МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГТУ», ВГТУ)

Факультет информационных технологий и компьютерной безопасности

Кафедра компьютерных интеллектуальных технологий проектирования

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №5

По дисциплине: «Системы хранения и обработки данных»

Тема: «Интеграция изменений в системе управления версиями Git»

Выполнил работу студент группы змиИВТ-241: Москвитина А.Г.

подпись, дата

Принял: Короленко В.В.

подпись, дата

Воронеж 2024

Цель работы: изучить процесс интеграции изменений в системе управления версиями с помощью операции Pull Request (Merge Request).

Основные задачи:

перенос (интеграция) изменений в главную ветку с помощью операции pull request (merge request);

рецензирование проекта коллегами и рецензирование проектов коллег.

Ход работы:

В начале мы добавим двух ревьюверов, которые могли бы оценить проект. Для этого надо добавить соответствующих пользователей в список соучастников проекта (рис. 1 и 2).

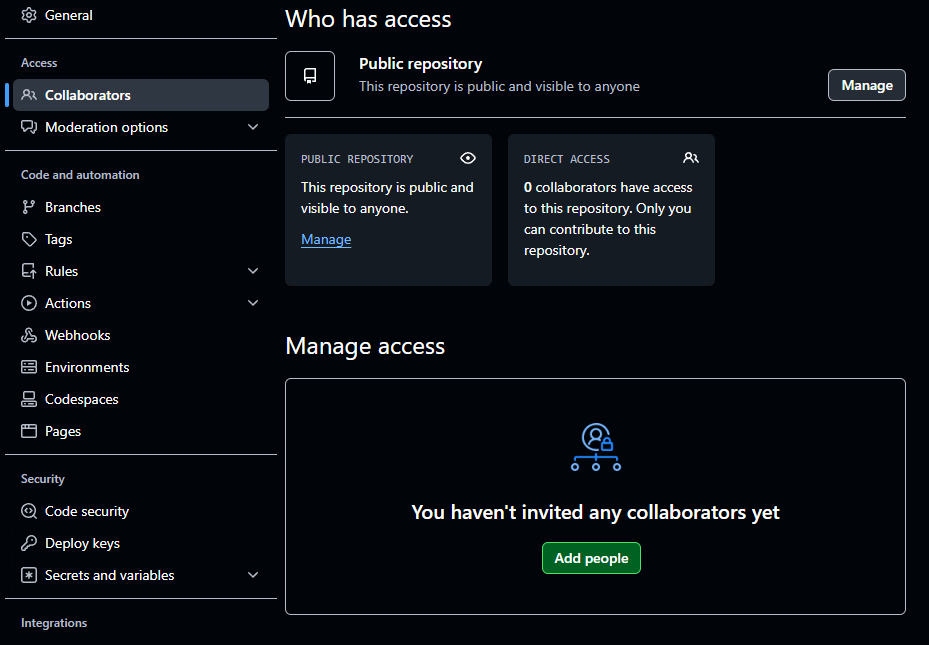


Рисунок 1 – Страница добавления ревьюверов

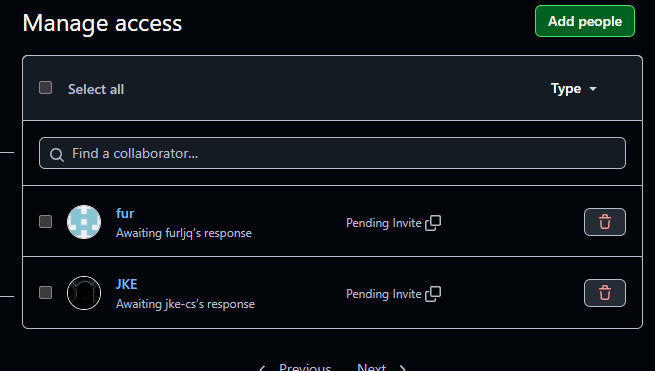


Рисунок 2 – Приглашение ревьюверов

Переходим на страницу проекта и нажимаем «Compare and pull request» (рис. 3 и 4).

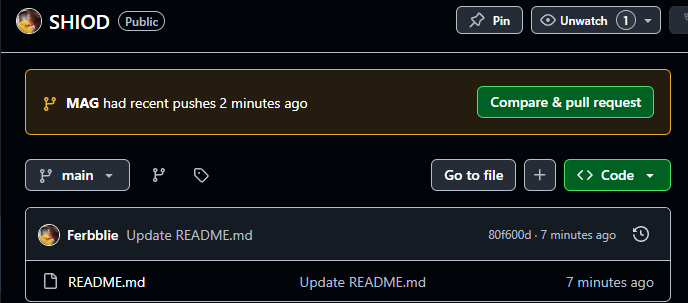


Рисунок 3 - Compare and pull request

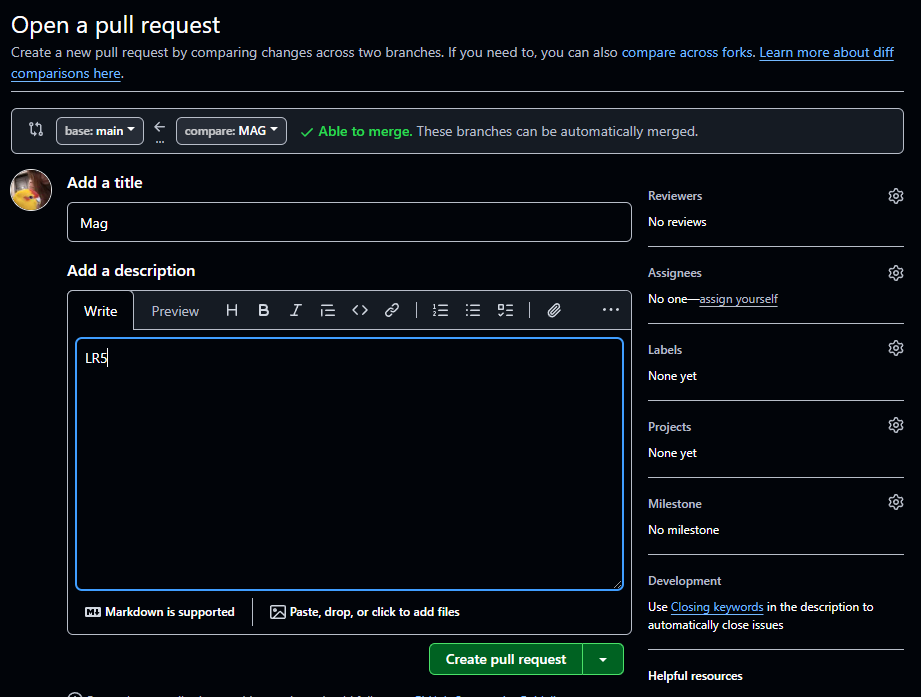


Рисунок 4 – Страница Pull Request

Добавляем в проект наших двух ревьюверов. После того, как ревьюверы оценят изменения и одобрят Pull Request, мы можем провести слияние. На рисунке 6 показано меню после одобрения ревьюверов.

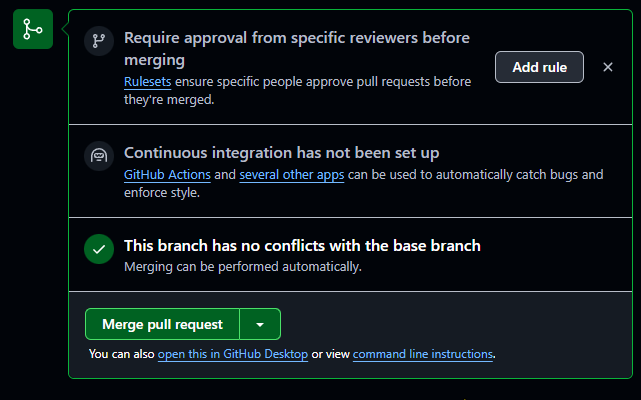


Рисунок 5 – Pull Request после оценки

Далее мы нажимаем на кнопку Merge pull reques тем самым подтверждаем слияние веток и завершаем Pull Request. Результат показан на рисунках 6 и 7.



Рисунок 6 – Merge pull reques

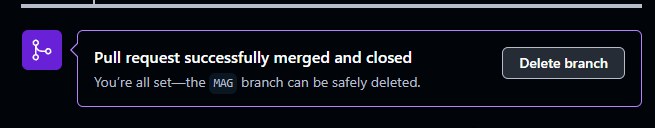


Рисунок 7 – Уведомление о завершении Pull Request

Проведем сравнительный анализ процесса работы с системой Git на разных операционных системах.

Работа с системой Git на различных операционных системах предоставляет уникальные опыты для пользователей, и эти различия в основном касаются средств взаимодействия и визуализации. На операционной системе Windows часто предпочитают использовать графические интерфейсы, такие как Git Bash или SourceTree, что делает работу с репозиторием более интуитивной. Это особенно удобно для тех, кто не привык работать в терминале. Однако иногда при использовании командной строки на Windows могут возникнуть проблемы с совместимостью символов.

В среде Linux и macOS Git интегрирован более естественным образом, что облегчает работу через командную строку и поддерживает более гибкое управление репозиториями. Пользователи этих операционных систем обычно предпочитают работать в терминале, где у них есть более полный контроль над командами и возможностью автоматизации различных процессов.

Несмотря на различия в интерфейсах и интеграции, важно отметить, что основные концепции работы с Git остаются неизменными на всех платформах. Все операционные системы обеспечивают возможность отслеживания изменений, ветвления, слияния и другие основные функции контроля версий.

Таким образом, можно заключить, что Git предоставляет универсальные инструменты для контроля версий, а выбор между графическими интерфейсами и командной строкой зависит от предпочтений пользователя и специфики проекта. Эффективное использование Git требует от пользователя понимания основных концепций системы, вне зависимости от операционной системы, на которой он работает.

Вывод: в данной лабораторной работе виртуальные машины были подготовлены к установке программного обеспечения, предназначенного для анализа данных.

Контрольные вопросы

1. Как произвести откат до предыдущего коммита?

Для выполнения отката до предыдущего коммита используется команда "git reset" с опцией "--hard" и указанием хэша коммита или с использованием относительных ссылок, таких как "HEAD^" для предыдущего коммита.

1. Какой репозиторий называется публичным?

Публичным репозиторием называется репозиторий в системе контроля версий, к которому имеют доступ все пользователи без необходимости авторизации.

1. Опишите операцию Pull Request.

Операция Pull Request позволяет разработчикам предложить изменения из своей ветки в другую ветку и запросить их интеграцию. Этот процесс создает пространство для обсуждения предложенных изменений и обеспечивает прозрачность внесенных модификаций. Участники проекта могут рецензировать, комментировать и обсуждать код, а также предоставлять обратную связь. Разработчик создает Pull Request, указывая исходную ветку (откуда будут взяты изменения) и целевую ветку (куда они будут интегрированы). При создании PR часто прикладывается описание, поясняющее цель изменений и внесенные модификации. Члены команды и другие участники проекта могут рецензировать код, оставлять комментарии и выражать свое мнение относительно предложенных изменений. Обсуждение и внесение комментариев происходят в интерфейсе системы контроля версий. После обсуждения и, при необходимости, внесения корректив, ответственный за мердж (слияние) производит интеграцию изменений в целевую ветку. При успешном слиянии изменения становятся частью основного кодовой базы проекта. Pull Request позволяет управлять исходными изменениями, обеспечивать качество кода в процессе разработки программного обеспечения.