

# **Лабораторная работа-03**

**Контроль версий git**

**Бровкин Александр НБИбд-01-21**

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Выполнение лабораторной работы</b>	<b>6</b>
<b>3 Вывод</b>	<b>16</b>
<b>4 Контрольные вопросы</b>	<b>17</b>
<b>Список литературы</b>	<b>22</b>

# Список иллюстраций

2.1	Аккаунт на гитхаб . . . . .	6
2.2	Параметры репозитория . . . . .	7
2.3	Создаю ключи . . . . .	8
2.4	Создаю ключи . . . . .	9
2.5	Загружаю ключи . . . . .	10
2.6	Загрузка ключей . . . . .	10
2.7	Параметры репозитория . . . . .	11
2.8	Параметры репозитория . . . . .	12
2.9	Связь репозитория с аккаунтом . . . . .	13
2.10	Связь репозитория с аккаунтом . . . . .	14
2.11	Загрузка шаблона . . . . .	14
2.12	Загрузка шаблона . . . . .	14
2.13	Загрузка шаблона . . . . .	15
2.14	Загрузка шаблона . . . . .	15
2.15	Первый коммит . . . . .	15

# **Список таблиц**

# **1 Цель работы**

Целью данной работы является изучение идеологии и применения средств контроля версий и освоение умений работать с git.

## 2 Выполнение лабораторной работы

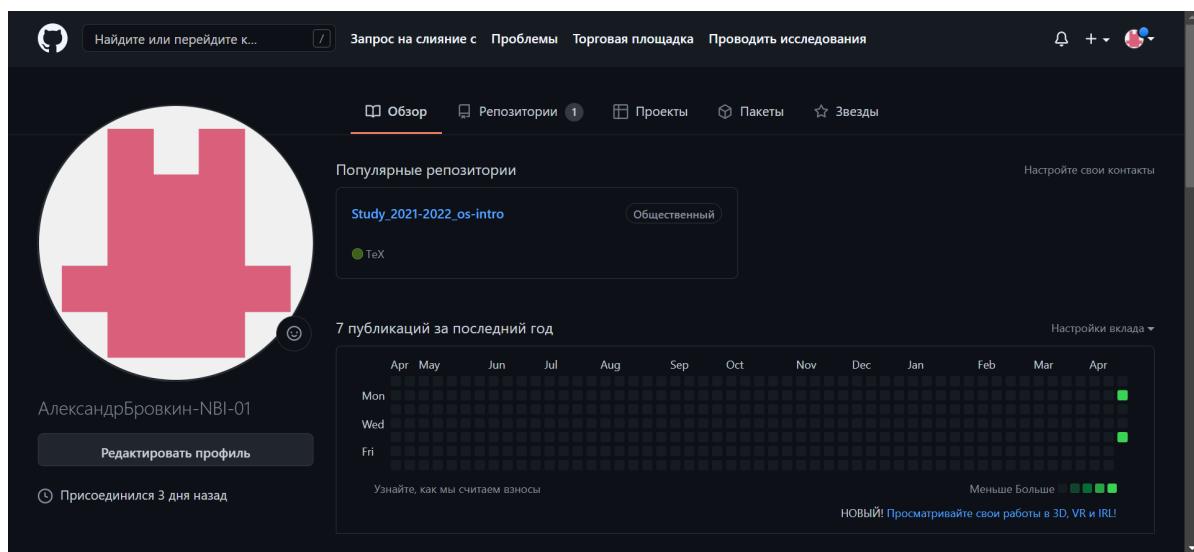


Рис. 2.1: Аккаунт на гитхаб

Зададим имя и email владельца репозитория, кодировку и прочие параметры.  
Создаю аккаунт на github.

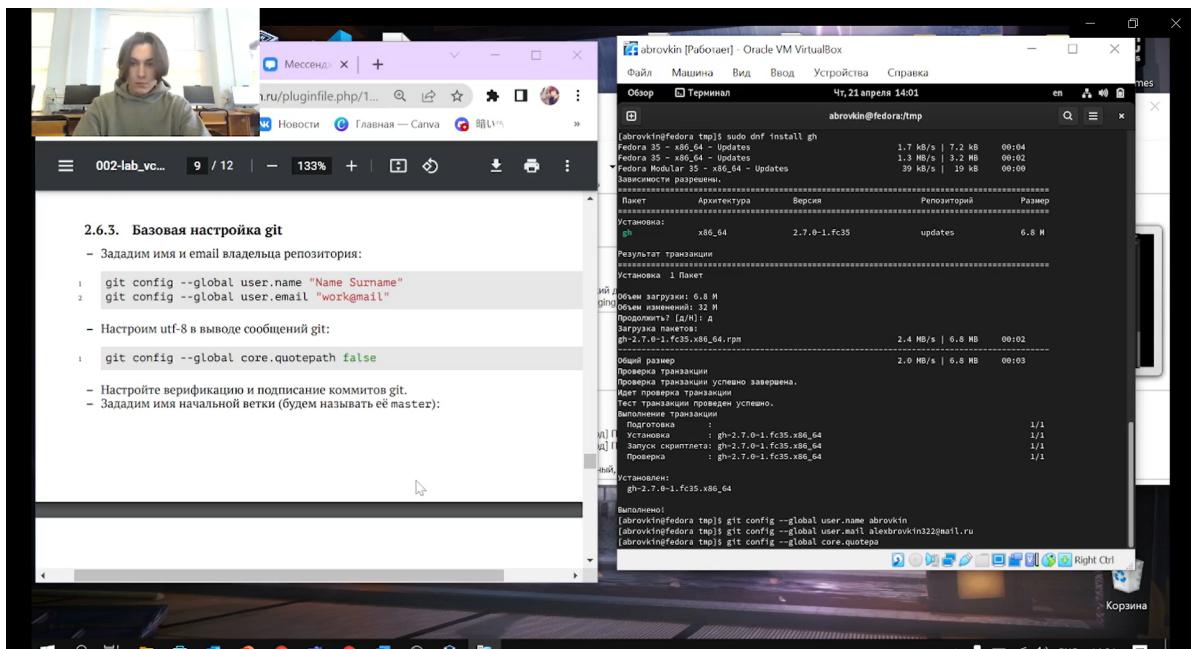


Рис. 2.2: Параметры репозитория

## Создаем ключи

Обзор Терминал Чт, 21 апреля 14:25 abrovkin@fedora:/tmp

```
pub rsa4096 2022-04-21 [SC]
 375116D34841622A559A09FA96BDC05786F177C6
uid           alexander <alexbrovkin322@mail.ru>
sub rsa4096 2022-04-21 [E]

[abrovkin@fedora tmp]$ gpg --list-secret-keys --keyid-format LONG
gpg: проверка таблицы доверия
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: глубина: 0 доверенных: 1 подписанных: 0 доверие: 0-, 0q, 0n, 0m, 0f, 1u
/home/abrovkin/.gnupg/pubring.kbx

sec rsa4096/96BDC05786F177C6 2022-04-21 [SC]
 375116D34841622A559A09FA96BDC05786F177C6
uid           [ абсолютно ] alexander <alexbrovkin322@mail.ru>
ssb rsa4096/992F63305E659972 2022-04-21 [E]

[abrovkin@fedora tmp]$ ^C
[abrovkin@fedora tmp]$ gpg --armor --export ^C
[abrovkin@fedora tmp]$ gpg --armor --export 96BDC05786F177C6 | xclip -sel clip
bash: xclip: command not found...
Install package 'xclip' to provide command 'xclip'? [N/y] y

* Waiting in queue...
The following packages have to be installed:
xclip-0.13-15.git11c6a61.fc35.x86_64  Command line clipboard grabber
Proceed with changes? [N/y] y

* Waiting in queue...
* Waiting for authentication...
* Waiting in queue...
* Downloading packages...
* Requesting data...
* Testing changes...
* Installing packages...

[abrovkin@fedora tmp]$ gpg --armor --export 96BDC05786F177C6 | xclip -sel clip
[abrovkin@fedora tmp]$
```

Рис. 2.3: Создаю ключи

Обзор Терминал Чт, 21 апреля 14:08 abrovkin@fedora:/tmp

```
Enter same passphrase again:  
Your identification has been saved in /home/abrovkin/.ssh/id_rsa  
Your public key has been saved in /home/abrovkin/.ssh/id_rsa.pub  
The key fingerprint is:  
SHA256:W3g31V9Q500KsJjgaU6x82cu/o2tzJsWL9qqKiJXnI4 abrovkin@fedora  
The key's randomart image is:  
----[RSA 4096]----+  
+ o . . . +|  
. X o . . *o|  
B = . . o |=|  
= o . . o |  
o S o o . |  
. ..B . . |  
o .E +oo |  
+ . o. +oo. |  
[oo.oo.o+Bo |  
----[SHA256]----+  
[abrovkin@fedora tmp]$ ssh-keygen -t ed25519  
Generating public/private ed25519 key pair.  
Enter file in which to save the key (/home/abrovkin/.ssh/id_ed25519):  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /home/abrovkin/.ssh/id_ed25519  
Your public key has been saved in /home/abrovkin/.ssh/id_ed25519.pub  
The key fingerprint is:  
SHA256:ptmdtEwUsbwZIh8vVt8JV9x3ee0m0hgvmoz503hGDFQ abrovkin@fedora  
The key's randomart image is:  
----[ED25519 256]--+  
.E. .=|  
.. o oB|  
..o * . .+|  
o.*.= +..o|  
Sox++..oo |  
* *=o+ |  
o+.*=. . |  
o * + |  
..+ |  
----[SHA256]----+  
[abrovkin@fedora tmp]$ gpg --full-generate-key
```

Рис. 2.4: Создаю ключи

The screenshot shows the GitHub account settings for 'AlexanderBrovkin-NBI-01'. On the left, a sidebar lists various account management options like Public profile, Account, Appearance, Accessibility, Notifications, Access, Billing and plans, Emails, Password and authentication, and SSH and GPG keys (which is currently selected). The main content area has two sections: 'SSH keys' and 'GPG keys'. Both sections indicate that no keys are associated with the account. There are 'New SSH key' and 'New GPG key' buttons respectively. Below these sections is a 'Vigilant mode' section with a checkbox for 'Flag unsigned commits as unverified', which includes a note about existing unsigned commits and a link to learn more.

Рис. 2.5: Загружаю ключи

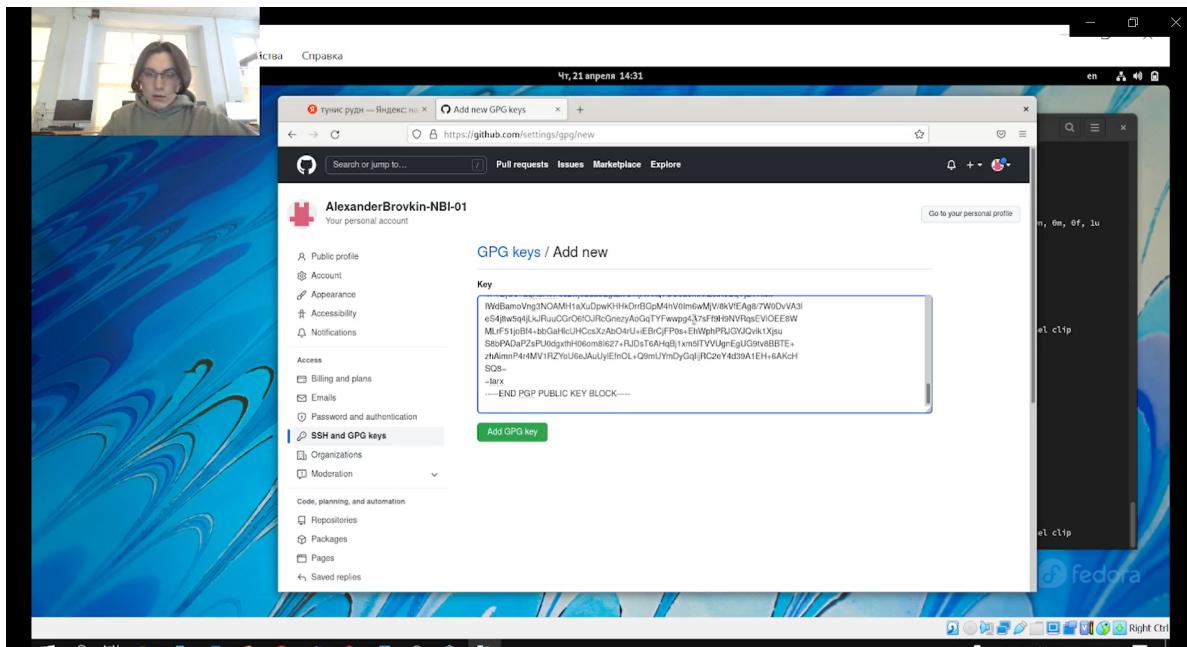


Рис. 2.6: Загрузка ключей

Настройка автоматических подписей коммитов git

```
abrovkin [Работает] - Oracle VM VirtualBox
Файл Машина Вид Ввод Устройства Справка
Обзор Терминал Чт, 21 апреля 14:36
en 🔍 ⌂ ✎
+ abrovkin@fedora:/tmp
375116D34B41622A559A09FA96BDC05786F177C6
uid alexander <alexbrovkin322@mail.ru>
sub rsa4096 2022-04-21 [E]

[abrovkin@fedora tmp]$ gpg --list-secret-keys --keyid-format LONG
gpg: проверка таблицы доверия
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: глубина: 0 достоверных: 1 подписанных: 0 доверие: 0-, 0q, 0n, 0m, 0f, 1u
/home/abrovkin/.gnupg/pubring.kbx
-----
sec rsa4096/96BDC05786F177C6 2022-04-21 [SC]
375116D34B41622A559A09FA96BDC05786F177C6
uid [ абсолютно ] alexander <alexbrovkin322@mail.ru>
ssb rsa4096/992F63305E659972 2022-04-21 [E]

[abrovkin@fedora tmp]$ ^C
[abrovkin@fedora tmp]$ gpg --armor --export ^C
[abrovkin@fedora tmp]$ gpg --armor --export 96BDC05786F177C6 | xclip -sel clip
bash: xclip: command not found...
Install package 'xclip' to provide command 'xclip'? [N/y] y

* Waiting in queue...
The following packages have to be installed:
xclip-0.13-15.git11cba61.fc35.x86_64 Command line clipboard grabber
Proceed with changes? [N/y] y

* Waiting in queue...
* Waiting for authentication...
* Waiting in queue...
* Downloading packages...
* Requesting data...
* Testing changes...
* Installing packages...

[abrovkin@fedora tmp]$ gpg --armor --export 96BDC05786F177C6 | xclip -sel clip
[abrovkin@fedora tmp]$ git config --global user.signingkey 96BDC05786F177C6
[abrovkin@fedora tmp]$ git config --commit
```

Рис. 2.7: Параметры репозитория

abrovkin [Работает] - Oracle VM VirtualBox

Файл Машина Вид Ввод Устройства Справка

Обзор Терминал Чт, 21 апреля 14:42 en

abrovkin@fedora:/tmp — gh auth login

```
--rename-section      переименовать раздел: старое-имя новое-имя
--remove-section     удалить раздел: имя
-l, --list           показать весь список
--fixed-value        use string equality when comparing values to 'value-pattern'
-e, --edit           открыть в редакторе
--get-color          найти настроенный цвет: раздел [по-умолчанию]
--get-colorbool      проверить, существует ли настроенный цвет: раздел [stdout-есть-tty]

Тип
-t, --type <type>   value is given this type
--bool                значение – это «true» (правда) или «false» (ложь)
--int                 значение – это десятичное число
--bool-or-int         значение – это --bool или --int
--bool-or-str         value is --bool or string
--path                значение – это путь (к файлу или каталогу)
--expiry-date        значение – это дата окончания срока действия

другое
-z, --null            завершать значения НУЛЕВЫМ байтом
--name-only          показывать только имена переменных
--includes           учитывать директивы include (включения файлов) при запросе
--show-origin         показать источник настройки (файл, стандартный ввод, двоичный объект, командная строка)
--show-scope          show scope of config (worktree, local, global, system, command)
--default <value>     with --get, use default value when missing entry

[abrovkin@fedora tmp]$ git config gpg.program $(which gpg2)
fatal: not in a git directory
[abrovkin@fedora tmp]$ git config --global gpg.program $(which gpg2)
[abrovkin@fedora tmp]$ git config --global commit.gpgsign true
[abrovkin@fedora tmp]$ git config --global gpg.program $(which gpg2)
[abrovkin@fedora tmp]$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? SSH
? Upload your SSH public key to your GitHub account? Skip
? How would you like to authenticate GitHub CLI? Login with a web browser

! First copy your one-time code: 3E14-A0F7
Press Enter to open github.com in your browser...
```

Рис. 2.8: Параметры репозитория

## Настройка gh

abrovkin [Работает] - Oracle VM VirtualBox

Файл Машина Вид Ввод Устройства Справка

Обзор Терминал Чт, 21 апреля 14:42 en ⚡ 🔍 🌐 🛡️ 🗑️

```
abrovkin@fedora:/tmp — gh auth login

--rename-section    переименовать раздел: старое-имя новое-имя
--remove-section   удалить раздел: имя
-l, --list          показать весь список
--fixed-value       use string equality when comparing values to 'value-pattern'
-e, --edit          открыть в редакторе
--get-color         найти настроенный цвет: раздел [по-умолчанию]
--get-colorbool     проверить, существует ли настроенный цвет: раздел [stdout-есть-tty]

Тип
-t, --type <>      value is given this type
--bool               значение – это «true» (правда) или «false» (ложь)
--int                значение – это десятичное число
--bool-or-int        значение – это --bool или --int
--bool-or-str        value is --bool or string
--path               значение – это путь (к файлу или каталогу)
--expiry-date        значение – это дата окончания срока действия

Другое
-z, --null           завершать значения НУЛЕВЫМ байтом
--name-only          показывать только имена переменных
--includes           учитывать директивы include (включения файлов) при запросе
--show-origin         показать источник настройки (файл, стандартный ввод, двоичный объект, командная строка)
--show-scope          show scope of config (worktree, local, global, system, command)
--default <value>    with --get, use default value when missing entry

[abrovkin@fedora tmp]$ git config gpg.program $(which gpg2)
fatal: not in a git directory
[abrovkin@fedora tmp]$ git config --global gpg.program $(which gpg2)
[abrovkin@fedora tmp]$ git config --global commit.gpgsign true
[abrovkin@fedora tmp]$ git config --global gpg.program $(which gpg2)
[abrovkin@fedora tmp]$ gh auth login
? What account do you want to log into? GitHub.com
? What is your preferred protocol for Git operations? SSH
? Upload your SSH public key to your GitHub account? Skip
? How would you like to authenticate GitHub CLI? Login with a web browser

First copy your one-time code: 3E14-A0F7
Press Enter to open github.com in your browser...
```

Рис. 2.9: Связь репозитория с аккаунтом

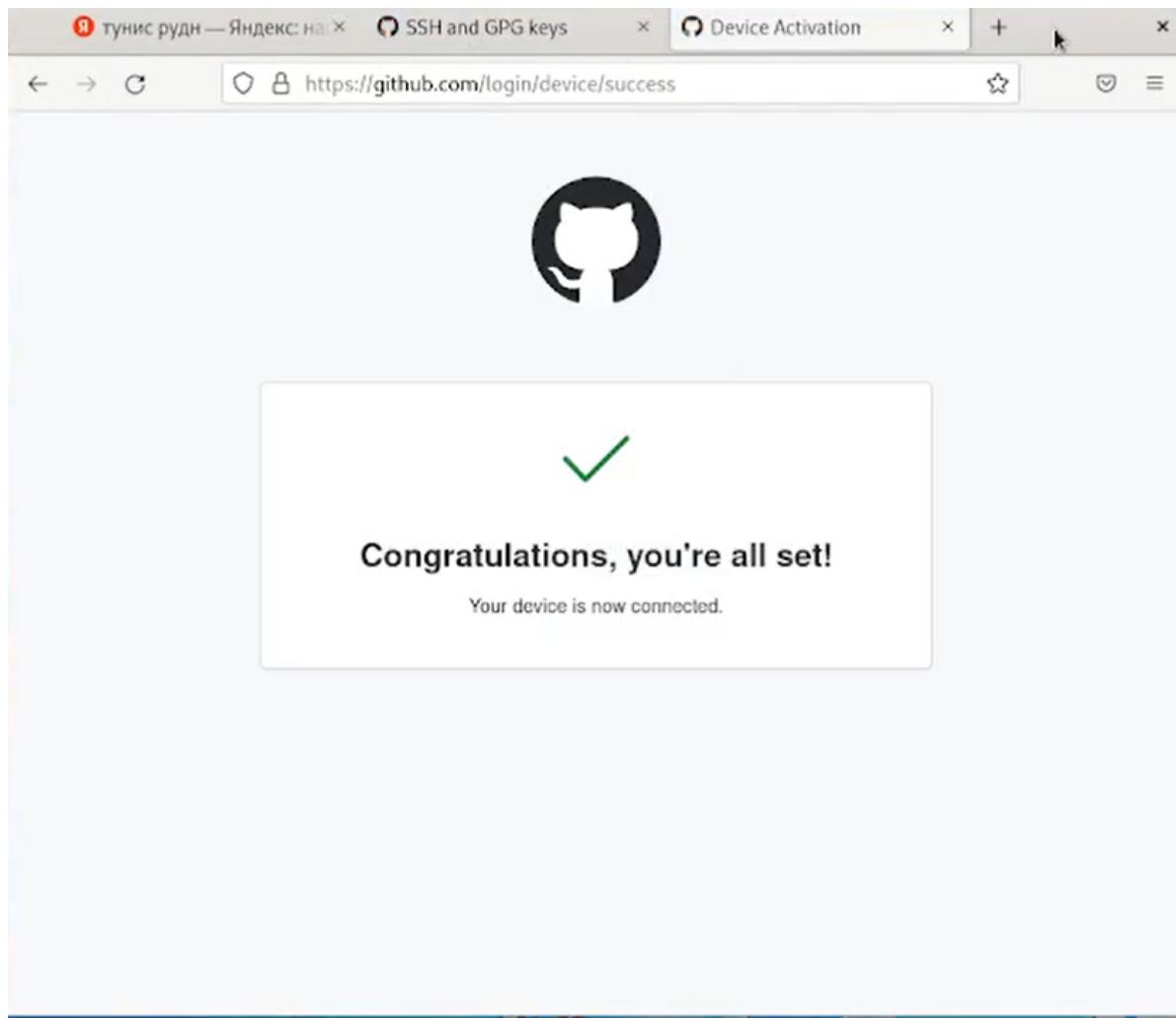


Рис. 2.10: Связь репозитория с аккаунтом

Загрузка шаблона репозитория и синхронизация

```
[abrovkin@fedora tmp]$ mkdir -p ~/work/study/2021-2022/"Операционные системы"
```

Рис. 2.11: Загрузка шаблона

```
[abrovkin@fedora tmp]$ cd ~/work/study/2021-2022/"Операционные системы"  
[abrovkin@fedora Операционные системы]$ gh repo create study_2021-2022_os-intro  
--template=yamadharma/course-directory-student-template --public  
Welcome to GitHub CLI!
```

Рис. 2.12: Загрузка шаблона

```
Logged in as AlexanderBrovkin-NBI-01
[abrovkin@fedora Операционные системы]$ git clone --recursive git@github.com:AlexanderBrovkin-NBI-01/study_2021-2022_os-intro.git os-intro
Клонирование в «os-intro»...
```

Рис. 2.13: Загрузка шаблона

```
abrovkin@fedora:~/work/study/2021-2022/Операционные системы/os-intro
create mode 100644 project-personal/stage2/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100644 project-personal/stage2/report/report.md
create mode 100644 project-personal/stage3/presentation/Makefile
create mode 100644 project-personal/stage3/presentation/presentation.md
create mode 100644 project-personal/stage3/report/Makefile
create mode 100644 project-personal/stage3/report/bib/cite.bib
create mode 100644 project-personal/stage3/report/image/placeimg_800_600_tech.jpg
create mode 100644 project-personal/stage3/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100644 project-personal/stage3/report/report.md
create mode 100644 project-personal/stage4/presentation/Makefile
create mode 100644 project-personal/stage4/presentation/presentation.md
create mode 100644 project-personal/stage4/report/Makefile
create mode 100644 project-personal/stage4/report/bib/cite.bib
create mode 100644 project-personal/stage4/report/image/placeimg_800_600_tech.jpg
create mode 100644 project-personal/stage4/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100644 project-personal/stage4/report/report.md
create mode 100644 project-personal/stage5/presentation/Makefile
create mode 100644 project-personal/stage5/presentation/presentation.md
create mode 100644 project-personal/stage5/report/Makefile
create mode 100644 project-personal/stage5/report/bib/cite.bib
create mode 100644 project-personal/stage5/report/image/placeimg_800_600_tech.jpg
create mode 100644 project-personal/stage5/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100644 project-personal/stage5/report/report.md
create mode 100644 project-personal/stage6/presentation/Makefile
create mode 100644 project-personal/stage6/presentation/presentation.md
create mode 100644 project-personal/stage6/report/Makefile
create mode 100644 project-personal/stage6/report/bib/cite.bib
create mode 100644 project-personal/stage6/report/image/placeimg_800_600_tech.jpg
create mode 100644 project-personal/stage6/report/pandoc/csl/gost-r-7-0-5-2008-numeric.csl
create mode 100644 project-personal/stage6/report/report.md
create mode 100644 structure
[abrovkin@fedora os-intro]$ git push
Перечисление объектов: 20, готово.
Подсчет объектов: 100% (20/20), готово.
Сжатие объектов: 100% (16/16), готово.
Запись объектов: 100% (19/19), 265.87 КиБ | 2.03 МиБ/с, готово.
Всего 19 (изменений 2), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To github.com:AlexanderBrovkin-NBI-01/study_2021-2022_os-intro.git
 afeb362..97836b3 master -> master
[abrovkin@fedora os-intro]$
```

Рис. 2.14: Загрузка шаблона

## Подготовка репозитория и коммит изменений

```
git add .
git commit -am 'feat(main): make course structure'
git push
```

Рис. 2.15: Первый коммит

## **3 Вывод**

Мы приобрели практические навыки работы с сервисом github.

## 4 Контрольные вопросы

1. Что такое системы контроля версий (VCS) и для решения каких задач они предназначаются?

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

2. Объясните следующие понятия VCS и их отношения: хранилище, commit, история, рабочая копия.

- хранилище - пространство на накопителе где расположен репозиторий
- commit - сохранение состояния хранилища
- история - список изменений хранилища (коммитов)
- рабочая копия - локальная копия сетевого репозитория, в которой работает программист. Текущее состояние файлов проекта, основанное на версии, загруженной из хранилища (обычно на последней)

3. Что представляют собой и чем отличаются централизованные и децентрализованные VCS? Приведите примеры VCS каждого вида.

Централизованные системы контроля версий представляют собой приложения типа клиент-сервер, когда репозиторий проекта существует в единственном экземпляре и хранится на сервере. Доступ к нему осуществлялся через специальное клиентское приложение. В качестве примеров таких программных продуктов можно привести CVS, Subversion.

Распределенные системы контроля версий (Distributed Version Control System, DVCS) позволяют хранить репозиторий (его копию) у каждого разработчика, работающего с данной системой. При этом можно выделить центральный репозиторий (условно), в который будут отправляться изменения из локальных и, с ним же эти локальные репозитории будут синхронизироваться. При работе с такой системой, пользователи периодически синхронизируют свои локальные репозитории с центральным и работают непосредственно со своей локальной копией. После внесения достаточного количества изменений в локальную копию они (изменения) отправляются на сервер. При этом сервер, чаще всего, выбирается условно, т.к. в большинстве DVCS нет такого понятия как “выделенный сервер с центральным репозиторием”.

4. Опишите действия с VCS при единоличной работе с хранилищем.

Один пользователь работает над проектом и по мере необходимости делает коммиты, сохраняя определенные этапы.

5. Опишите порядок работы с общим хранилищем VCS.

Несколько пользователей работают каждый над своей частью проекта. При этом каждый должен работать в своей ветки. При завершении работы ветка пользователя сливаются с основной веткой проекта.

6. Каковы основные задачи, решаемые инструментальным средством git?

- Ведение истории версий проекта: журнал (log), метки (tags), ветвления (branches).

- Работа с изменениями: выявление (diff), слияние (patch, merge).
- Обеспечение совместной работы: получение версии с сервера, загрузка обновлений на сервер.

7. Назовите и дайте краткую характеристику командам git.

- git config - установка параметров
- git status - полный список изменений файлов, ожидающих коммита
- git add . - сделать все измененные файлы готовыми для коммита.
- git commit -m “[descriptive message]” - записать изменения с заданным сообщением.
- git branch - список всех локальных веток в текущей директории.
- git checkout [branch-name] - переключиться на указанную ветку и обновить рабочую директорию.
- git merge [branch] — соединить изменения в текущей ветке с изменениями из заданной.
- git push - запушить текущую ветку в удаленную ветку.
- git pull - загрузить историю и изменения удаленной ветки и произвести слияние с текущей веткой.

8. Приведите примеры использования при работе с локальным и удалённым репозиториями.

- git remote add [имя] [url] — добавляет удалённый репозиторий с заданным именем;
- git remote remove [имя] — удаляет удалённый репозиторий с заданным именем;
- git remote rename [старое имя] [новое имя] — переименовывает удалённый репозиторий;
- git remote set-url [имя] [url] — присваивает репозиторию с именем новый адрес;

- `git remote show [имя]` — показывает информацию о репозитории.

## 9. Что такое и зачем могут быть нужны ветви (branches)?

Ветвление — это возможность работать над разными версиями проекта: вместо одного списка с упорядоченными коммитами история будет расходиться в определённых точках. Каждая ветвь содержит легковесный указатель HEAD на последний коммит, что позволяет без лишних затрат создать много веток. Ветка по умолчанию называется `master`, но лучше назвать её в соответствии с разрабатываемой в ней функциональностью.

## 10. Как и зачем можно игнорировать некоторые файлы при commit?

Нет проблем если шаблон для игнорирования подходит для файла под контролем версий, или вы добавили файл, который игнорируется. Шаблоны не имеют никакого эффекта на файлы под контролем версий, они только определяют показываются неизвестные файлы, или просто игнорируются. Файл `gitignore` обычно должен быть под контролем версий, что бы новые копии ветки видели такие же шаблоны: `git add . gitignore git commit -m "Добавлены шаблоны для игнорирования"`. Многие деревья с исходным кодом содержат файлы, которые не нужно хранить под контролем версий, например, резервные файлы текстового редактора, объектные файлы и собранные программы. Вы можете просто не добавлять их, но они всегда будут обнаруживаться как неизвестные. Вы также можете сказать `bzr` игнорировать их добавив их в файл в корне рабочего дерева. Этот файл содержит список шаблонов файлов, по одному в каждой строчке. Обычное содержимое может быть таким: `.o ~ .tmp .py [ со ]` Если шаблон содержит слеш, то он будет сопоставлен с полным путем начиная от корня рабочего дерева; иначе он сопоставляется только с именем файла. Таким образом пример выше игнорирует файлы с расширением `.o` во всех подкаталогах, но пример ниже игнорирует только `config.h` в корне рабочего дерева и HTML файлы в каталоге `doc/`: `./config.h doc/.html` Для получения списка файлов которые игнорируются и со-

*ответствующих им шаблонов используйте команду git ignored : \$ git ignored config.h  
./config.h configure.in~ ~ \$*

Ссылка на литературу [1]

# **Список литературы**

1. Лекция Системы контроля версий [Электронный ресурс]. URL: [http://uii.mpei.ru/study/courses/sdt/16/lecture02.2\\_vcs.slides.pdf](http://uii.mpei.ru/study/courses/sdt/16/lecture02.2_vcs.slides.pdf).
  1. Лекция Системы контроля версий