Министерство образования Республики Беларусь  
Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Инженерно-экономический факультет

Кафедра экономической информатики

Современные технологии серверной разработки

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №1  
на тему:  
«Системы контроля версий (Git)»**

Выполнила Шимановская М. О.

372302

Проверила Ничепорук В. В.

Минск 2025

**Цель работы:** приобрести навыки работы с системой контроля версий Git, ознакомиться с особенностями платформы GitHub.

**Теоретические сведения:** GitHub – это веб-сервис, который использует Git для хостинга репозиториев. Он предоставляет интерфейс для управления проектами и совместной разработки.

Git – это система контроля версий, которая позволяет разработчикам отслеживать изменения в коде, управлять версиями и работать над проектами в команде. Он установлен локально на компьютере разработчика и предоставляет команды для работы с репозиториями.

Вкратце: Git – это инструмент для контроля версий, а GitHub – платформа для совместной работы с использованием Git.

**Задание 1.**

1 Создайте новую директорию.

2 Создайте в ней новый репозиторий.

3 Добавьте в директорию несколько файлов.

4 Проиндексируйте файлы.

5 Создайте коммит с комментарием.

6 Измените файлы.

7 Проиндексируйте измененные файлы.

8 Создайте второй коммит.

9 Просмотрите историю коммитов.

**Скрипт выполнения задания 1 в терминале Git:**

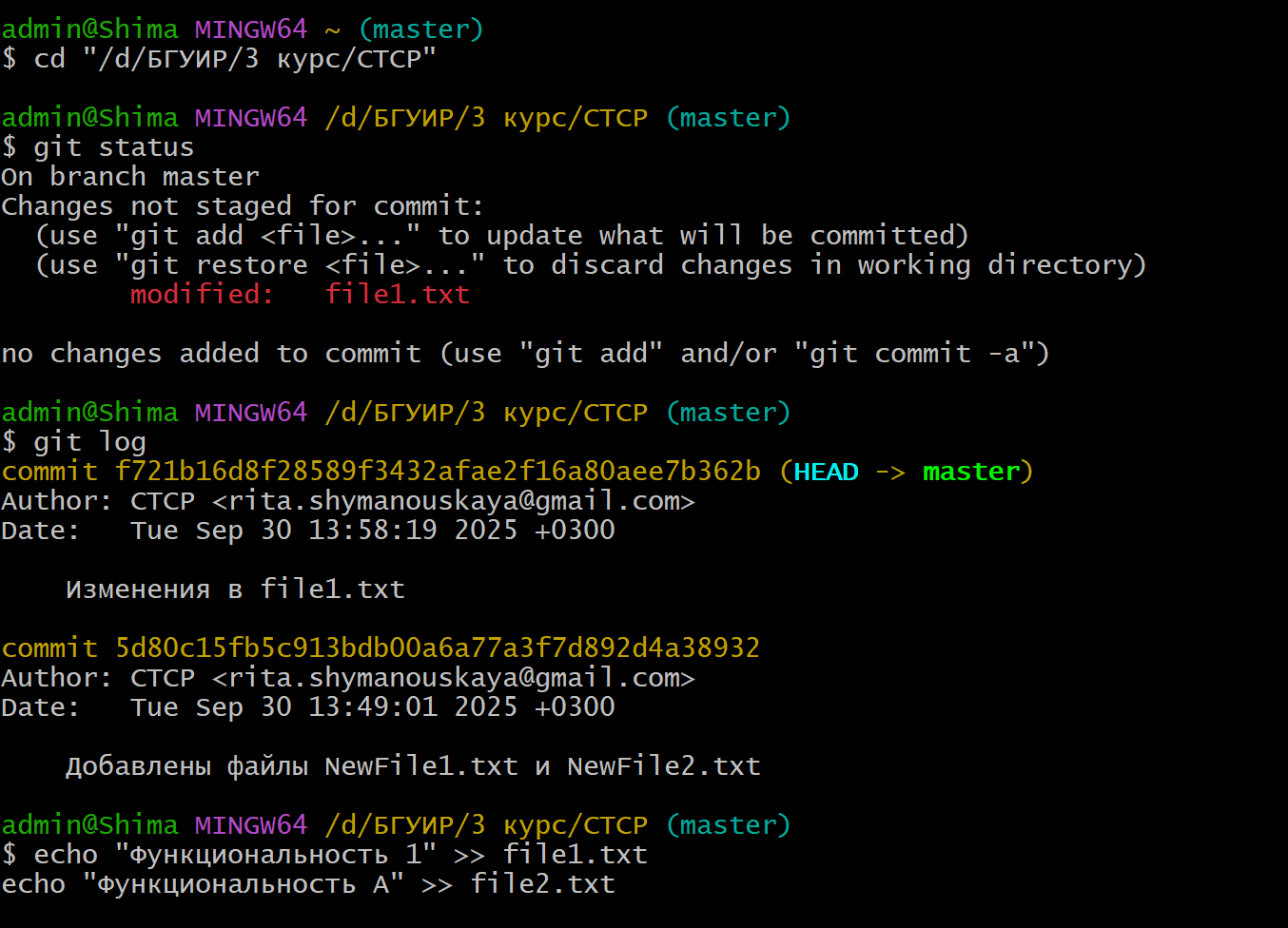


Рисунок 1 – Задание 1

**Задание 2.**

1 Откройте ранее созданную директорию.

2 Добавьте новую ветку Dev.

3 Изменяйте файлы в директории, создайте несколько коммитов в ветке

Dev и сделайте merge, rebase или cherry pick на выбор.

4 Восстановите состояние файлов любым способом, предоставляемым

технологией Git.

**Скрипт выполнения задания 2 в терминале Git:**

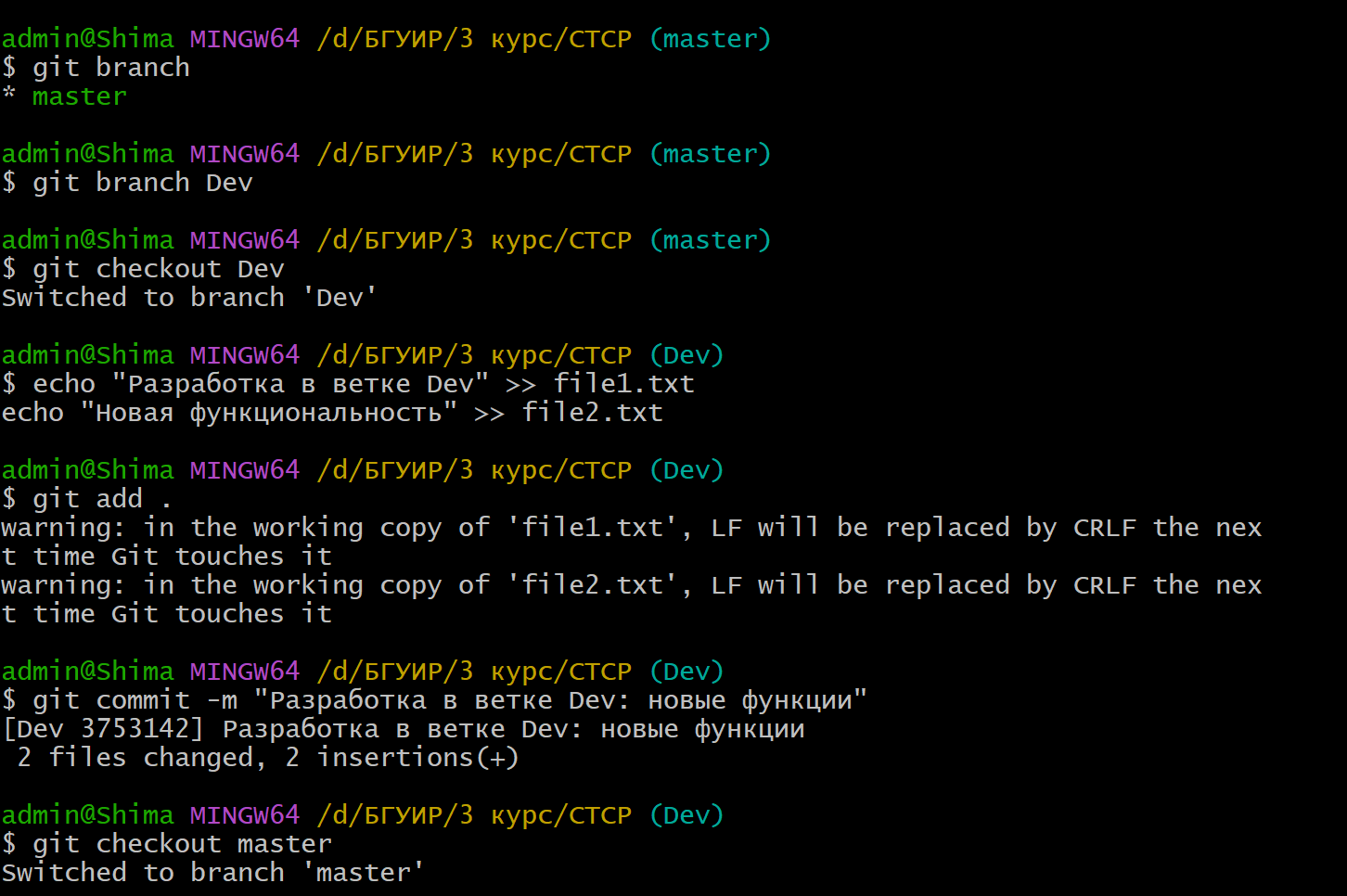


Рисунок 1 – Задание 1

**Задание 3.**

1 Создайте удалённый репозиторий.

2 Отправьте изменения из ранее созданного локального репозитория в удалённый.

3 Передайте напарнику ссылку на свой репозиторий.

4 Предоставьте ему доступ к своему репозиторию (Settings > Collaborators).

5 Напарник создаёт Fork.

6 Напарник клонирует репозиторий.

7 Напарник вносит изменения в файлы и загружает изменения в удалённый репозиторий.

8 Напарник создаёт Pull Request.

9 Просмотрите Pull Request и сделайте Merge.

10 Загрузите актуальную версию файлов из удалённого репозитория в свой локальный, используя Rebase для слияния.

**Скрипт выполнения задания 3 в терминале Git:**

