

Лабараторная работа №2

Отчет

Славинский Владислав Вадимович

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	14
5	Ответы на контрольные вопросы	15

Список иллюстраций

3.1	Установка git	7
3.2	Установка gh	7
3.3	Данные репозитория	8
3.4	utf-8,autocrlf,safecrlf	8
3.5	Создание ключей	9
3.6	Добавление ключа	10
3.7	Подписи	10
3.8	Настройка gh	11
3.9	Создание рабочего пространства	12
3.10	Настройка каталога курса	13

Список таблиц

1 Цель работы

Изучить Идеологию и применение средств контроля версий. Освоить умения по работе с git.

2 Задание

Установка git Установка gh Базовая настройка git Создание ключей ssh и pgr
Настройка github Добавление pgr ключа в github Настройка автоматический
подписей коммитов git Настройка gh Создание репозитория курса на основе
шаблона Настройка каталога курса Контрольные вопросы

3 Выполнение лабораторной работы

В начале установим git (рис. 3.1)

```
root
[alavinskiyev@alavinskiyev ~]$ sudo dnf install git
[sudo] пароль для alavinskiyev:
Обновление и загрузка репозитория:
[]
```

Рис. 3.1: Установка git

Установим gh (рис. 3.2)

```
Нечего делать.
[alavinskiyev@alavinskiyev ~]$ sudo dnf install gh
Обновление и загрузка репозитория:
Репозитории загружены.
Пакет "gh-2.65.8-1.fc41.x86_64" уже установлен.
```

Рис. 3.2: Установка gh

Задал имя и email своего репозитория (рис. 3.3)

```

[alavimkiyev@alavimkiyev ~]$ git config --global user.name "Vladislav Slavimkiy"
[alavimkiyev@alavimkiyev ~]$ git config --global user.email "alavimkiy@50500@gmail.com"
[alavimkiyev@alavimkiyev ~]$

```

Рис. 3.3: Данные репозитория

Настройка utf-8, задаю имя начальной ветки, ввел параметр autocrlf и safecrlf (рис. 3.4)

```

[alavimkiyev@alavimkiyev ~]$ git config --global core.quotePath false
[alavimkiyev@alavimkiyev ~]$ git config --global init.defaultBranch master
[alavimkiyev@alavimkiyev ~]$ git config --global core.autocrlf input
[alavimkiyev@alavimkiyev ~]$

```

Рис. 3.4: utf-8, autocrlf, safecrlf

Создал ключи ssh (рис. 3.5)


```

Generating public/private rsa key pair.
Enter file in which to save the key [/home/alavinskiyvv/.ssh/id_rsa]:
Created directory '/home/alavinskiyvv/.ssh'.
Enter passphrase for '/home/alavinskiyvv/.ssh/id_rsa' (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/alavinskiyvv/.ssh/id_rsa
Your public key has been saved in /home/alavinskiyvv/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:KSmrH08L/1jUvV79yNc5dFotHDnJ1dFMjyPa18wQ alavinskiyvv@alavinskiyvv
The key's randomart image is:
+---[RSA 4096]---+
|
|      .
|      o o
|      +. ^ o E|
|      . S oS = +|
|      . . o^S o o|
|      . . . +. o +^|
|      o . . . . o^X|
|      o+o      +S^|
+---[SHA256]-----+
[alavinskiyvv@alavinskiyvv ~]$ ssh-keygen -t ed25519
Generating public/private ed25519 key pair.
Enter file in which to save the key [/home/alavinskiyvv/.ssh/id_ed25519]:
Enter passphrase for '/home/alavinskiyvv/.ssh/id_ed25519' (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/alavinskiyvv/.ssh/id_ed25519
Your public key has been saved in /home/alavinskiyvv/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:9Cy019cuzMa+jH5528xpd1Tz21C5wvS8.838eL/K1Y alavinskiyvv@alavinskiyvv
The key's randomart image is:
+---[ED25519 256]---+
|
|      .
|      o
|      o,=
|      ..Oo.
|      oS0o
|      oS, ^ . E
|      ++=oo+.
|      X++^o+o.
|      ooX@++ o..
+---[SHA256]-----+
[alavinskiyvv@alavinskiyvv ~]$

```

Рис. 3.5: Создание ключей

Добавление ssh ключа на git (рис. 3.6)

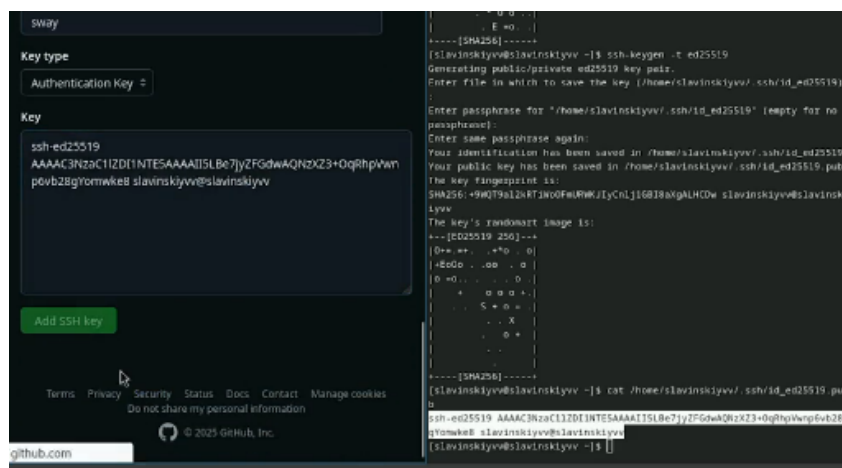


Рис. 3.6: Добавление ключа

Настройка автоматических подписей коммитов git (рис. 3.7)

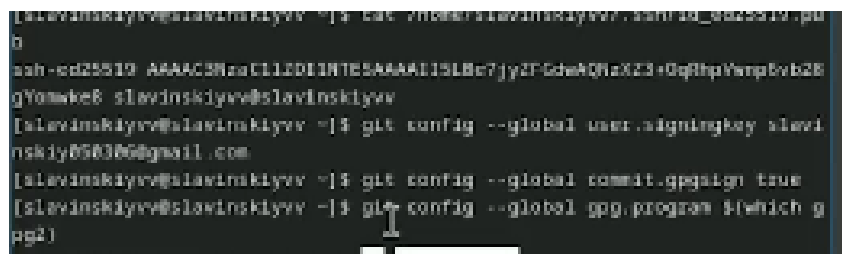


Рис. 3.7: Подписи

Настройка gh (рис. 3.8)

```

? How would you like to authenticate GitHub CLI? [Use arrows to move, t
ype to filter]
  Login with a web browser
+ Paste an authentication token

[slavinskiyvv@slavinskiyvv ~]$ gh auth login
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? HTTPS
? Authenticate Git with your GitHub credentials? (Y/n)

[slavinskiyvv@slavinskiyvv ~]$ gh auth login
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? SSH
? Upload your SSH public key to your GitHub account? /home/slavinskiyvv/.ssh/id_ed25519.pub
? Title for your SSH key: sway
? How would you like to authenticate GitHub CLI? Login with a web browse
r

! First copy your one-time code: 879A-A325
Press Enter to open https://github.com/login/device in your browser...
^[[A
^[[A^[[A^[[A^[[B^C
[slavinskiyvv@slavinskiyvv ~]$ gh auth login
? Where do you use GitHub? GitHub.com
? What is your preferred protocol for Git operations on this host? SSH
? Upload your SSH public key to your GitHub account? /home/slavinskiyvv/.ssh/id_ed25519.pub
? Title for your SSH key: sway
? How would you like to authenticate GitHub CLI? Login with a web browse
r

! First copy your one-time code: CADE-781A
Press Enter to open https://github.com/login/device in your browser...
✓ Authentication complete.
- gh config set -h github.com git_protocol ssh
✓ Configured git protocol
✓ SSH key already existed on your GitHub account: /home/slavinskiyvv/.ssh/id_ed25519.pub
✓ Logged in as Zoro1337Z
[slavinskiyvv@slavinskiyvv ~]$ █

```

Рис. 3.8: Настройка gh

Произвел операции над созданием рабочего пространства (рис. 3.9)

```
| First copy your one-time code: GABE-781A
Press Enter to open https://github.com/login/device in your browser...
✓ Authentication complete.
- gh config set -h github.com git_protocol ssh
✓ Configured git protocol
✓ SSH key already existed on your GitHub account: /home/slavinskiyvv/.ssh/id_ed25519.pub
✓ Logged in as slavinskiyvv
[slavinskiyvv@slavinskiyvv ~]$ mkdir -p -/work/study/2024-2025/"Операционные системы"
[slavinskiyvv@slavinskiyvv ~]$ cd -/work/study/2022-2023/"Операционные системы"
bash: cd: /home/slavinskiyvv/work/study/2022-2023/Операционные системы:
Нет такого файла или каталога
[slavinskiyvv@slavinskiyvv ~]$ cd -/work/study/2024-2025/"Операционные системы"
[slavinskiyvv@slavinskiyvv Операционные системы]$ gh repo create study_2022-2023_os-intro --template=yamasharma/course-directory-student-template --public
```

Рис. 3.9: Создание рабочего пространства

Настроил каталог курса, удалил лишние файлы, создал необходимые каталоги и отправил их на сервер (рис. 3.10)

```

tic.cnl
create mode 100755 project-personal/stage5/report/pandoc/filters/pandoc_extras.py
create mode 100755 project-personal/stage5/report/pandoc/filters/pandoc_figures.py
create mode 100755 project-personal/stage5/report/pandoc/filters/pandoc_sections.py
create mode 100755 project-personal/stage5/report/pandoc/filters/pandoc_tableaux.py
create mode 100644 project-personal/stage5/report/pandoc/filters/pandocxref/__init_
_Py
create mode 100644 project-personal/stage5/report/pandoc/filters/pandocxref/core.py
create mode 100644 project-personal/stage5/report/pandoc/filters/pandocxref/main.py
create mode 100644 project-personal/stage5/report/pandoc/filters/pandocxref/pandocx
tributes.py
create mode 100644 project-personal/stage5/report/report.ed
create mode 100644 project-personal/stage5/presentation/.projectile
create mode 100644 project-personal/stage5/presentation/.toolboxroot
create mode 100644 project-personal/stage5/presentation/Makefile
create mode 100644 project-personal/stage5/presentation/image/kulyakov.jpg
create mode 100644 project-personal/stage5/presentation/presentation.ed
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/placating_200_600_tech.jpg
create mode 100644 project-personal/stage5/report/pandoc/cal/gmt-t-7-8-5-2005-runs
tic.cnl
create mode 100755 project-personal/stage5/report/pandoc/filters/pandoc_extras.py
create mode 100755 project-personal/stage5/report/pandoc/filters/pandoc_figures.py
create mode 100755 project-personal/stage5/report/pandoc/filters/pandoc_sections.py
create mode 100755 project-personal/stage5/report/pandoc/filters/pandoc_tableaux.py
create mode 100644 project-personal/stage5/report/pandoc/filters/pandocxref/__init_
_Py
create mode 100644 project-personal/stage5/report/pandoc/filters/pandocxref/core.py
create mode 100644 project-personal/stage5/report/pandoc/filters/pandocxref/main.py
create mode 100644 project-personal/stage5/report/pandoc/filters/pandocxref/pandocx
tributes.py
create mode 100644 project-personal/stage5/report/report.ed
[slavinskiy@slavinskiy cs-intro]$ git push
Пересылка объектов: 40, готово.
Пакет объектов: 100% (40/40), готово.
При сжатии наименьшей используется до 8 потоков
Создание объектов: 100% (38/38), готово.
Сумма объектов: 100% (38/38), 342.32 KiB | 2.85 MiB/s, готово.
Total 38 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (4/4), completed with 1 local object.
To github.com:Dere1337D/study_2024-2025_cs-intro.git
 992cb23..4f62b3b master -> master
[slavinskiy@slavinskiy cs-intro]$

```

Рис. 3.10: Настройка каталога курса

4 Выводы

В ходе выполнения лабораторной работы был установлен git, его настройка, были созданы ключи для авторизации и подписи. Был создан репозиторий.

5 Ответы на контрольные вопросы

1. VCS-это инструменты для отслеживания изменений в файлах и управления проектами, позволяющие сохранять версии и координировать работу.
2. Хранилище-место хранения файлов и их истории. Commit-сохранение изменений в хранилище. История-последовательность всех коммитов, отражающая изменения в проекте. Рабочая копия-локальная версия файлов, с которой работает разработчик.
3. Централизованные: имеют одно центральное хранилище, к которому подключаются все пользователи. Децентрализованные каждый разработчик имеет полную копию хранилища, включая всю историю.
4. Создание хранилища. Внесение изменений в рабочую копию. Выполнение команды commit для сохранения изменений. Просмотр истории изменений.
5. Клонирование удаленного репозитория. Внесение изменений и создание коммитов. Синхронизация с удаленным репозиторием (pull/push). Разрешение конфликтов, если они возникают.
6. Отслеживание изменений в коде. Восстановление предыдущих версий. Совместная работа над проектами. Управление ветвями.
7. git init: инициализация нового репозитория. git clone: клонирование удаленного репозитория. git add: добавление изменений в индекс. git commit:

сохранение изменений в хранилище. `git push`: отправка изменений в удаленный репозиторий. `git pull`: получение изменений из удаленного репозитория. `git branch`: управление ветвями. `git merge`: слияние ветвей.

8. Локальный репозиторий: `git init`, `git add .`, `git commit -m "Initial commit"`.
Удаленный репозиторий: `git clone`, `git push origin main`.
9. Ветви позволяют создавать параллельные линии разработки, что упрощает работу над новыми функциями или исправлениями, не затрагивая основную ветвь
10. Файлы можно игнорировать с помощью файла `.gitignore`, чтобы исключить их из коммитов (например, временные файлы, конфигурации среды), что помогает поддерживать чистоту репозитория.