Лабораторная работа №2

Система контроля версий Git

Седохин Даниил Алексеевич

Содержание

# 1 Цель работы

Целью работы является изучить идеологию и применение средств контроля версий. Приобрести практические навыки по работе с системой git.

# 2 Теоретическое введение

Системы контроля версий (Version Control System, VCS) применяются при работе нескольких человек над одним проектом. Обычно основное дерево проекта хранится в локальном или удалённом репозитории, к которому настроен доступ для участников проекта. При внесении изменений в содержание проекта система контроля версий позволяет их фиксировать, совмещать изменения, произведённые разными участниками проекта, производить откат к любой более ранней версии проекта, если это требуется. 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.

Здесь описываются теоретические аспекты, связанные с выполнением работы.

Например, в табл. [1](#tbl:std-dir) приведено краткое описание стандартных каталогов Unix.

Table 1: Описание некоторых каталогов файловой системы GNU Linux

| Имя каталога | Описание каталога |
| --- | --- |
| / | Корневая директория, содержащая всю файловую |
| /bin | Основные системные утилиты, необходимые как в однопользовательском режиме, так и при обычной работе всем пользователям |
| /etc | Общесистемные конфигурационные файлы и файлы конфигурации установленных программ |
| /home | Содержит домашние директории пользователей, которые, в свою очередь, содержат персональные настройки и данные пользователя |
| /media | Точки монтирования для сменных носителей |
| /root | Домашняя директория пользователя root |
| /tmp | Временные файлы |
| /usr | Вторичная иерархия для данных пользователя |

# 3 Выполнение лабораторной работы

3.1 Базовая настройка git

3.1.1. Делаем предварительную конфигурацию git. Для этого в терминале вводим следующие команды, указав имя и email владельца репозитория. (рис. ??)

Предварительная конфигурация git

Предварительная конфигурация git

3.1.2. Настройка utf-8 в выводе сообщений git. (рис. ??)

utf-8

utf-8

3.1.3. Задаём имя начальной ветке. (рис. ??)

Задаём имя начальной ветке

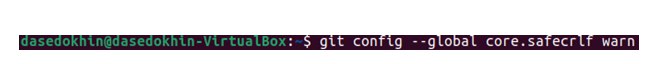
Задаём имя начальной ветке

3.1.4. Параметр autocrlf. (рис. ??)

Параметр autocrlf

Параметр autocrlf

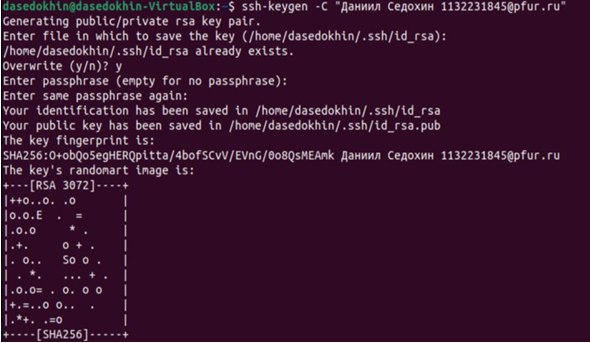
3.1.5. Параметр safecrlf. (рис. ??)



Параметр safecrlf

3.2 Создание SSH ключа

3.2.1. Для идентификации пользователя на сервере репозиториев генерируем несколько ключей. (рис. ??)



Генерация ключей

3.2.2. Теперь нам необходимо загрузить сгенерированный открытый ключ. Для этого заходим на сайт http://github.org/ под своей учётной записью и переходим в меню «Setting». После этого выбираем в боковом меню «SSH and GPG keys» и нажимаем на кнопку «New SSH key». Вставляем ключ в появившееся на сайте поле и указываем для ключа имя (Title). (рис. ??)



Создаём “New SSH key”

3.2.3. Копируем из локальной консоли ключ в буфер обмена. (рис. ??)

Копирование ключа

Копирование ключа

3.3 Создание рабочего пространства и репозитория курса на основе шаблона

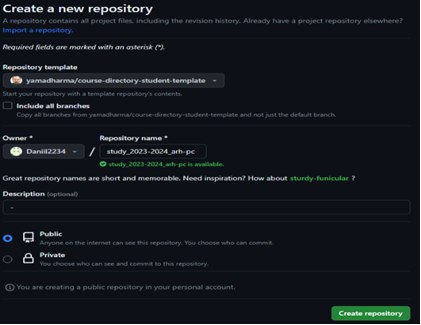
3.3.1. Создаём каталог для предмета «Архитектура компьютера». (рис. ??)

Создание нового каталога

Создание нового каталога

3.4 Создание репозитория курса на основе шаблона

3.4.1. Переходим на страницу репозитория с шаблоном курса https://github.com/yamadharma/course-directory-student-template. Задаём имя репозитория и создаём его. (рис. ??)



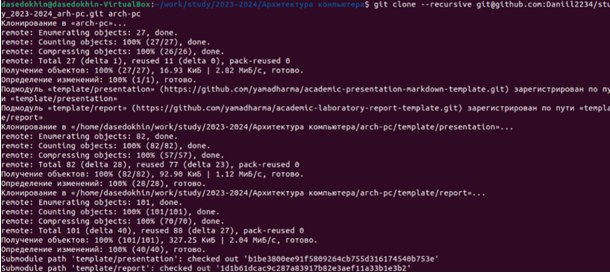
Создание репозитория

3.4.2. Переходим в каталог курса. (рис. ??)

Переходим в каталог курса

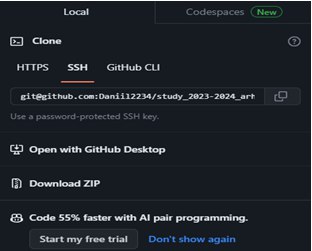
Переходим в каталог курса

3.4.3. Клонируем созданный репозиторий. (рис. ??)



Клонирование созданного репозитория

3.4.4. Предварительно перед копированием копируем ссылку для клонирования на странице созданного репозитория. (рис. ??)



Ссылка для клонирования репозитория

3.5 Настройка каталога курса

3.5.1. Переходим в каталог курса. (рис. ??)

Каталог курса

Каталог курса

3.5.2. Удаление лишних файлов. (рис. ??)

Удаление лишних файлов

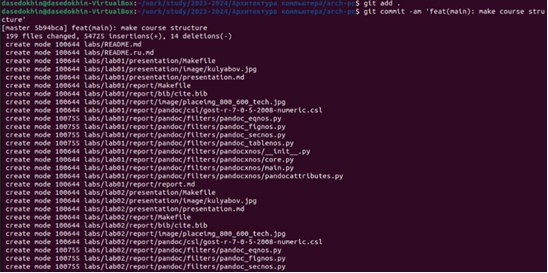
Удаление лишних файлов

3.5.3. Создание необходимых каталогов. (рис. ??)

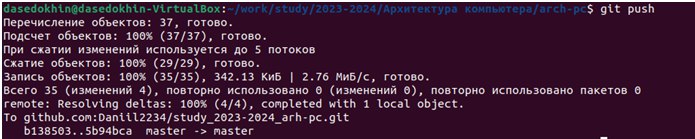
Создание необходимых каталогов

Создание необходимых каталогов

3.5.4. Отправка файлов на сервер. (рис. ?? ??)



Отправка файлов на сервер



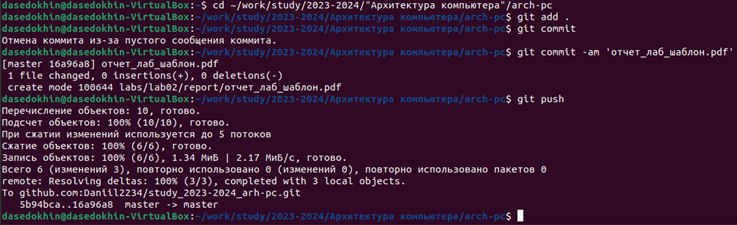
Отправка файлов на сервер

3.6 Задание для самостоятельной работы

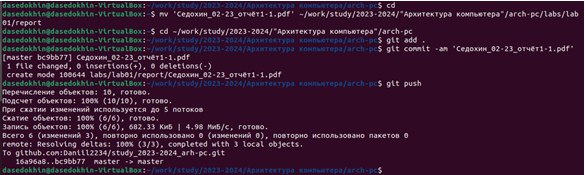
3.6.1. Создаём отчёт по выполнению лабораторной работы в соответствующем каталоге рабочего пространства (labs>lab02>report). Копируем отчёты по выполнению предыдущих лабораторных работ в соответствующие каталоги созданного рабочего пространства и загружаем файлы на github. (рис. ?? ?? ??)

Копирование отчётов

Копирование отчётов



Загрузка файлов на github



Загрузка файлов 1-й лабораторной работы на github

# 4 Выводы

В ходе выполнения лабораторной работы №2 я изучил идеологию и применение средств контроля версий. Приобрёл практические навыки по работе с системой git.