МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт математики и информационных систем

Факультет автоматики и вычислительной техники

Кафедра систем автоматизации управления

**Система контроля версий Git**

Отчет по лабораторной работе №2  
по дисциплине

«Управление IT-проектами»

Выполнил студент гр. ИТб-4302-02-00 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Вершинин П.А./

(Подпись)

Руководитель ст. преподаватель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Родионов К.В./

(Подпись)

Работа защищена с оценкой «\_\_\_\_\_\_\_\_\_\_\_» «\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2023 г.

Тема: «Система контроля версий Git».

Цель: научиться управлять версиями проекта с помощью системы Git; освоить работу с репозиториями GitHub.

В ходе работы необходимо выполнить следующее задание:

* создать git-репозиторий;
* определить состояние с помощью команды «git status»;
* подготовить файлы с помощью команды «git add»;
* зафиксировать изменения командой «git commit»;
* откатить изменения командой «git reset»;
* создать файл «gitignore» для игнорирования файлов или директорий в git-репозитории;
* создать удаленный git-репозиторий;
* отправить данные в удаленный git-репозиторий;
* принять данные из удаленного git-репозитория;
* клонировать репозиторий командой «git clone»;
* создать, сделать слияние и удалить git-ветку командой «git branch».

Для инициализации репозитория в консоли необходимо ввести команду git init, предварительно выбрав директорию для работы (рис. 1).

Создадим директорию, откроем git-консоль и перейдем в рабочую директорию.

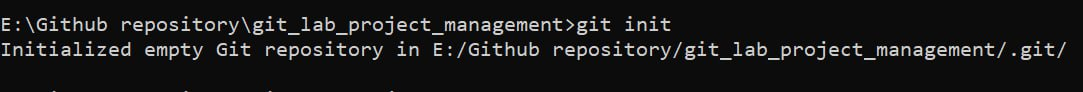


Рисунок 1 – Инициализация git-репозитория

«Initialized empty Git repository in [путь]» означает, что git-репозиторий был успешно инициализирован и в корне проекта была создана скрытая папка с

названием «.git», в которой хранятся git-настройки и прочие служебные файлы.

Создадим обычный текстовый файл «file\_for\_save.txt» с любым содержимым.

Вводим в git-консоль команду: «git status».

Результат выполнения команды представлен на рисунке 2.

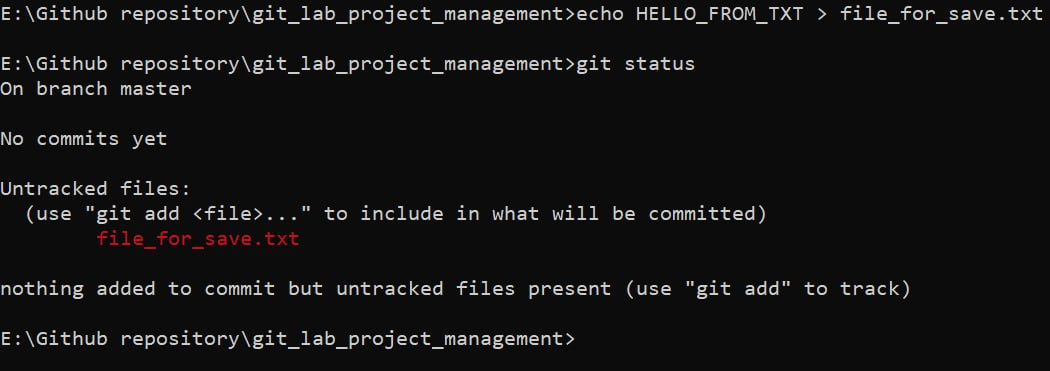


Рисунок 2 – Просмотр состояния «git status»

Необходимо добавить файл и произвести «git commit» для фиксации изменений и указания сообщения.

Для добавления конкретного файла к текущей фиксации изменения вводим команду: «git add [файл]». Добавление всех файлов можно использовать команды: «git add .», «git add -A».

Результат выполнения команды представлен на рисунке 3.

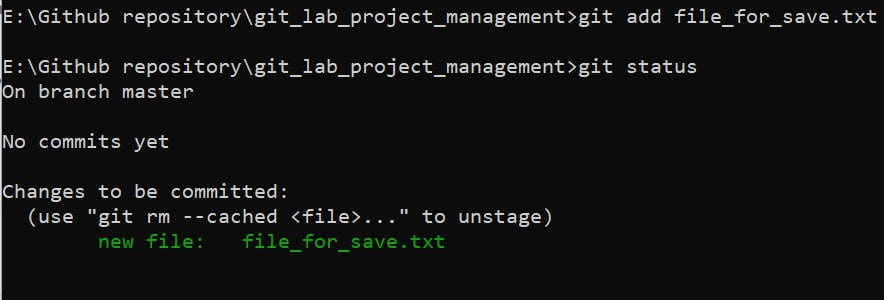


Рисунок 3 – Подготовка файлов «git add»

Фиксация изменений (git-коммит) в Git необходимы для отслеживания изменений. С указанием ключа «-m “текст изменений”» в конкретном изменении позволяет предоставить краткое описание фиксации изменения, что предоставляет понимание какие изменения были внесены.

Для фиксации изменения воспользуемся командой: «git commit -m “[сообщение]”». Для просмотра истории git-коммитов достаточно ввести команду: «git log».

Результат выполнения команд представлен на рисунке 4.

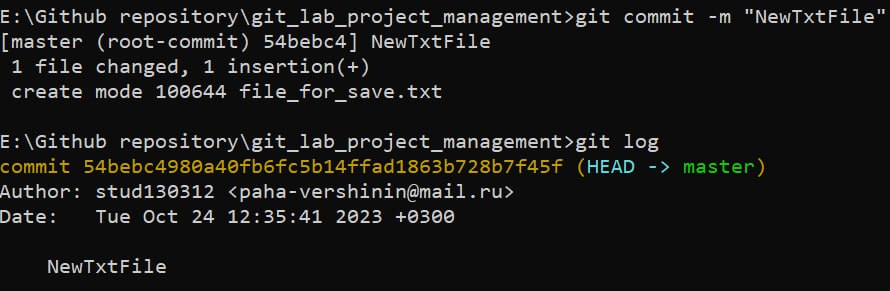


Рисунок 4 – Фиксация изменений «git commit»

В данном ответе содержится идентификатор состояния, автор, дата создания git-коммита.

Для отката изменения создадим еще один текстовый файл, добавим и зафиксируем изменения. Введем команду для отката изменения: «git reset [идентификатор изменения]». К данной команде существует несколько ключей, рассмотрим каждый из них:

«--soft» мягкий откат к конкретному git-коммиту, удаление состояния не происходит;

«--hard» жесткий откат к конкретному git-коммиту, удаление происходит.

Результат выполнения команды представлен на рисунке 5.

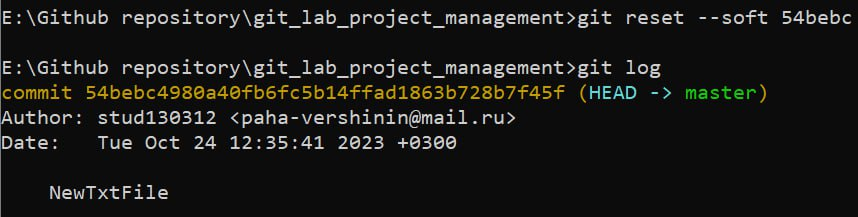


Рисунок 5 – Откат изменения «git reset»

Создадим новый удаленный репозиторий на github (рисунок 6).

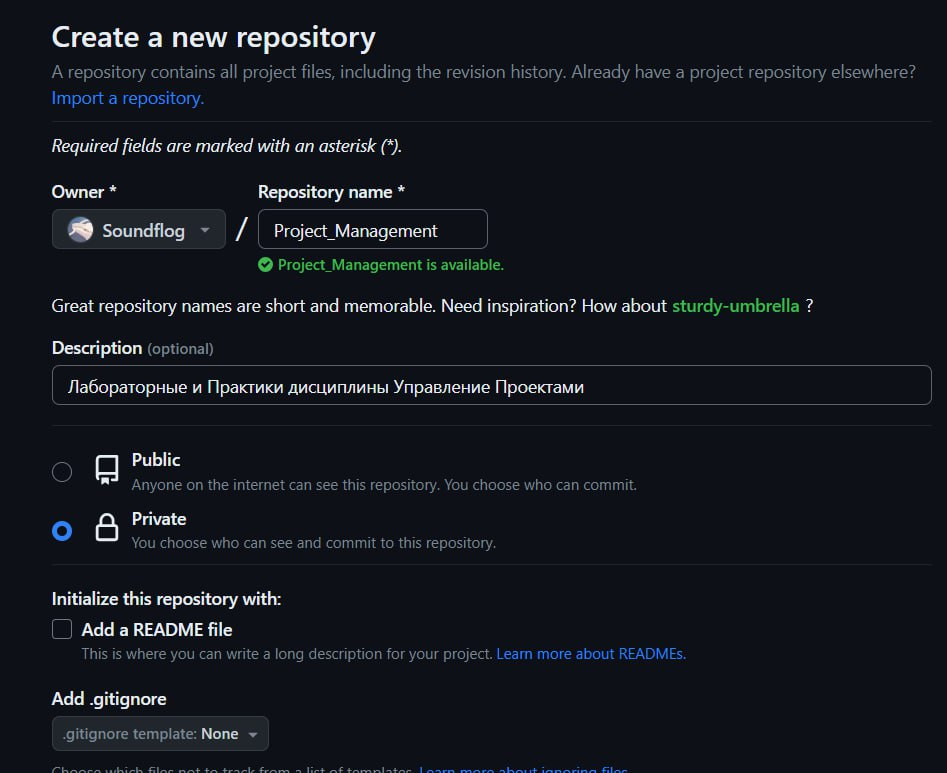


Рисунок 6 – Создание нового удаленного репозитория

Переходим в рабочую директорию локального репозитория и вводим в консоль команду: «git remote add origin [ссылка на репозиторий]».

Теперь локальный git-репозиторий успешно связан с удаленным.

После успешного подключения к репозиторию на GitHub можно отправлять данные командой: «git push origin [ветка, в которую отправить данные]». По умолчанию ветка main или master. Origin – имя репозитория по умолчанию.

Для скачивания git-репозитория из GitHub, необходимо использовать команду «git clone [ссылка\_на\_репозиторий]» (рисунок 8).

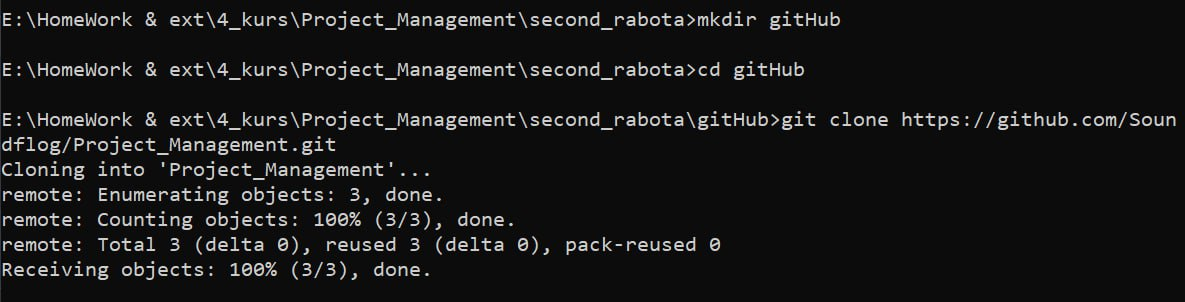


Рисунок 8 – Клонирование репозитория

Благодаря веткам Git можно создавать параллельные линии разработки, чтобы отделить новые функции или исправления от основного кода проекта. Для создания новой ветки используем команду «git branch [имя]» (рисунок 9).

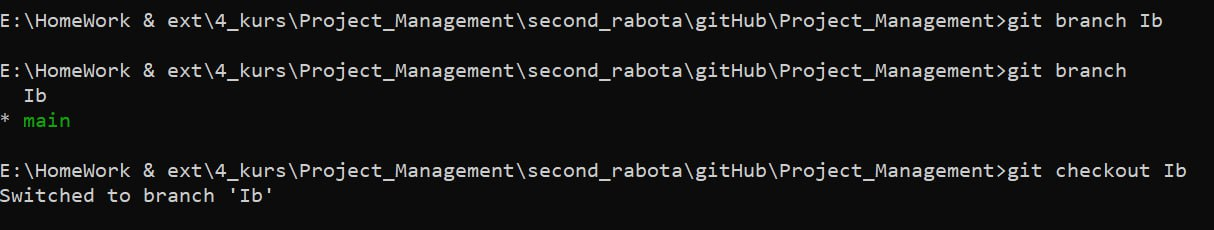


Рисунок 9 – Создание новой ветки

Команда «git checkout [имя\_ветки]» позволяет переключаться между ветками, которые уже созданы. Когда Git переключается, файлы проекта автоматически обновляются в соответствии с выбранной веткой.

С помощью файла «.gitignore» можно указать файлы и директории, которые должны быть проигнорированы Git и не отслеживаться как часть репозитория. Созданный ранее файл «text\_2\_ib.txt» был проигнорирован во время фиксации изменений (рисунок 10).

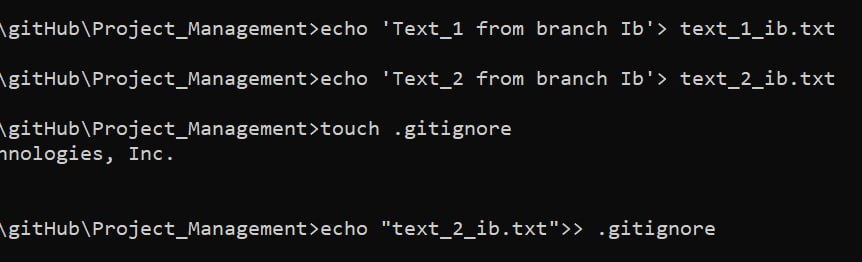


Рисунок 10 – Создание «.gitignore» файла

Чтобы перенести изменения из одной ветки в другую, используем команду «git merge [имя\_ветки]». Если изменения в обеих ветках не конфликтуют, объединение произойдет автоматически. Если возникнут проблемы, Git сообщит о них.

На рисунке 11 показано, как происходит фиксация изменений и объединение двух веток в одну.

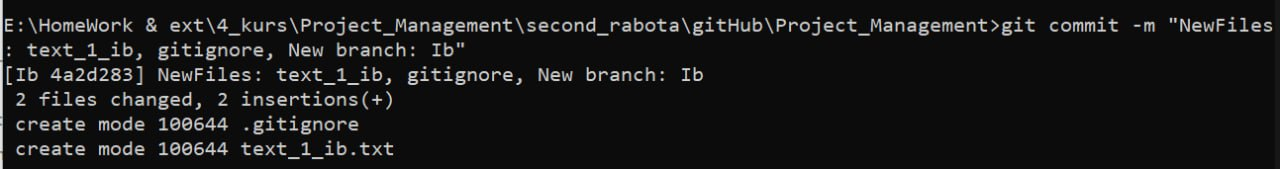


Рисунок 11 – Фиксация изменений в новой ветке

Кoманда «git branch -d [имя\_ветки]» позволяет удалить ветку после того, как она больше не нужна (рисунок 12).

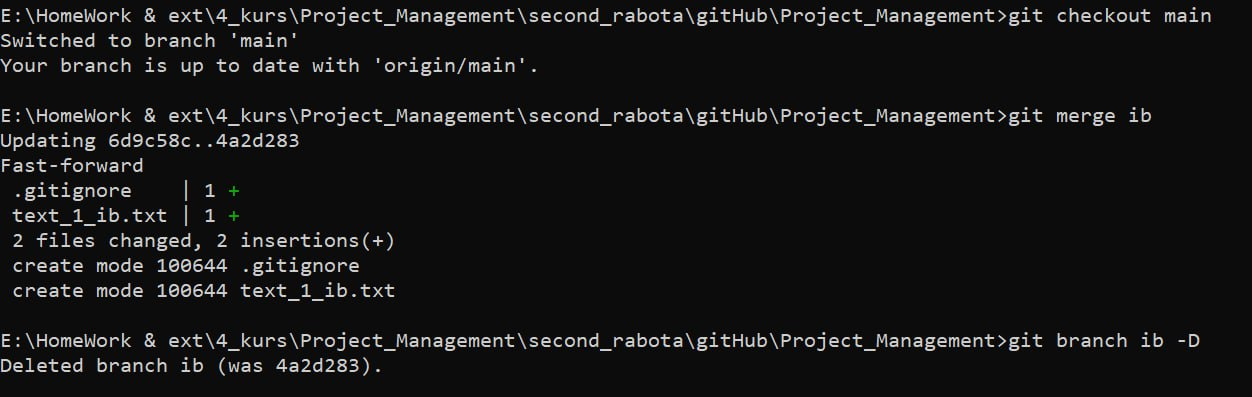


Рисунок 12 – Удаление ветки

Однако удаление ветки может быть выполнено только после того, как она была объединена с другой веткой.

Команда «git push» используется для переноса фиксаций изменений из локального репозитория в удаленный репозиторий (рисунок 13).

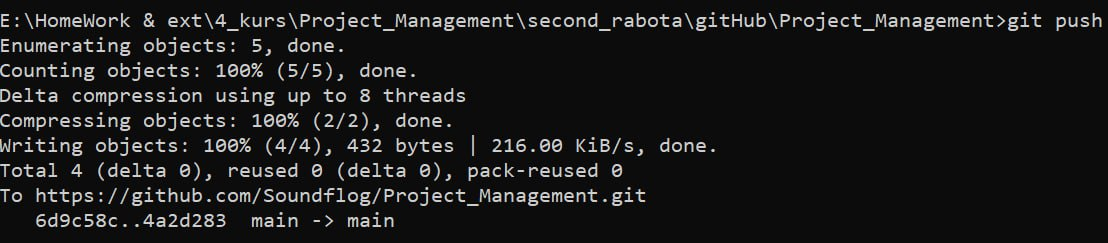


Рисунок 13 – Отправка на удаленный репозиторий

Команда «git push» без параметров передает изменения из текущей локальной ветки в соответствующую удаленную ветку.

В результате выполнения лабораторной работы были освоены основные команды по работе с Git и удаленными репозиториями на платформе GitHub.