Отчет по выполнению лабораторной работы 2

Дисциплина: Архитектура компьютеров и опрерационные системы:Операционные системы

Ефремова Полина Александровна

Содержание

Список иллюстраций

Список таблиц

# 1 Цель работы

Научиться работать с языком разметки Markdown посредством выполнения отчета лабораторной работы №2.

# 2 Задание

1. Настройка GitHub.
2. Базовая настройка Git.
3. Создание SSH-ключа.
4. Создание рабочего пространства и репозитория курса на основе шаблона.
5. Верификация коммитов с помощью PGP.
6. Настройка каталога курса.

# 3 Теоретическое введение

Система контроля версий ( Version Control System, VCS ) — это инструмент, используемый разработчиками программного обеспечения для управления изменениями в исходном коде и других файловых ресурсах. Системы контроля версий разработаны специально для того, чтобы максимально упростить и упорядочить работу над проектом (вне зависимости от того, сколько человек в этом участвуют). СКВ дает возможность видеть, кто, когда и какие изменения вносил; позволяет формировать новые ветви проекта, объединять уже имеющиеся; настраивать контроль доступа к проекту ; осуществлять откат до предыдущих версий. В классических системах контроля версий используется централизованная модель, предполагающая наличие единого репозитория для хранения файлов. Выполнение большинства функций по управлению версиями осуществляется специальным сервером. Участник проекта (пользователь) перед началом работы посредством определённых команд получает нужную ему версию файлов. После внесения изменений, пользователь размещает новую версию в хранилище. При этом предыдущие версии не удаляются из центрального хранилища и к ним можно вернуться в любой момент. Сервер может сохранять не полную версию изменённых файлов, а производить так называемую дельта-компрессию — сохранять только изменения между последовательными версиями, что позволяет уменьшить объём хранимых данных. Системы контроля версий поддерживают возможность отслеживания и разрешения конфликтов, которые могут возникнуть при работе нескольких человек над одним файлом. Можно объединить (слить) изменения, сделанные разными участниками (автоматически или вручную), вручную выбрать нужную версию, отменить изменения вовсе или заблокировать файлы для изменения. В зависимости от настроек блокировка не позволяет другим пользователям получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом, привилегированный доступ только одному пользователю, работающему с файлом. Демидова А. В. 14 Архитектура ЭВМ Системы контроля версий также могут обеспечивать дополнительные, более гибкие функциональные возможности.Например, они могут поддерживать работу с несколькими версиями одного файла, сохраняя общую историю изменений до точки ветвления версий и собственные истории изменений каждой ветви. Кроме того, обычно доступна информация о том, кто из участников, когда и какие изменения вносил. Обычно такого рода информация хранится в журнале изменений, доступ к которому можно ограничить. В отличие от классических, в распределённых системах контроля версий центральный репозиторий не является обязательным. Среди классических VCS наиболее известны CVS, Subversion, а среди распределённых — Git, Bazaar, Mercurial. Принципы их работы схожи, отличаются они в основном синтаксисом используемых в работе команд. Система контроля версий Git представляет собой набор программ командной строки. Доступ к ним можно получить из терминала посредством ввода команды git с различными опциями. Благодаря тому, что Git является распределённой системой контроля версий, резервную копию локального хранилища можно сделать простым копированием или архивацией

# 4 Выполнение лабораторной работы

## 4.1 Установка программного обеспечаения

1. С помощью команд dnf install gh и dnf install git-all устанавливаю необходимое обеспечение (рис. 1).

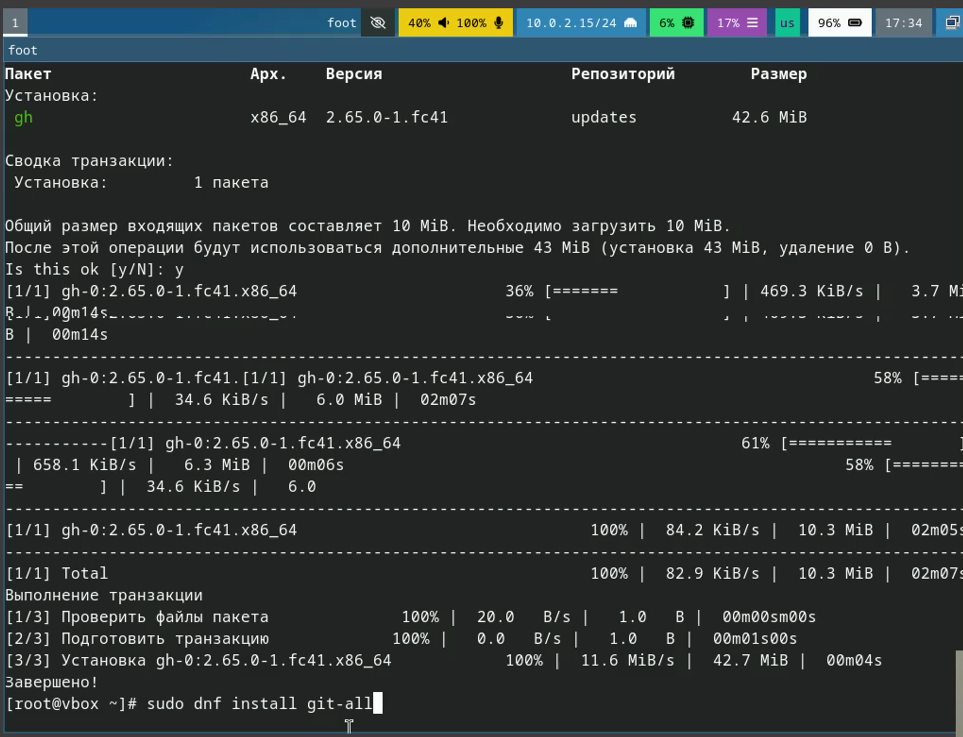


Рис. 1: Установка ПО

## 4.2 Базовая настройка Git

1. Задаю имя и email владельца репозитория, настраиваю utf-8 в выводе сообщения git, задаю имя начальной ветки а также задаю параметры autocrlf и safecrlf (рис. 2).

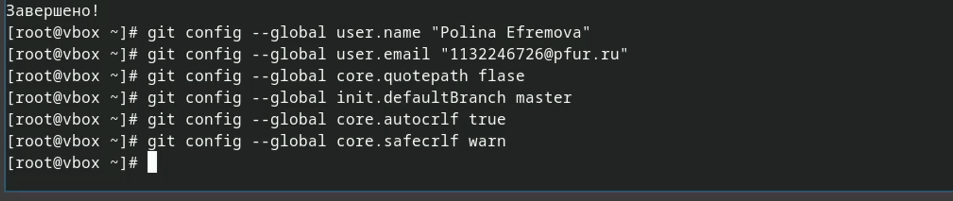


Рис. 2: Базовая настройка Git

## 4.3 Создание ключа SSH

1. Создаю ключ по алгоритму 4096 (рис. 3).

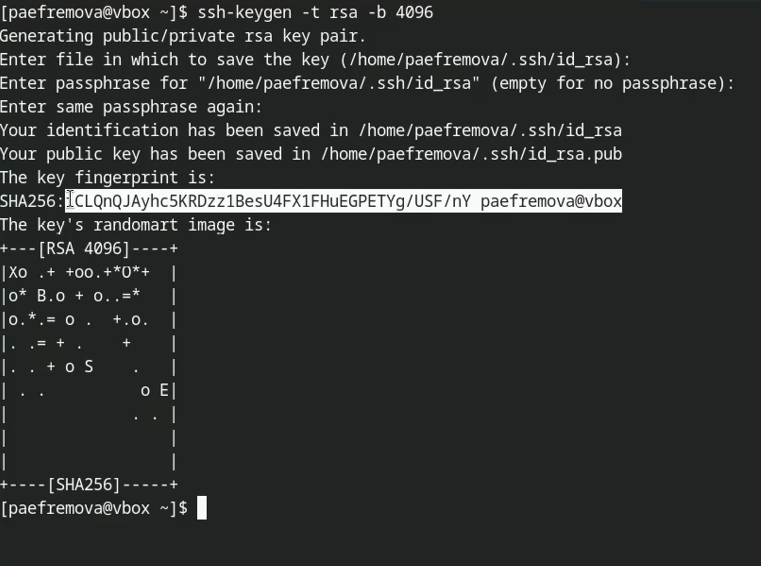


Рис. 3: Алгоритм 4096

1. Создаю ключ по алгоритму ed25519 (рис. 4).

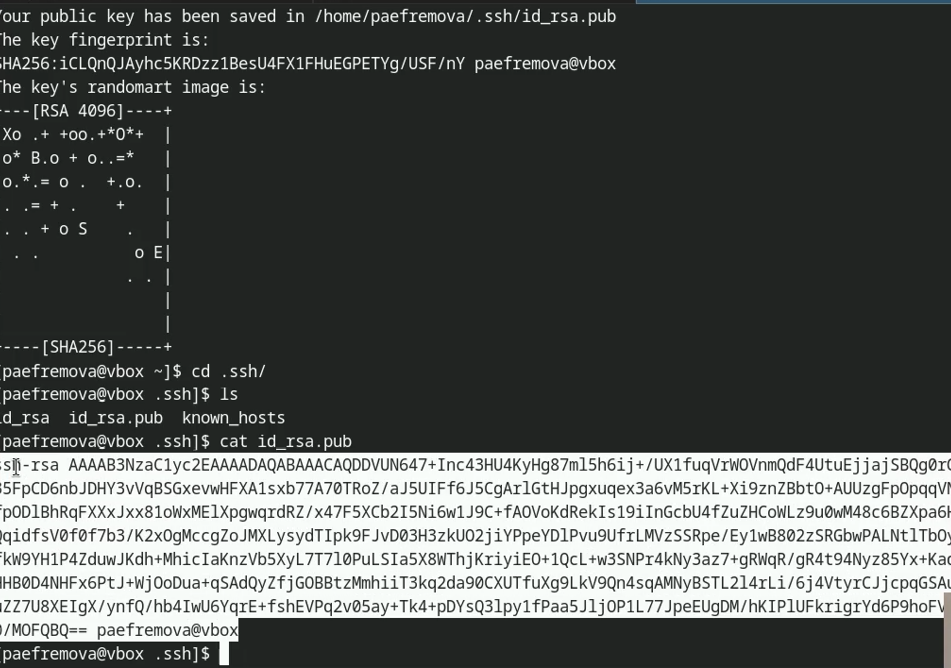


Рис. 4: Алгоритм ed25519

1. Добавляю скопированный SSH ключ на GitHub (рис. 5).

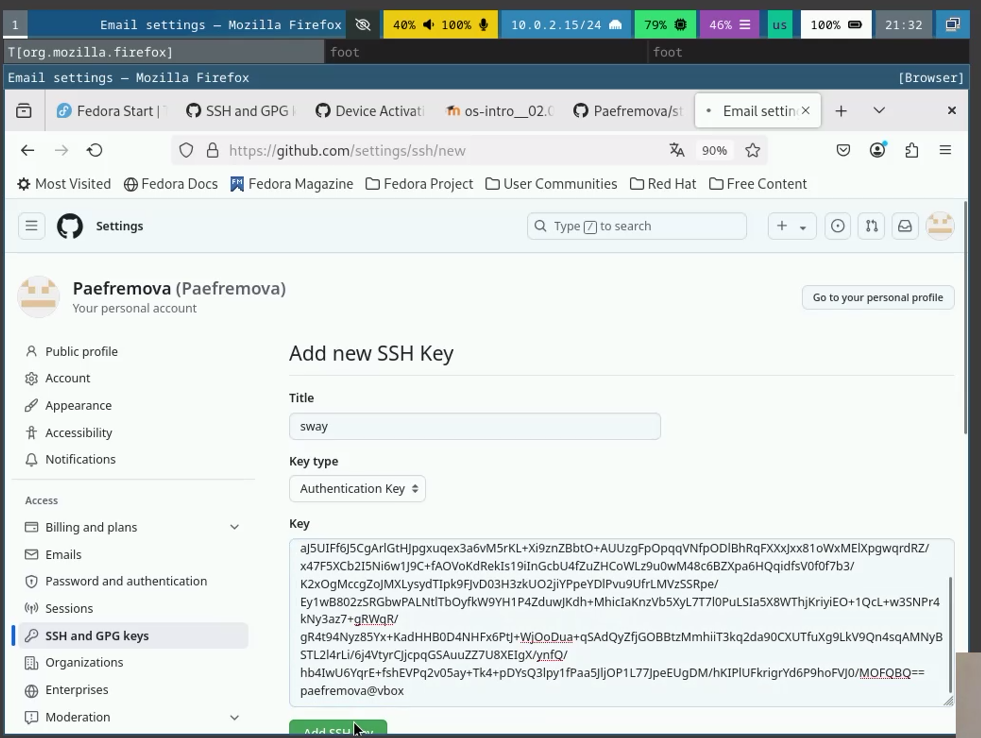


Рис. 5: Загрузка ключа на Github

## 4.4 Создание gpg ключа

1. Генерирую gpg ключ (рис. 6).

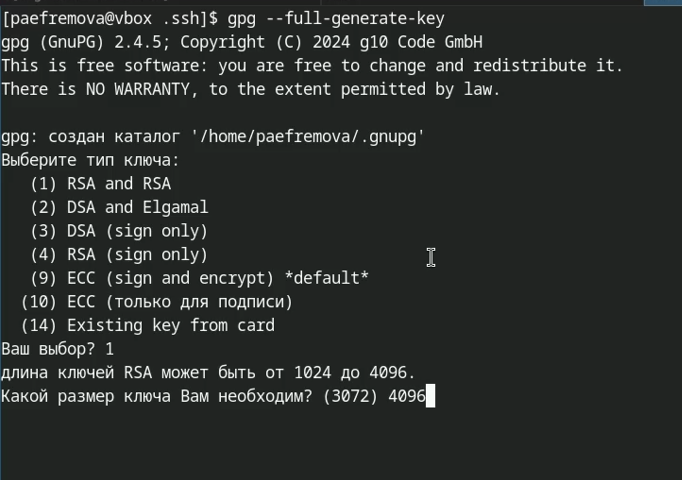


Рис. 6: GPG ключ

1. Настраиваю этот gpg ключ указанным способом (рис. 7).

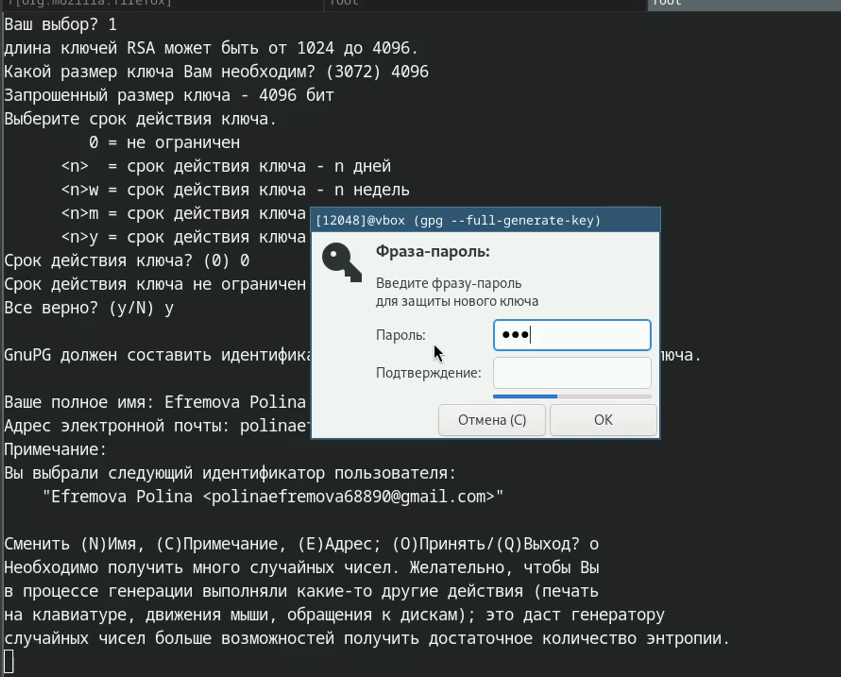


Рис. 7: GPG ключ - настройка

1. Вывожу список ключей и копирую отпечаток приватного ключа (рис. 8).



Рис. 8: Список ключей

1. Копирую сгенерированный PGP ключ (рис. 9).

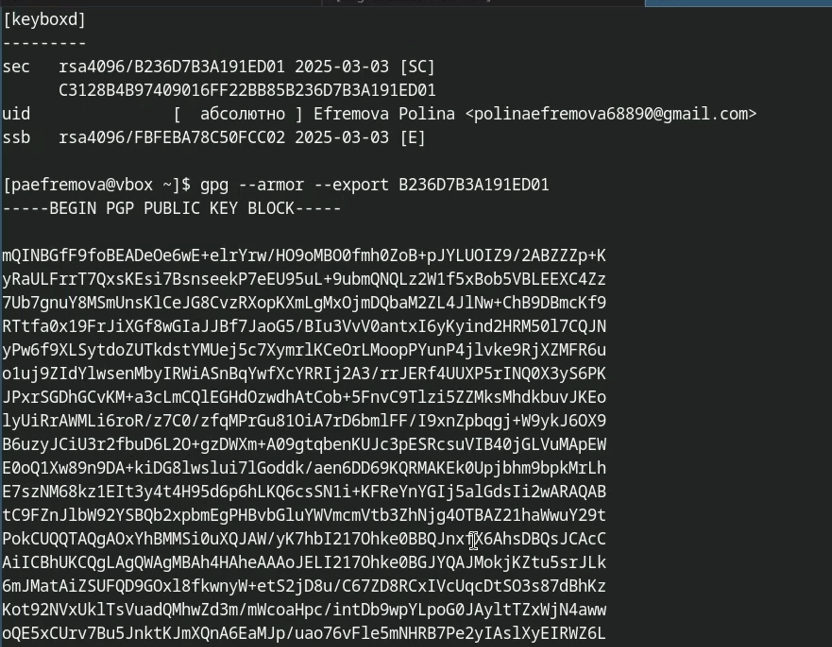


Рис. 9: GPG ключ - копирование

1. Добавляю ключ на Github (рис. 10).

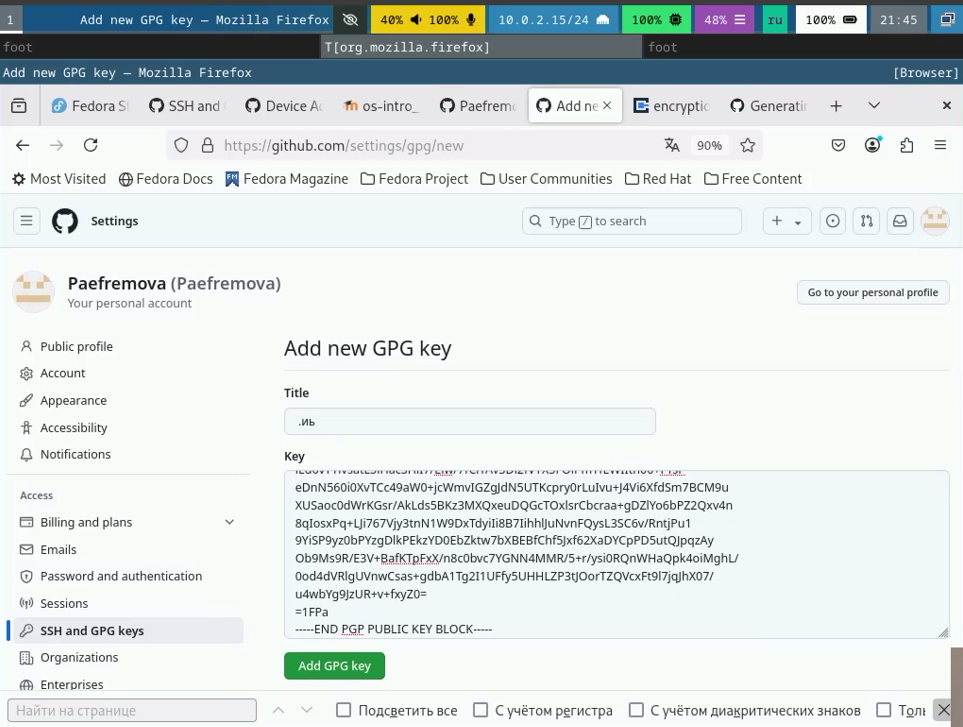


Рис. 10: GPG ключ - GitHub

1. Авторизируюсь, отвечаю на контрольные вопросы (рис. 11).

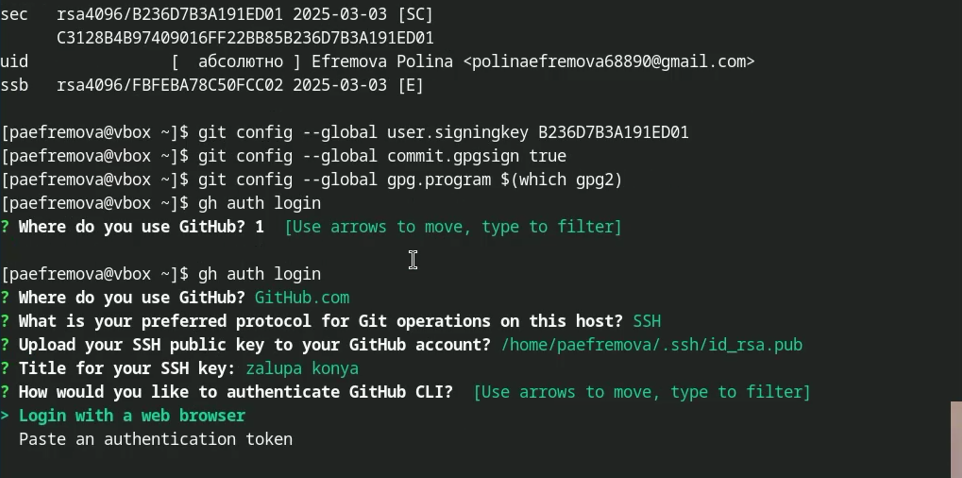


Рис. 11: Авторизация

1. Ввожу код (рис. 12)

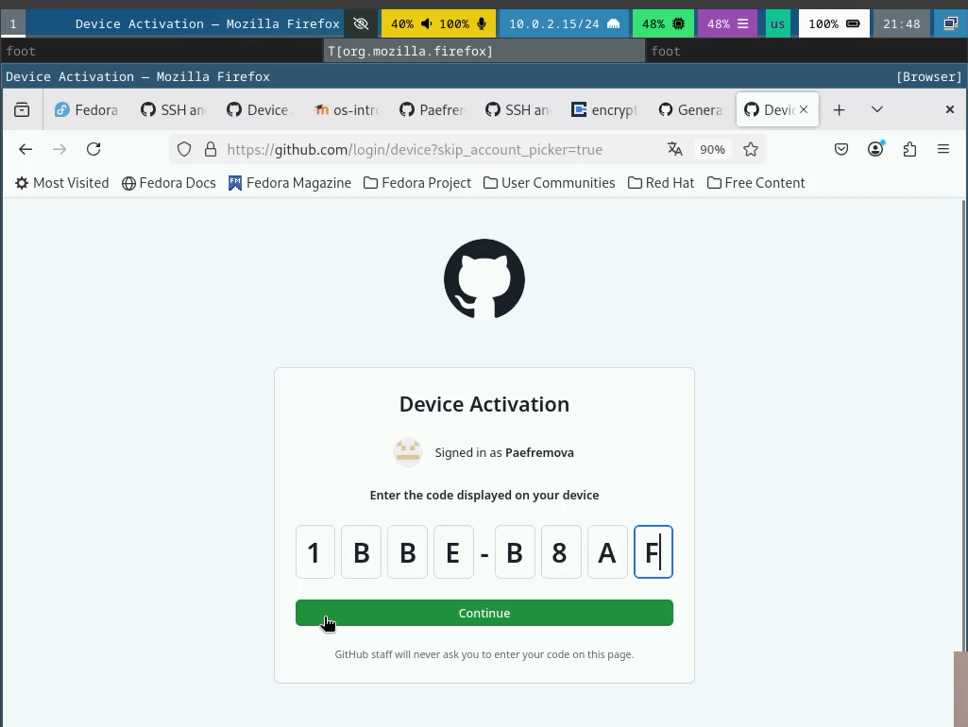


Рис. 12: Код

## 4.5 Создание репозитория курса на основе шаблона

1. Создаю репозиторий (рис. 13).

Рис. 13: Репозиторий

Рис. 13: Репозиторий

1. Перехожу в него и копирую шаблон (рис. 14).

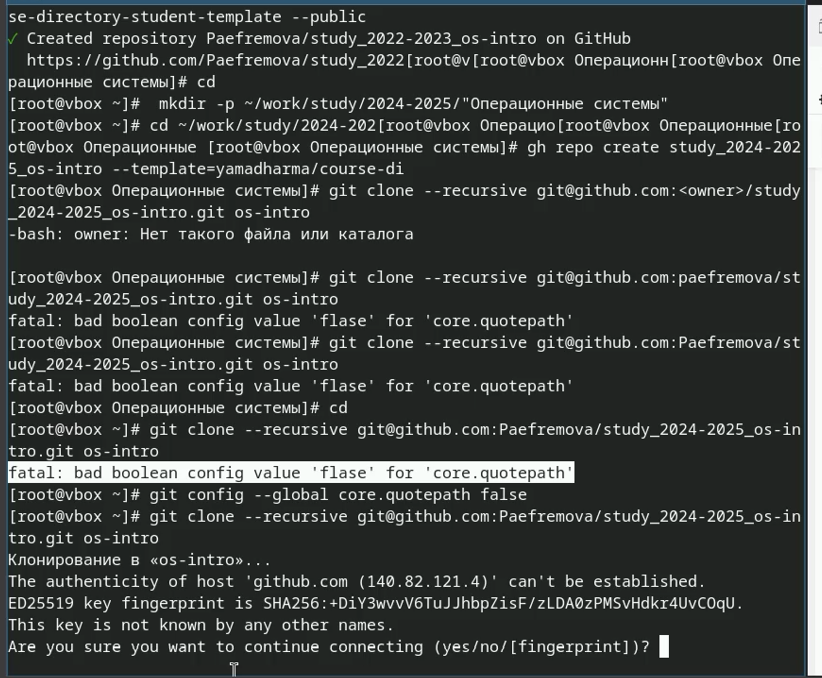


Рис. 14: Добавление шаблона

1. Перехожу в каталог курса (рис. 15).

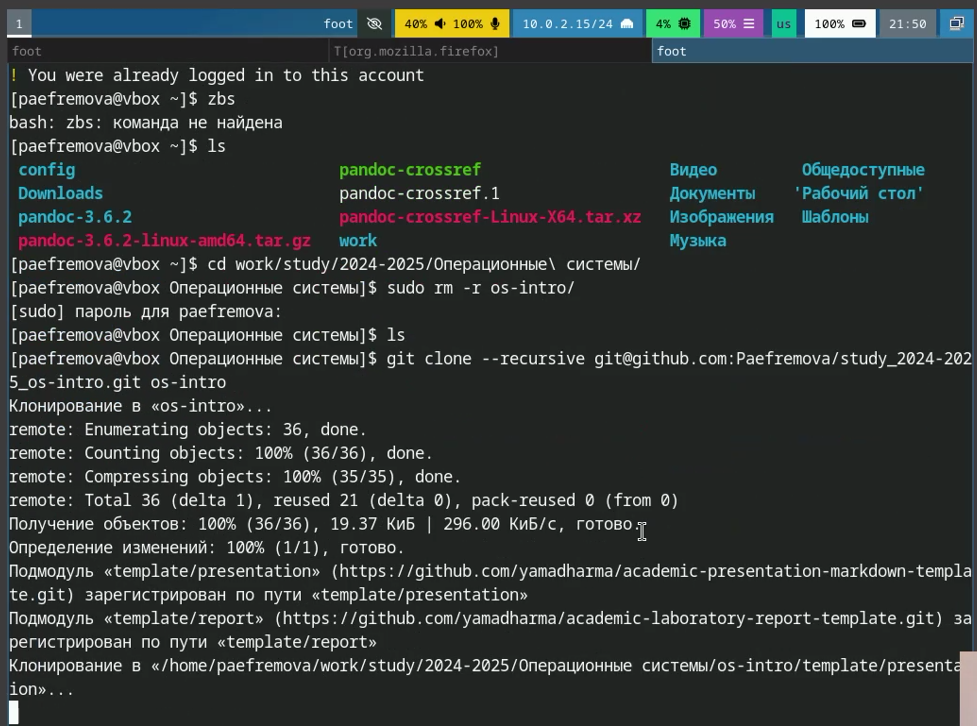


Рис. 15: Каталог курса

1. Создаю необходимые каталоги (рис. 16).

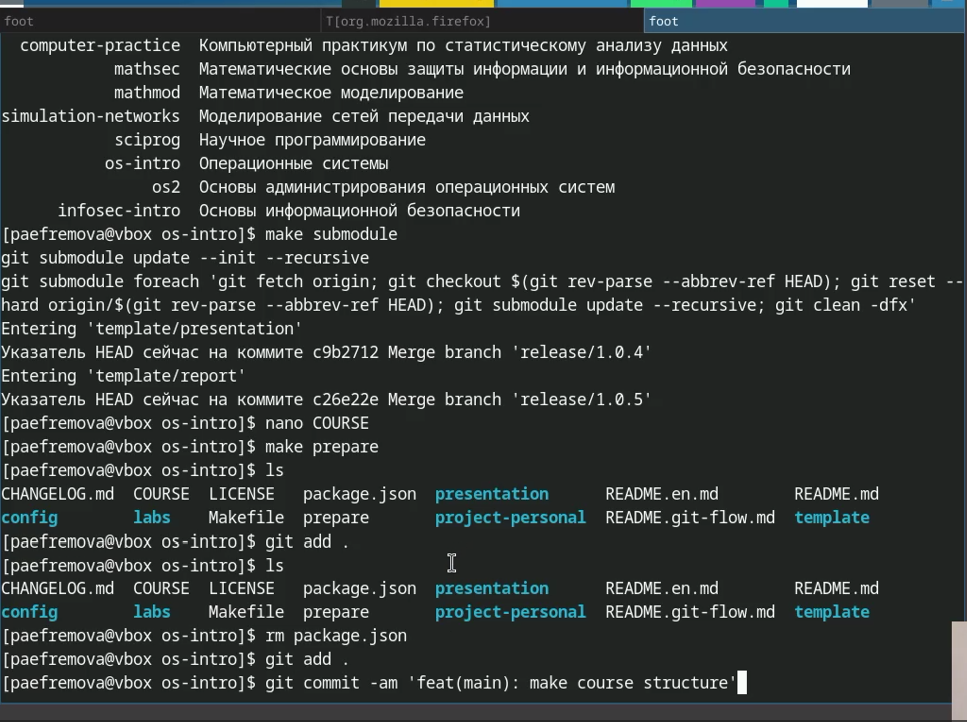


Рис. 16: Необходимые каталоги

1. Отправляю файлы на сервер (рис. 17).

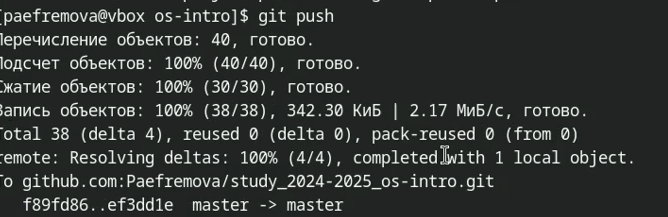


Рис. 17: Сервер - загрузка

# 5 Выводы

В заключение хочется отметить, что данная лабораторная работа позволила мне научиться работать с системой Git. Я практиковала свои навыки в работе с командной строкой, теперь уже связывая выполнимое с директориями GitHub.

Кроме этого, я научилась работать с Markdown (вспмонила, опираясь на материалы из прошлого семестра в том числе)

# 6 Список литературы

1. [Архитектура ЭВМ](https://esystem.rudn.ru/pluginfile.php/2089082/mod_resource/content/0/Лабораторная%20работа%20№2.%20Система%20контроля%20версий%20Git.pdf)
2. [Ефремова Полина, Лабораторная работа №2, Архитектура компьютеров](https://github.com/Paefremova/study_2024-2025_arh-pc/blob/master/labs/lab02/report/report.md)
3. [Лабораторная 2](https://esystem.rudn.ru/mod/page/view.php?id=1224371)
4. [лабораторная 3](https://esystem.rudn.ru/pluginfile.php/2586858/mod_resource/content/3/003-lab_markdown.pdf)