

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2.

Система контроля версий Git

Замула Егор Сергеевич

Содержание

Цель работы	5
Задание	6
Теоретическое введение	7
Выполнение лабораторной работы	9
Выводы	15
Список литературы	16

Список таблиц

Список иллюстраций

Цель работы

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

Задание

1. Настройка GitHub.
2. Базовая настройка Git.
3. Создание SSH-ключа.
4. Создание рабочего пространства и репозитория курса на основе шаблона.
5. Создание репозитория курса на основе шаблона.
6. Настройка каталога курса.
7. Выполнение заданий для самостоятельной работ

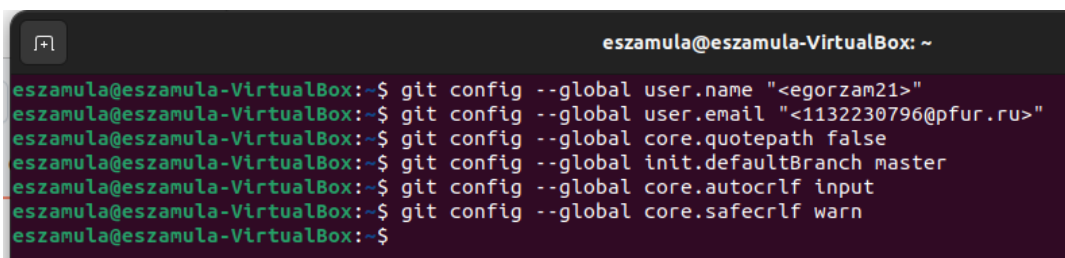
Теоретическое введение

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 пользовате-

лям получить рабочую копию или препятствует изменению рабочей копии файла средствами файловой системы ОС, обеспечивая таким образом, привилегированный доступ только одному пользователю, работающему с файлом. 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.

Выполнение лабораторной работы

1. Создаю учётную запись на сайте <https://github.com/> и заполните основные данные.
2. Открыл терминал и ввел следующие команды, указав своё имя и email, далее настроил utf-8 в выводе сообщений git. Задал имя начальной ветки (будем называть её master). Также задал параметр autocrlf и параметру safecrlf.



```
eszamula@eszamula-VirtualBox: ~  
eszamula@eszamula-VirtualBox:~$ git config --global user.name "<egorzam21>"  
eszamula@eszamula-VirtualBox:~$ git config --global user.email "<1132230796@pfur.ru>"  
eszamula@eszamula-VirtualBox:~$ git config --global core.quotePath false  
eszamula@eszamula-VirtualBox:~$ git config --global init.defaultBranch master  
eszamula@eszamula-VirtualBox:~$ git config --global core.autocrlf input  
eszamula@eszamula-VirtualBox:~$ git config --global core.safecrlf warn  
eszamula@eszamula-VirtualBox:~$
```

3. Создание SSH-ключа. Для последующей идентификации пользователя на сервере репозитория необходимо сгенерировать пару ключей (приватный и открытый). Для этого есть команда `ssh-keygen -C "Имя Фамилия work@mail"`. Ключи сохраняться в каталоге `~/.ssh/`.


```

eszamula@eszamula-VirtualBox:~$ ssh-keygen -C "egorzam21 <1132230796@pfur.ru>"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/eszamula/.ssh/id_rsa): KEYS
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in KEYS
Your public key has been saved in KEYS.pub
The key fingerprint is:
SHA256:geVtYPFyUu+ev+mW83hxEeemqgcu0LOQVyuUNWnrny0Bk egorzam21 <1132230796@pfur.ru>
The key's randomart image is:
+----[RSA 3072]-----+
|      =.++      |
|      =Bo.o    ..|
|      . * =+.   .o|
|      E*o.     .o|
|      oSB .    o.|
|      + B o.    ....|
|      + = .o.   .o|
|      o . o.+o.|
|      ..o o*=.  |
+----[SHA256]-----+
eszamula@eszamula-VirtualBox:~$

```

4. Далее я скопировал ключ из файла KYES (также можно было использовать хclip) и вставил ключ в появившееся на сайте поле и указал для ключа имя

You have successfully added the key 'ZAM'. ✕


egorzam21 (egorzam21)
 Your personal account

[Go to your personal profile](#)


- Public profile
- Account
- Appearance
- Accessibility
- Notifications
- Access
 - Billing and plans
 - Emails
 - Password and authentication
 - Sessions
 - SSH and GPG keys**
 - Organizations
 - Enterprises
 - Moderation

SSH keys

[New SSH key](#)

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

Authentication Keys

 ZAM SHA256:geVtYPFyUu+ev+mW83hxEeemqgcu0LOQVyuUNWnrny0Bk Added on Oct 12, 2023 Never used — Read/write	Delete
--	------------------------

[Check out our guide to generating SSH keys or troubleshoot common SSH problems.](#)

GPG keys

[New GPG key](#)

There are no GPG keys associated with your account.

[Learn how to generate a GPG key and add it to your account.](#)

5. Создание рабочего пространства. Открыл терминал и создайте каталог для предмета «Архитектура компьютера»

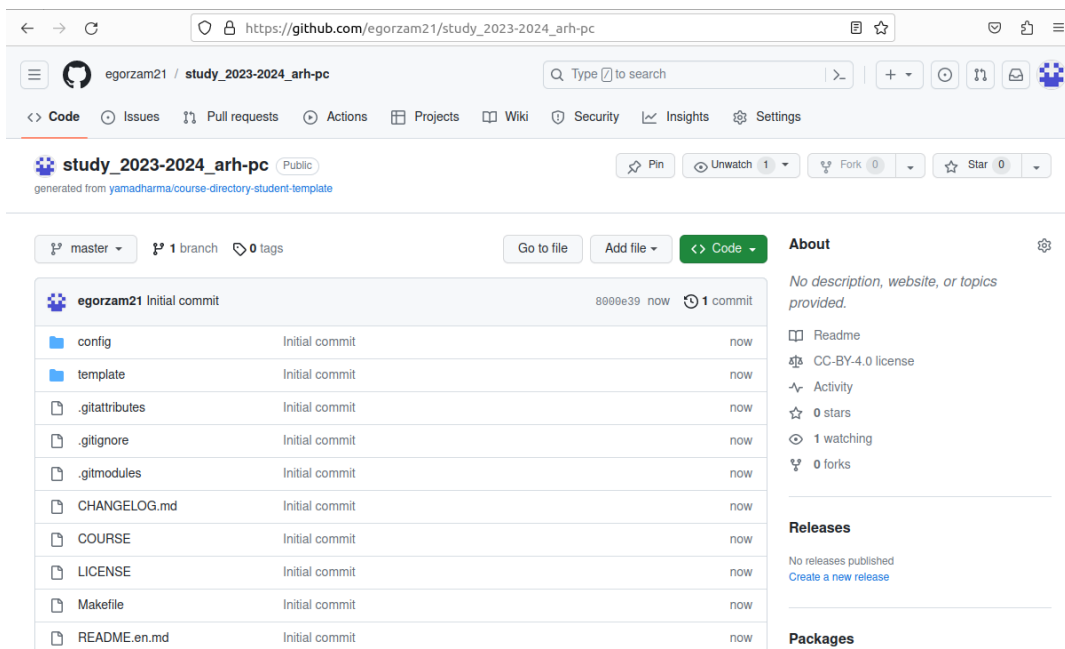
```

eszamula@eszamula-VirtualBox:~$ mkdir -p ~/work/study/2023-2024/"Архитектура компьютеров"
eszamula@eszamula-VirtualBox:~$

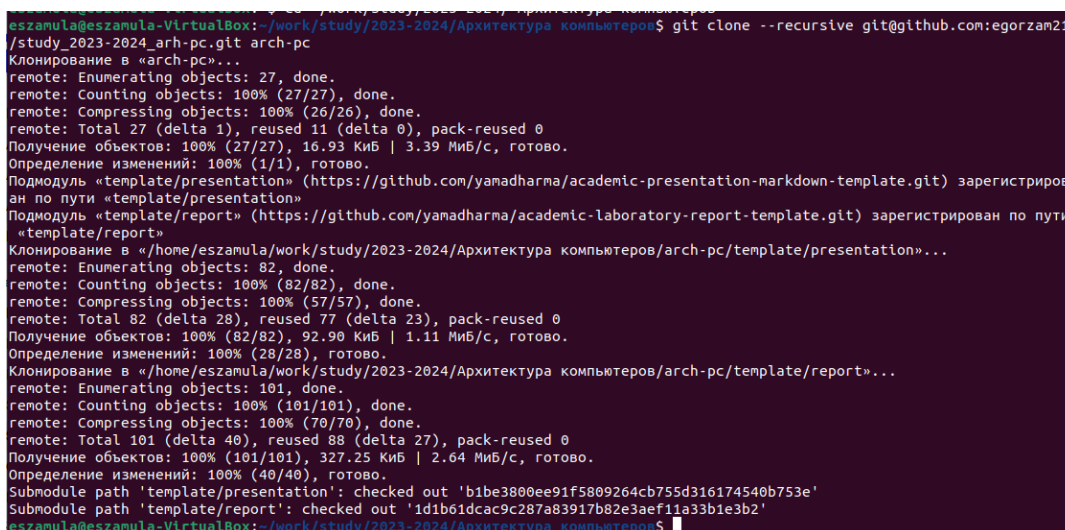
```

6. Создание репозитория курса.Репозиторий на основе шаблона можно создать через web-интерфейс github.Перешёл на станицу репозитория с шаблоном

курса <https://github.com/yamadharma/course-directory-student-template>. Далее выбрал Use this template. В открывшемся окне задайте имя репозитория (Repository name) study_2023-2024_arh-pc и создайте репозиторий (кнопка Create repository from template).



- Открыл терминал и перейдите в каталог курса. Клонировал созданный репозиторий (Ссылку для клонирования можно скопировать на странице созданного репозитория Code -> SSH)



- Настройка каталога курса. Перешёл в каталог курса, удалил все лишние файлы и

создали необходимые каталоги

```
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров$ cd ~/work/study/2023-2024/Архитектура компьютеров/arch-h-pc
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-h-pc$ rm package.json
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-h-pc$ echo arch-pc > COURSE
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-h-pc$ make
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-h-pc$
```

9. Далее я отправляю файлы на сервер

```
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc$ git add .
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc$ git commit -am 'feat(main):course structure'
[master b23af2b] feat(main):course structure
199 files changed, 54725 insertions(+), 14 deletions(-)
create mode 100644 labs/README.md
create mode 100644 labs/README.ru.md
create mode 100644 labs/lab01/presentation/Makefile
create mode 100644 labs/lab01/presentation/image/kulyabov.jpg
create mode 100644 labs/lab01/presentation/presentation.md
create mode 100644 labs/lab01/report/Makefile
create mode 100644 labs/lab01/report/bib/cite.bib
create mode 100644 labs/lab01/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab01/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_fignos.py
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_secnos.py
create mode 100755 labs/lab01/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/core.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/main.py
create mode 100644 labs/lab01/report/pandoc/filters/pandocxnos/pandocattributes.py
create mode 100644 labs/lab01/report/report.md
create mode 100644 labs/lab02/presentation/Makefile
create mode 100644 labs/lab02/presentation/image/kulyabov.jpg
create mode 100644 labs/lab02/presentation/presentation.md
create mode 100644 labs/lab02/report/Makefile
create mode 100644 labs/lab02/report/bib/cite.bib
create mode 100644 labs/lab02/report/image/placeimg_800_600_tech.jpg
create mode 100644 labs/lab02/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_eqnos.py
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_fignos.py
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_secnos.py
create mode 100755 labs/lab02/report/pandoc/filters/pandoc_tablenos.py
create mode 100644 labs/lab02/report/pandoc/filters/pandocxnos/__init__.py
create mode 100644 labs/lab02/report/pandoc/filters/pandocxnos/core.py
```

10. Проверяю правильность создания иерархии рабочего пространства в локальном репозитории и на странице github.

The screenshot shows a web browser window with the URL https://github.com/egorzam21/study_2023-2024_arh-pc/tree/master/labs. The page displays the file structure of the repository. On the left, a sidebar shows the file explorer with folders like 'config', 'labs', and 'presentation'. The 'labs' folder is expanded, showing subfolders 'lab01' through 'lab11'. The main content area shows a table of files and folders with their commit messages and dates. The commit message for all files is 'feat(main):course structure' and the date is '1 minute ago'.

Name	Last commit message	Last commit date
..		
lab01	feat(main):course structure	1 minute ago
lab02	feat(main):course structure	1 minute ago
lab03	feat(main):course structure	1 minute ago
lab04	feat(main):course structure	1 minute ago
lab05	feat(main):course structure	1 minute ago
lab06	feat(main):course structure	1 minute ago
lab07	feat(main):course structure	1 minute ago
lab08	feat(main):course structure	1 minute ago
lab09	feat(main):course structure	1 minute ago
lab10	feat(main):course structure	1 minute ago
lab11	feat(main):course structure	1 minute ago
README.md	feat(main):course structure	1 minute ago
README.ru.md	feat(main):course structure	1 minute ago

11. Выполнение заданий для самостоятельной работы. Для начала я перехожу в директорию labs/lab02/report с помощью cd. С помощью touch создал файл для отчета по второй лабораторной работе. Отчёт я буду делать в libreoffice writer

```
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc$ cd labs/lab02/report
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc/labs/lab02/report$ touch L02_Zamula_Report
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc/labs/lab02/report$
```

12. Копирую отчёт по первой лабораторной в labs/lab01/report с помощью cp и проверяю наличие

```
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc/labs/lab02/report$ cd ..
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc/labs/lab02$ cd ..
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc/labs$ cd lab01
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc/labs/lab01$ cd report
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc/labs/lab01/report$
```

13. Далее нужно добавить файлы на сервер и сохраняю изменения на сервере. Делаю точно такие же действия, только уже с отчётом по второй лабораторной работе (перехожу в директорию labs/lab02/report и отправка на сервер). Отправляю в центральный репозиторий сохраненные изменения командой git push -f origin master

```
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc/labs/lab02/report$ git push -f origin master
Перечисление объектов: 17, готово.
Подсчет объектов: 100% (15/15), готово.
При сжатии изменений используется до 3 потоков
Сжатие объектов: 100% (11/11), готово.
Запись объектов: 100% (11/11), 1.52 Миб | 2.04 Миб/с, готово.
Всего 11 (изменений 5), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (5/5), completed with 3 local objects.
To github.com:egorzan21/study_2023-2024_arh-pc.git
 b23af2b..bf18b6a master -> master
eszanula@eszanula-VirtualBox: ~/work/study/2023-2024/Архитектура компьютеров/arch-pc/labs/lab02/report$
```

14. Теперь необходимо убедиться, что наши отчёты пришли на репозиторий, как можно заметить время lab01 and lab02 кардинально отличается от остальных папок

← → ↺

https://github.com/egorzam21/study_2023-2024_arh-pc/tree/master/labs

☆

🔍 📄 📁 📄 📄

☰

egorzam21 / study_2023-2024_arh-pc

🔍 Type to search

> + 🔍 ⌂ 📄 📄

<> Code

🕒 Issues

🔗 Pull requests

🔄 Actions

📁 Projects

📖 Wiki

🛡 Security

📊 Insights

⚙ Settings

📁 Files

👤 master + 🔍

🔍 Go to file t

> 📁 config

▼ 📁 labs

> 📁 lab01

> 📁 lab02

> 📁 lab03

> 📁 lab04

> 📁 lab05

> 📁 lab06

> 📁 lab07

> 📁 lab08

> 📁 lab09

> 📁 lab10

> 📁 lab11

📄 README.md

study_2023-2024_arh-pc / labs /

Add file ...

🔗 egorzam21 Add files bf18b6a · 3 minutes ago 🕒 History

Name	Last commit message	Last commit date
📁 ..		
📁 lab01	Add files	5 minutes ago
📁 lab02	Add files	3 minutes ago
📁 lab03	feat(main):course structure	41 minutes ago
📁 lab04	feat(main):course structure	41 minutes ago
📁 lab05	feat(main):course structure	41 minutes ago
📁 lab06	feat(main):course structure	41 minutes ago
📁 lab07	feat(main):course structure	41 minutes ago
📁 lab08	feat(main):course structure	41 minutes ago
📁 lab09	feat(main):course structure	41 minutes ago

Выводы

В процессе выполнения этого лабораторного исследования я ознакомился с принципами и методами использования систем контроля изменений, а также улучшил свои практические навыки по использованию системы git

Список литературы