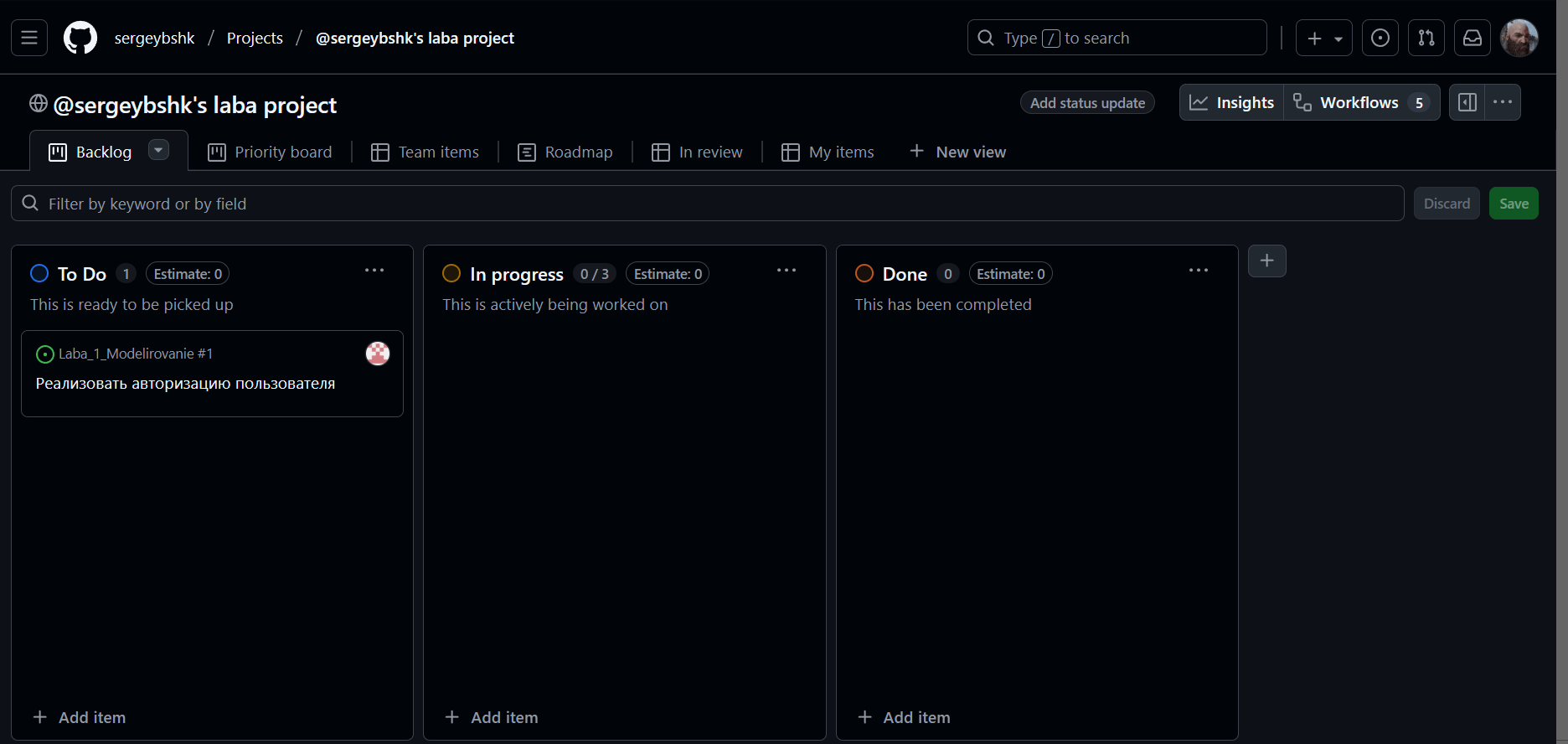
****

Рисунок 10 - Выполнение шага 6

**Шаг 7.** Работа с GitHub Wiki

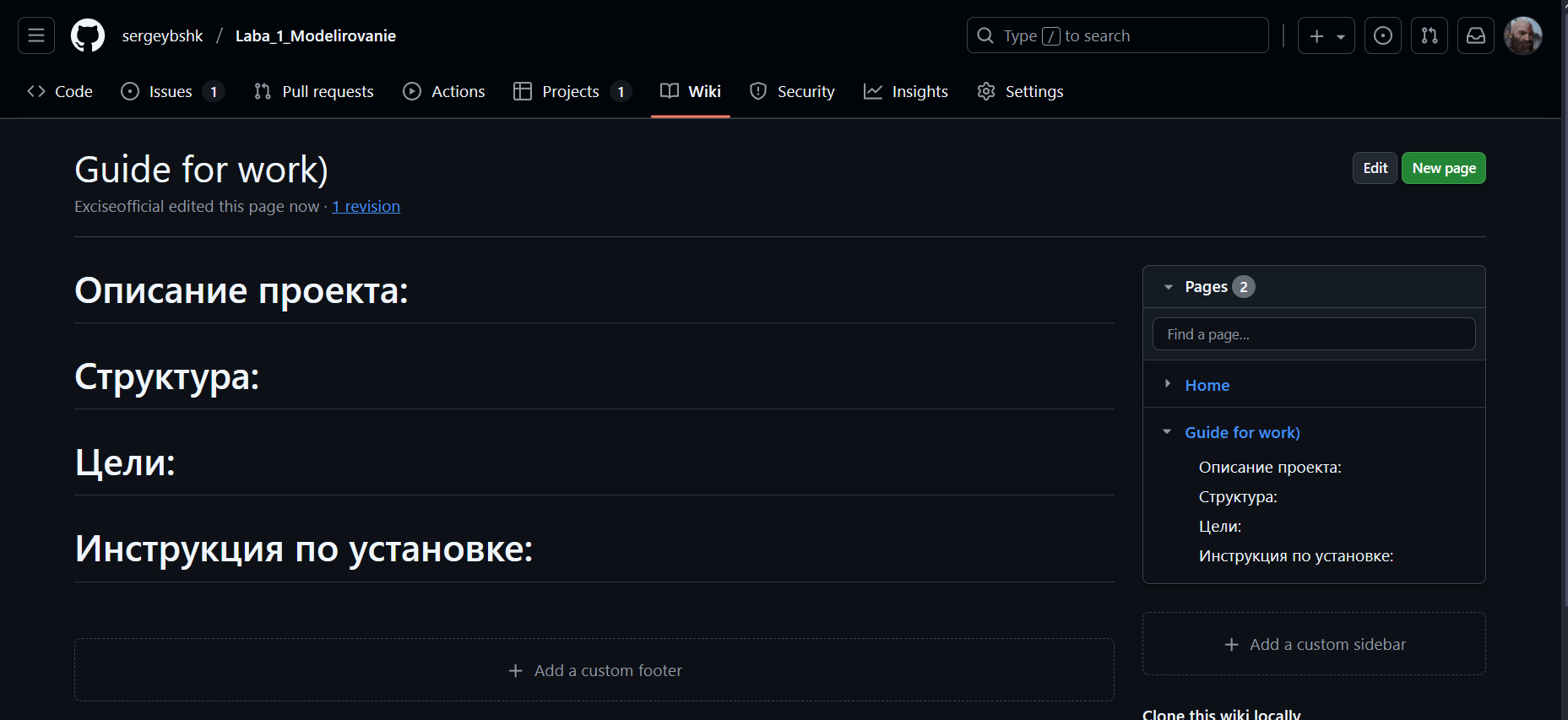
****

Рисунок 11 - Выполнение шага 7

**Шаг 8.** Разрешение конфликтов при слиянии (Merge Conflicts)

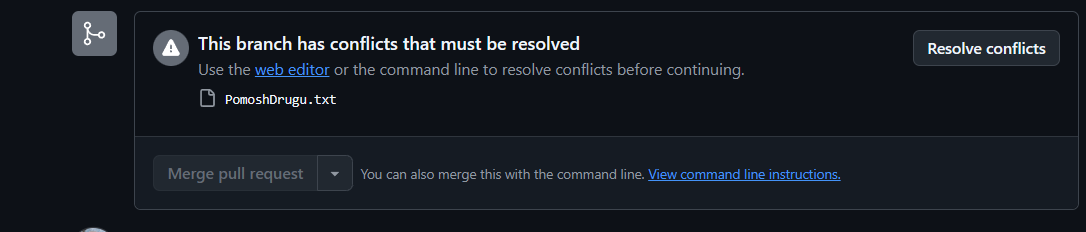
****

Рисунок 12 - Возник конфликт при слиянии

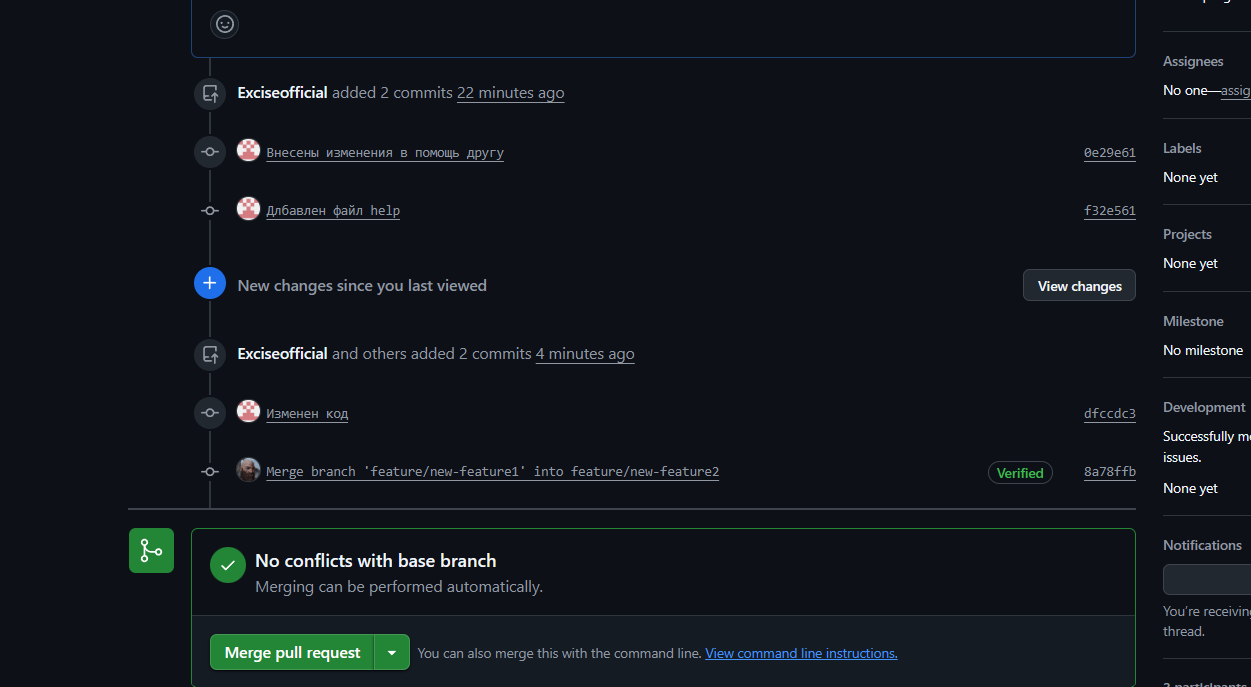
****

Рисунок 13 - Разрешение конфликта при слиянии

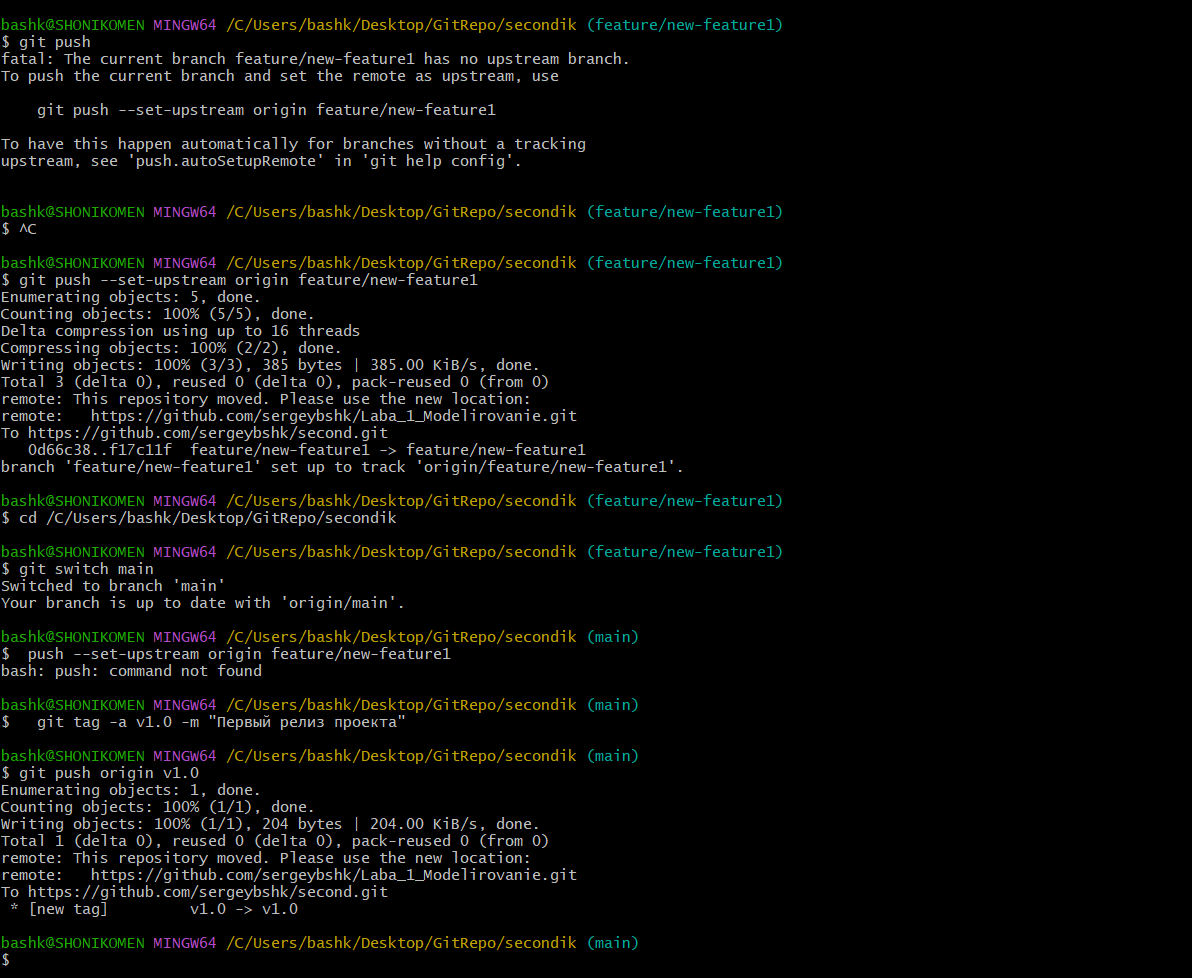
**Шаг 9.** Создание релизов и тегов

Рисунок 14 - Выполнение шага 9

**ЗАКЛЮЧЕНИЕ**

В результате выполнения лабораторной работы выполнено: установка и работа в системе контроля версий Git Bash, установка и регистрация на GitHub, создание репозитория, клонирование. Также изучены возможности GitНub.