**Отчет по самостоятельной работе №**3

**по дисциплине МДК 01.01 “Инструментальные средства разработки ПО”.**

Выполнил: студент

группы 319

Бакеев Рамзан Русланович

Дата 04.12.2024

**Цель работы**

Цель данной работы – изучение и применение системы контроля версий на практике, включая настройку типов импортируемых файлов, путей, фильтров и других параметров для эффективного управления репозиториями.

**Основная структура задания**

Задание 1.

1. Определим список типов файлов, которые будут допустимы для нашего проекта.

.txt (текстовые файлы)

.md (Markdown файлы)

.py (Python скрипты)

.jpg, .png (изображения)

.json (файлы конфигурации)

.gitignore (файл исключений Git)

txt, .md, .py, .json — основные типы файлов, используемые в проекте.

.jpg, .png — изображения, которые могут быть использованы в документации или интерфейсе.

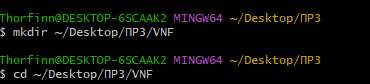
.gitignore — необходим для исключения ненужных файлов из репозитория.

2) Настроим систему контроля версий для работы с указанными типами файлов.

Git не требует специальной настройки для работы с определенными типами файлов, так как он работает с любыми файлами. Важно правильно настроить .gitignore, чтобы исключить ненужные файлы.

Задание 2.

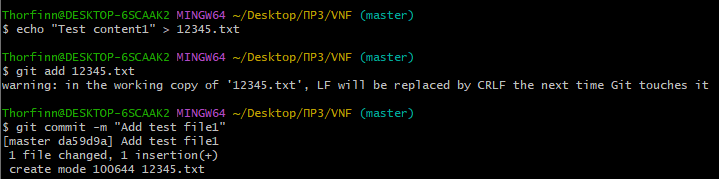
1. Настроем систему контроля версий для нашего проекта, указав абсолютные и относительные пути к основным директориям.



Инициализируем гит репозиторий:

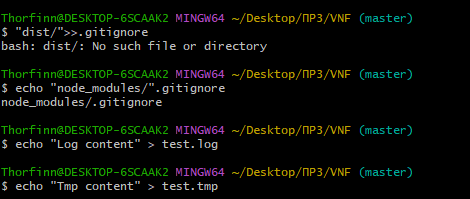


2) Проверим корректность настройки путем создания и сохранения тестового файла.

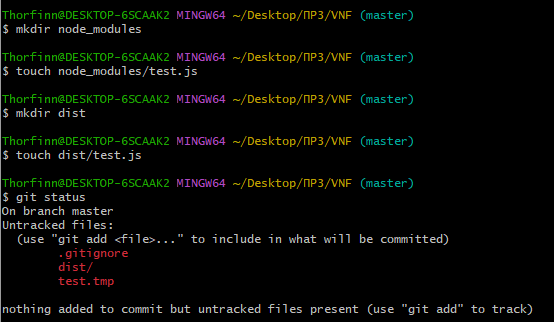


Задание 3.

1)Создаем файл .gitignore для нашего проекта, включив в него временные файлы и директории, которые не должны быть добавлены в репозиторий.



2) Проверим работу .gitignore, добавив тестовые файлы и убедившись, что они исключены.

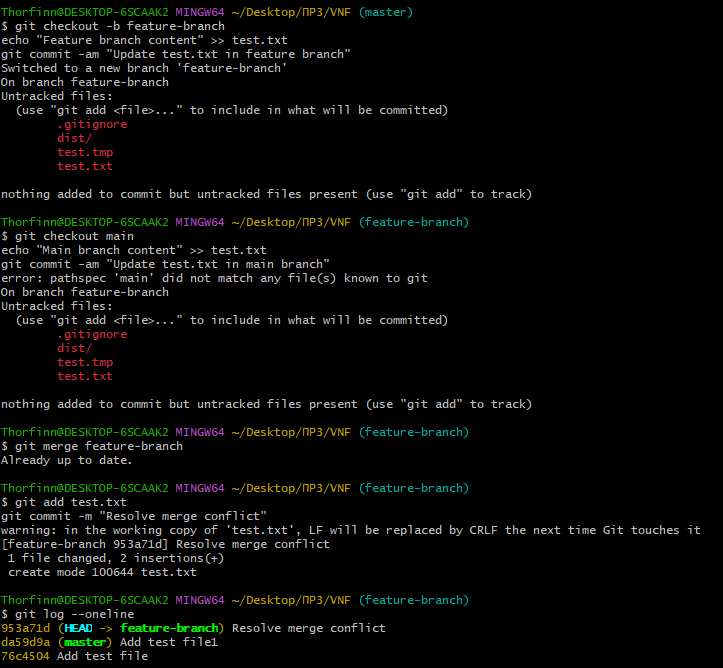


Задание 4

1. Настроим систему контроля версий так, чтобы при возникновении конфликтов использовались правила слияния.



1. Проверим работу системы при попытке одновременного изменения одного и того же файла с разных версий.



**Выводы работы**

Изучение и применение системы контроля версий на практике, включая настройку типов импортируемых файлов, путей, фильтров и других параметров для эффективного управления репозиториями.