РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук Кафедра компьютерных и информационных наук

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2

дисциплина: Архитектура компьютеров

Студент: Алексеев Тимофей Группа: НКАбд-07-25

МОСКВА

2025 г.

Page 1

Оглавление

1. Цель работы	3
2. Задание	
3. Теоретическое введение	5
4. Выполнение лабораторной работы	7
4.1 Техническое обеспечение	7
4.2 Базовая настройка Git	7
4.3 Создание рабочего пространства и репозит	
са на основе шаблона	9
4.4 Создание репозитория курса на основе шаб 12	лона. 10
5. Задания для самостоятельной работы	12
6. Выводы	13
Список литературы	14

1. Цель работы

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

2. Задание

На основе методических указаний провести работу с базовыми командами системы контроля версий git, выучить применение команд для разных случаев использования, настроить GitHub.

3. 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. 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.

Основные команды git

Команда	Описание
git init	создание основного дерева репозитория
git pull	получение обновлений (изменений) текущего дерева из центрального репозитория
git push	отправка всех произведённых изменений локального дерева в центральный репозиторий
git status	просмотр списка изменённых файлов в текущей директории
git diff	просмотр текущих изменений
git add .	добавить все изменённые и/или созданные файлы и/или каталоги
Команда	Описание
git add имена_файлов	добавить конкретные изменённые и/или созданные файлы и/или каталоги
git rm имена_файлов	удалить файл и/или каталог из индекса репозитория (при этом файл и/или каталог остаётся в локальной директории)
git commit -am 'Описание коммита'	сохранить все добавленные изменения и все изменённые файлы
git checkout -b имя_ветки	создание новой ветки, базирующейся на текущей
git checkout имя_ветки	переключение на некоторую ветку (при переключении на ветку, которой ещё нет в локальном репозитории, она будет создана и связана с удалённой)
git push origin имя_ветки	отправка изменений конкретной ветки в центральный репозиторий
git merge no-ff имя_ветки	слияние ветки с текущим деревом
git branch -d имя_ветки	удаление локальной уже слитой с основным деревом ветки
git branch -D имя_ветки	принудительное удаление локальной ветки
git push origin :имя_ветки	удаление ветки с центрального репозитория

Таблица 3.1 Описание некоторых команд системы контроля версий Git.

4. Выполнение лабораторной работы

4.1 Техническое обеспечение

Лабораторная работа была выполнена на домашнем компьютере под управлением операционной системы macOS.

4.2 Базовая настройка Git

Для начала я проведу предварительную конфигурацию Git, для этого открываю терминал и ввожу команды на (рис. 4.2.1)

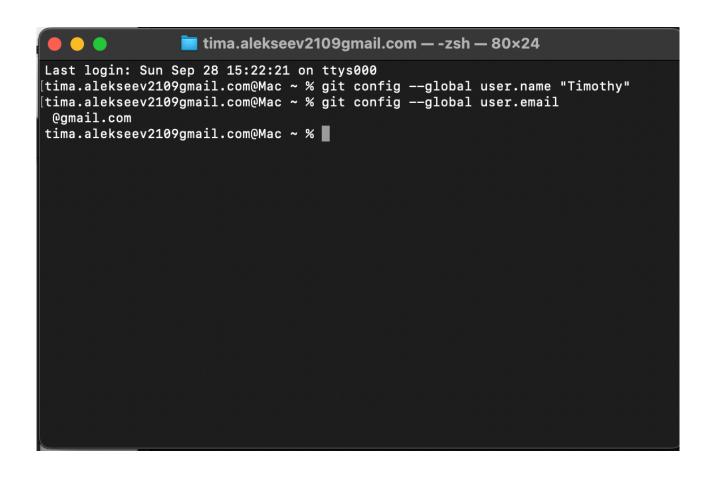


Рис. 4.2.1 Предварительная конфигурация Git.

Далее настраиваю параметры utf-8, имя начальной ветки, autocrlf и safecrlf (рис. 4.2.2)

```
Last login: Sun Sep 28 15:22:21 on ttys000

[tima.alekseev2109gmail.com@Mac ~ % git config --global user.name "Timothy"

[tima.alekseev2109gmail.com@Mac ~ % git config --global user.email

@gmail.com

[tima.alekseev2109gmail.com@Mac ~ % git config --global core.quotepath false

tima.alekseev2109gmail.com@Mac ~ % git config --global init.defaultBranch master

[tima.alekseev2109gmail.com@Mac ~ % git config --global core.autocrlf input

[tima.alekseev2109gmail.com@Mac ~ % git config --global core.safecrlf warn

tima.alekseev2109gmail.com@Mac ~ % git config --global core.safecrlf warn

tima.alekseev2109gmail.com@Mac ~ %
```

Рис. 4.2.2 Настройка параметров Git.

Далее создаю пару ssh ключей для интеграции с платформой GitHub (рис. 4.2.3)

```
tima.alekseev2109gmail.com — -zsh — 98×24
Generating public/private ed25519 key pair.
Enter file in which to save the key (/Users/tima.alekseev2109gmail.com/.ssh/id_ed25519):
/Users/tima.alekseev2109gmail.com/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase for "/Users/tima.alekseev2109gmail.com/.ssh/id_ed25519" (empty for no passphrase)
Enter same passphrase again:
Your identification has been saved in /Users/tima.alekseev2109gmail.com/.ssh/id ed25519
Your public key has been saved in /Users/tima.alekseev2109gmail.com/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:IDS5UzzcXttpP8lfq9bIakhFjtMfX0F14Z5uqsbEmWw tima.alekseev2109gmail.com@Mac.lan
The key's randomart image is:
 --[ED25519 256]--+
    0+ .
     .0.0 .=0 . ..
     0. ..0.++.. 0
         S = .+oo+.
          . E .*..
           = . 0=0
          . + +00.
           00+0.
    -[SHA256]-
tima.alekseev2109gmail.com@Mac ~ %
```

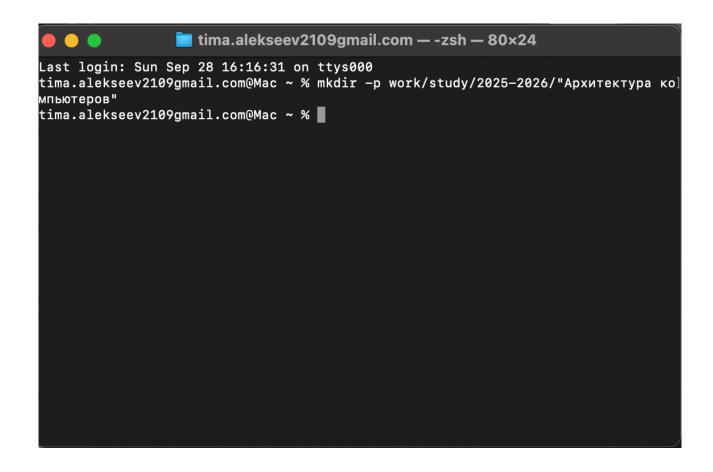
Рис. 4.2.3 Создание пары ssh ключей.

Далее я перехожу на сайт GitHub, авторизуюсь, перехожу в настройки аккаунта, вставляю публичный ключ в предназначенном для этого поле. (рис. 4.2.4)



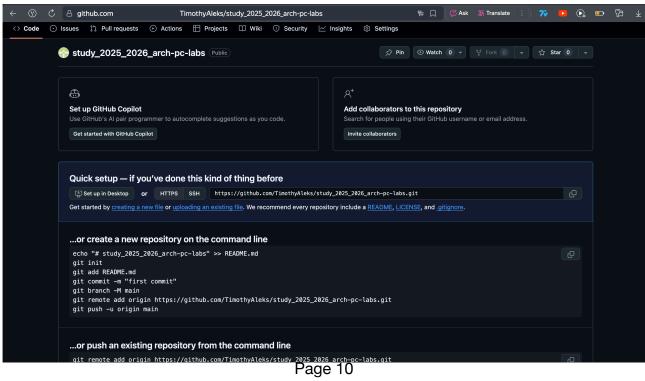
4.3 Создание рабочего пространства и репозитория курса на основе шаблона.

Рабочее пространство при выполнении лабораторных работ должно придерживаться определённой структурной иерархии, для этого я создаю директорию на своем рабочем компьютере (рис. 4.3.1)



4.4 Создание репозитория курса на основе шаблона.

Создаю репозиторий на основе имеющего шаблона (рис. 4.4.1) через функционал клонирования интерфейса GitHub. (рис 4.4.2)



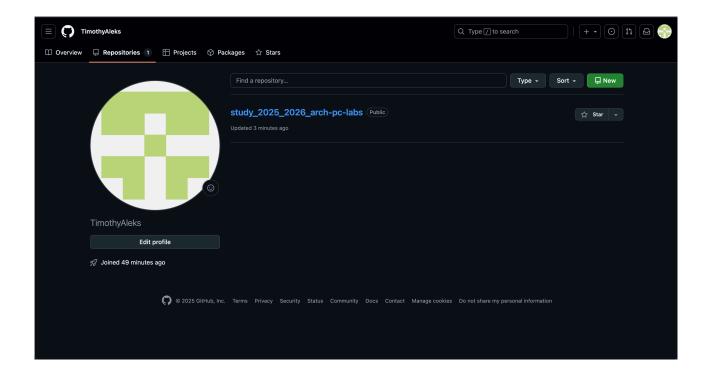
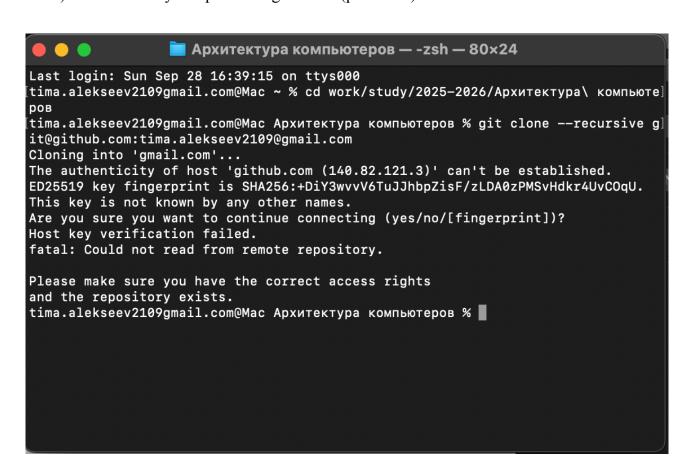


Рис. 4.4.1 Процесс клонирования репозитория через интерфейс GitHub.

Рис. 4.4.2 Шаблон для клонирования на платформе GitHub.

Сгенерированный репозиторий на основе шаблона клонирую на свой рабочий компьютер, для этого беру ссылку для клонирования через интерфейс GitHub (рис. 4.4.1) и затем ввожу в терминале git clone. (рис 4.4.3)



5. Задания для самостоятельно работы

Через терминал отправляю предыдущий отчет по лабораторной работе на свой удаленный репозиторий в GitHub (рис. 5.1), затем проверяю изменения на самом GitHub. (рис 5.2)

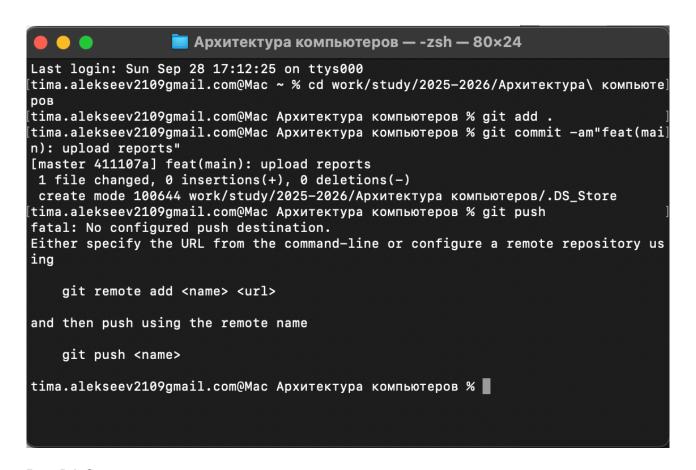


Рис 5.1 Отправка изменений на удаленный репозиторий.

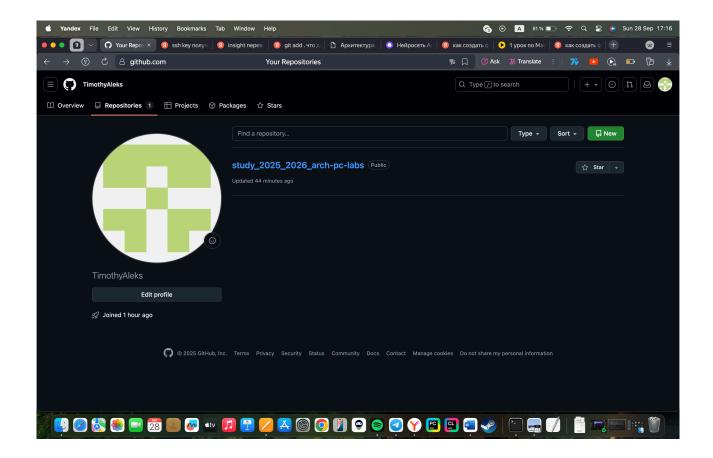


Рис 5.2 Проверка выполненных изменений на GitHub

6. Выводы

При выполнении данной лабораторной работы я изучил идеологию и применение средств контроля версий, а также приобрел практические навыки по работе с реализацией VSC git.

Список литературы

- 1.<u>https://esystem.rudn.ru/pluginfile.php/2945866/mod_resource/content/0/lab2.pdf</u>
- 2.<u>https://esystem.rudn.ru/mod/page/view.php?</u> id=1030466
- 3.https://esystem.rudn.ru/pluginfile.php/2088961/mod_resource/content/2/

Лекция_2_Многоуровневая%20компьютерная%20о рганизация_2018_ТУИС.pdf