Лабораторная работа №2

Архитектура вычислительных систем

Заболотная Кристина Александровна

Содержание

# 1 Цель работы

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

# 2 Задание

Создать отчет по выполнению лабораторной работы в соответствующем каталоге рабочего пространства (labs>lab03>report). Скопировать отчеты по выполнению предыдущих лабораторных работ в соответствующие каталоги созданного рабочего пространства. Загрузить файлы на github.

# 3 Теоретическое введение

1. Системы контроля версий. Общие понятия. Системы контроля версий 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. 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 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.
2. Система контроля версий Git. 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.
3. Стандартные процедуры работы при наличии центрального репозитория. Работа пользователя со своей веткой начинается с проверки и получения изменений из центрального репозитория (при этом в локальное дерево до начала этой процедуры не должно было вноситься изменений): git checkout master git pull git checkout -b имя\_ветки Затем можно вносить изменения в локальном дереве и/или ветке. После завершения внесения какого-то изменения в файлы и/или каталоги проекта необходимо разместить их в центральном репозитории. Для этого необходимо проверить, какие файлы изменились к текущему моменту: git status и при необходимости удаляем лишние файлы, которые не хотим отправлять в центральный репозиторий. Затем полезно просмотреть текст изменений на предмет соответствия правилам ведения чистых коммитов: git diff. Если какие-либо файлы не должны попасть в коммит, то помечаем только те файлы, изменения которых нужно сохранить. Для этого используем команды добавления и/или удаления с нужными опциями: git add имена\_файлов git rm имена\_файлов Если нужно сохранить все изменения в текущем каталоге, то используем: git add . Затем сохраняем изменения, поясняя, что было сделано: git commit -am “Some commit message” и отправляем в центральный репозиторий: git push origin имя\_ветки или git push.

# 4 Выполнение лабораторной работы

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 1)

1. Сначала сделаем предварительную конфигурацию git. Откроем терминал и введём команды, указав своё имя и email (как владельца репозитория).

Рис. 1: рис.21

Рис. 1: рис.21

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 2)

1. Настроим utf-8 в выводе сообщений git, зададим имя начальной ветки (будем называть её master), параметр autocrlf, параметр safecrlf.

Рис. 2: рис.22а

Рис. 2: рис.22а

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 3)

Пункт 2 - продолжение.

Рис. 3: рис.22б

Рис. 3: рис.22б

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 4)

1. Для последующей идентификации пользователя на сервере репозиториев необходимо сгенерировать пару ключей (приватный и открытый).

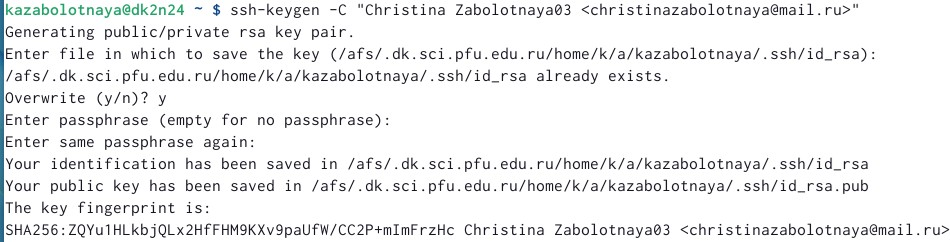


Рис. 4: рис.23

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 5)

1. Далее загрузим сгенерённый открытый ключ на сайте http://github.org/ под своей учётной записью и перейдем в меню Setting . После этого выберем в боковом меню SSH and GPG keys и нажмем кнопку New SSH key. Скопируем из локальной консоли ключ в буфер обмена. Вставляем ключ в появившееся на сайте поле и указываем для ключа имя (Title).

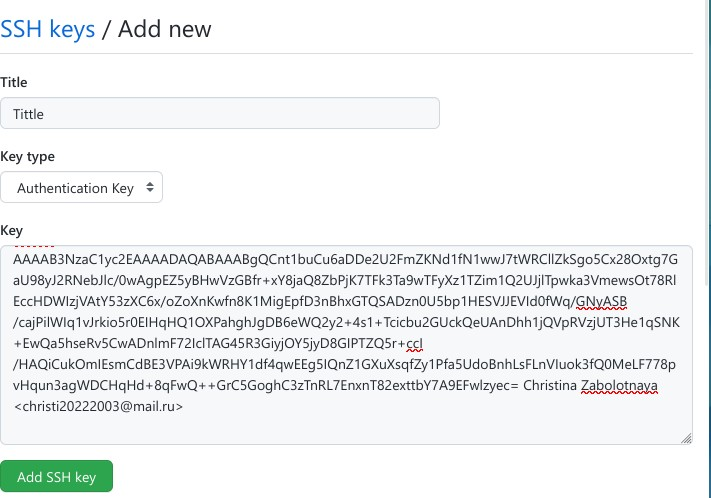


Рис. 5: рис.24а

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 6)

Пункт 4 - продолжение.

Рис. 6: рис.24б

Рис. 6: рис.24б

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 7)

1. Создадим репозиторий, дадим ему название (study\_2022–2023\_arh-pc).

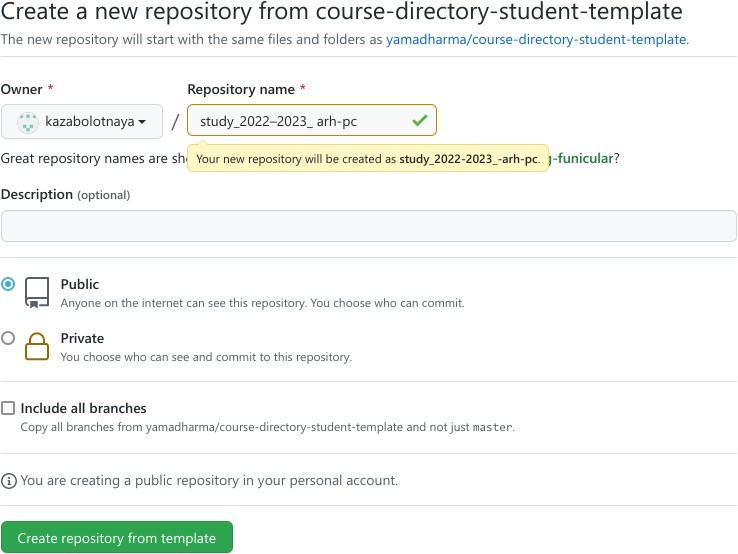


Рис. 7: рис.25

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 8)

1. Откроем терминал и создадим каталог для предмета «Архитектура компьютера», перейдём в каталог курса.

Рис. 8: рис.26

Рис. 8: рис.26

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 9)

1. Клонируем созданный репозиторий (Ссылку для клонирования скопируем на странице созданного репозитория).

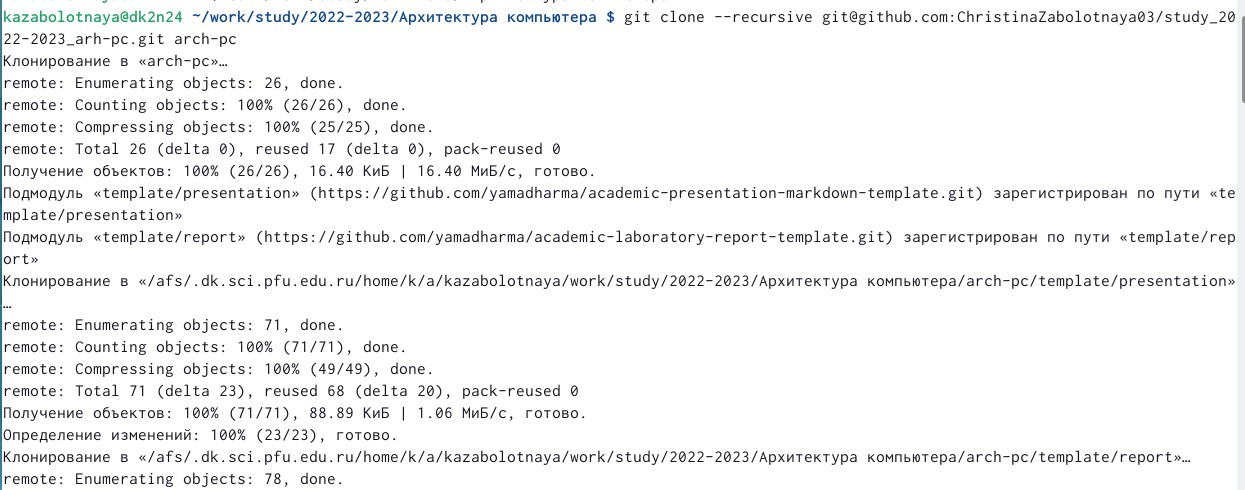


Рис. 9: рис.27а

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 10)

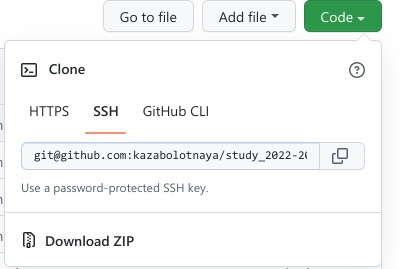


Рис. 10: рис.27б

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 11)

1. Перейдём в каталог курса. Удалим лишние файлы. Создадим необходимые каталоги.

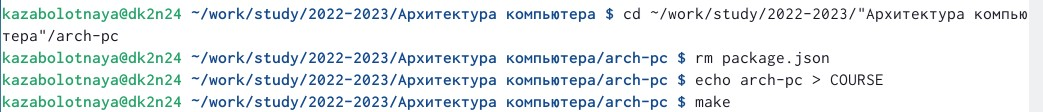


Рис. 11: рис.28

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 12)

1. Отправим файлы на сервер.

Рис. 12: рис.29а

Рис. 12: рис.29а

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 13)



Рис. 13: рис.29б

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 14)

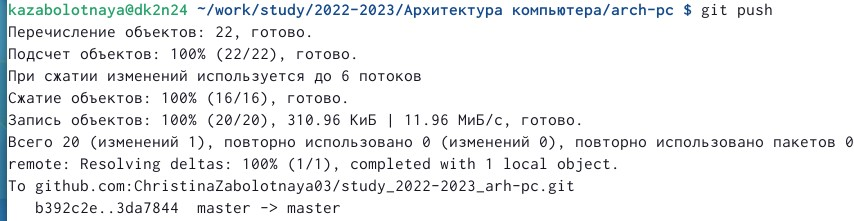


Рис. 14: рис.29в

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 15)

1. Отправим файлы на сервер.

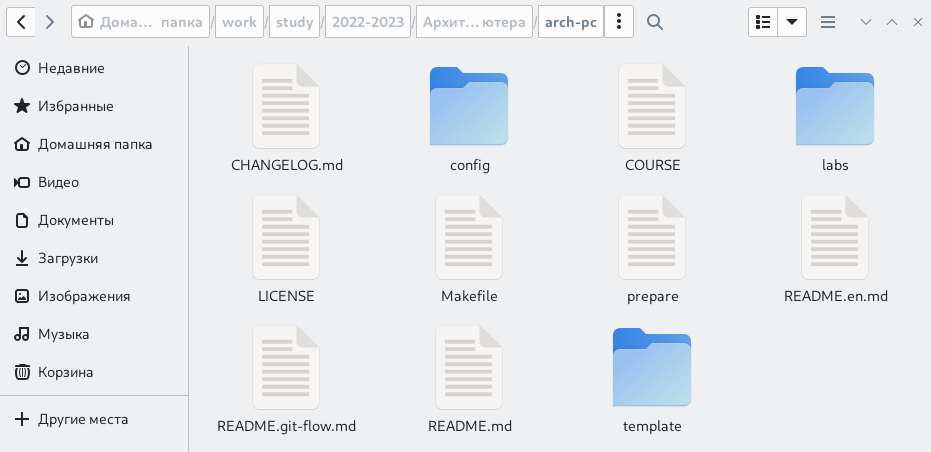


Рис. 15: рис.210

# 5 Выводы

В ходе изучения данной лабораторной работы были приобретены практические навыки по работе с системой git, научились создавать репозиторий, отправлять файлы на сервер. Изучили идеологию и применение средств контроля версий.

# Список литературы