

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2

дисциплина: Архитектура компьютера

Студент: Бережной Иван Александрович

Группа: НКАбд-01-#

МОСКВА

2023 г.

Оглавление

1 Цель работы	3
2 Задания	4
3 Выполнение лабораторной работы.....	5
4 Задание для самостоятельной работы	11
5 Выводы	13
Список литературы.....	14

1 Цель работы

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

2 Задания

1. Настройка github
2. Базовая настройка github
3. Создание SHH ключа
4. Создание рабочего пространства и репозитория курса на основе шаблона
5. Создание репозитория курса на основе шаблона
6. Настройка каталога курса
7. Задание для самостоятельной работы

3 Выполнение лабораторной работы

3.1 Настройка GitHub

Перейдём на сайт <https://github.com/>. Справа вверху нажмём кнопку «sign up» и пройдём регистрацию (рис.3.1.1).

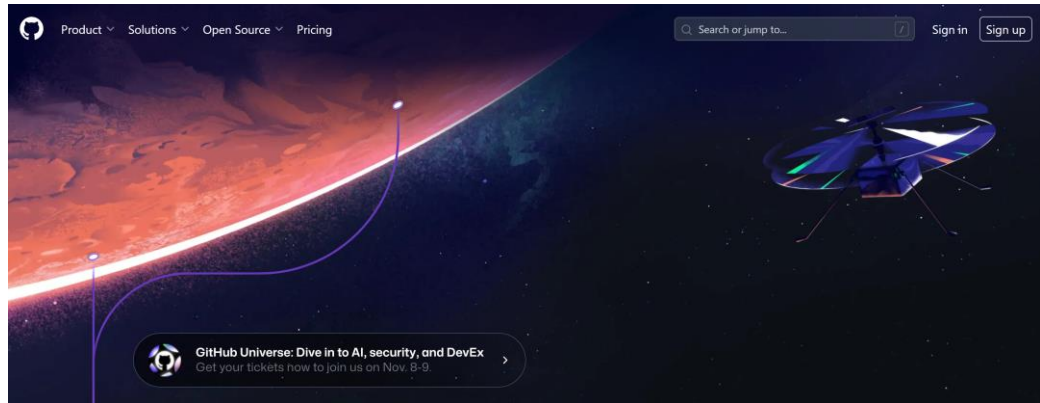


рис. 3.1.1: Кнопка регистрации GitHub

3.2 Базовая настройка git

Откроем терминал и укажем имя и email владельца репозитория с помощью команд, отображённых на рис. 3.2.1.

```
[iaberezhnoy@fedora ~]$ git config --global user.name "<Ivan Berezhnoy>"  
[iaberezhnoy@fedora ~]$ git config --global user.email "<darkgxd@vk.com>"  
[iaberezhnoy@fedora ~]$
```

рис. 3.2.1: Имя и email пользователя

Далее настроим utf-8 в выводе сообщений git для корректного отображения символов, а также зададим имя начальной ветки (master) и параметры autocrlf и safecrlf (рис. 3.2.2).

```
[iaberezhnoy@fedora ~]$ git config --global core.quotePath false  
[iaberezhnoy@fedora ~]$ git config --global init.defaultBranch master  
[iaberezhnoy@fedora ~]$ git config --global core.autocrlf input  
[iaberezhnoy@fedora ~]$ git config --global core.safecrlf warn  
[iaberezhnoy@fedora ~]$
```

рис. 3.2.2: Продолжение настройки git

3.3 Создание SSH ключа

Сгенерируем пару ключей (приватный и открытый). Для этого введём

команду `ssh-keygen -C "Ivan Berezhnoy <darkgxd@vk.com>"`. Ключи автоматически сохраняются в каталоге `~/.ssh/` (рис. 3.3.1).

```
[iaberezhnoy@fedora ~]$ ssh-keygen -C "Ivan Berezhnoy <darkgxd@vk.com>"
Generating public/private rsa key pair.
Enter file in which to save the key (/home/iaberezhnoy/.ssh/id_rsa):
Created directory '/home/iaberezhnoy/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/iaberezhnoy/.ssh/id_rsa
Your public key has been saved in /home/iaberezhnoy/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:hwAwjH0wk5wkjHdgQ/RntFlap3hHub8h9xQK7SaUxs0 Ivan Berezhnoy <darkgxd@vk.com>
The key's randomart image is:
+---[RSA 3072]-----+
|  =BBO=. o o.      |
| ..+=B=.B +.      |
| . ...0 o..*       |
|    o o oB E .     |
|      So.+ . .     |
|    .o B .         |
|      = =          |
|      . .          |
+-----[SHA256]-----+
[iaberezhnoy@fedora ~]$
```

рис. 3.3.1: Генерация SSH ключа

Скопируем сгенерированный открытый ключ с помощью команды `cat` и утилиты `xclip` (рис. 3.3.2).

```
[iaberezhnoy@fedora ~]$ cat ~/.ssh/id_rsa.pub | xclip -sel clip
```

рис. 3.3.2: Копирование открытого ключа

Зайдём на сайт GitHub под своей учётной записью, перейдём в настройки, далее в меню «SSH and GPG keys», нажмём на кнопку «New SSH key». Теперь можем вставить скопированный ключ в соответствующую строку, также укажем имя ключа (рис. 3.3.3).

Add new SSH Key

Title

iaberezhnoy18

Key type

Authentication Key

Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQGCrsIgtNtQcNJKBTxyvTmDE5Umx82batLwhDmMUUx2ggHMfpWiKnFKAiYgd
V6dpJ5jYA9P4Z4yYEjGn45mk6NvreYTTgT4ys14oQCb0MwYHGj7RbVXILrsoiPoHWtKkREYwAQtyH4BFuvFis60CS/wl39Z
SXsyGFVRyy24Xtk/aQJGjllsKONsY+svUzVSML+TnWMai43rK3XFMTRPXaLsyXK/0LDxqO1kPYo0GvbswZN8wbAaEHU1sl
olZ+T40MaMqpWntbmPm26nNskhaNkoBjFUQdgwtKypXGsQEWnXZVjcHQThrYvdp6ztCc5KvgT6cRzyr/ZBvKwE1Cm1x
aTQojZzXpP/9vkGzzKMKRltQ4KynjOPa+xlTmEziawG6jKq2HigyO9zHxE6exH6UBk/LlfMynBWv+VK8OaJTVFnY32/gwgRj
c3jfNYpAc1qennA/D2XVaABUiAdaQ0K5h9Ew5yja9la8x4EhRC1mK2waSBIQcgDbkrGCB0GFgb63wc= Ivan Berezhnoy
<darkgxd@vk.com>
```

Add SSH key

рис. 3.3.3: Загрузка SSH ключа

3.4 Создание рабочего пространства и репозитория курса на основе шаблона

Будем придерживаться структуре рабочего пространства. Для этого перейдём в терминал и создадим иерархию каталогов для предмета «Архитектура компьютера» командой `mkdir` с ключом `-p` (рис.3.4.1).

```
[iaberezhnoy@fedora ~]$ mkdir -p ~/work/study/2023-2024/"Архитектура компьютера"
```

рис. 3.4.1: Создание каталога «Архитектура компьютеров»

3.5 Создание репозитория курса на основе шаблона

Снова откроем браузер и перейдём к репозиторию с шаблоном курса по адресу <https://github.com/yamadharma/course-directory-student-template> и нажмём кнопку «Use this template» (рис. 3.5.1).

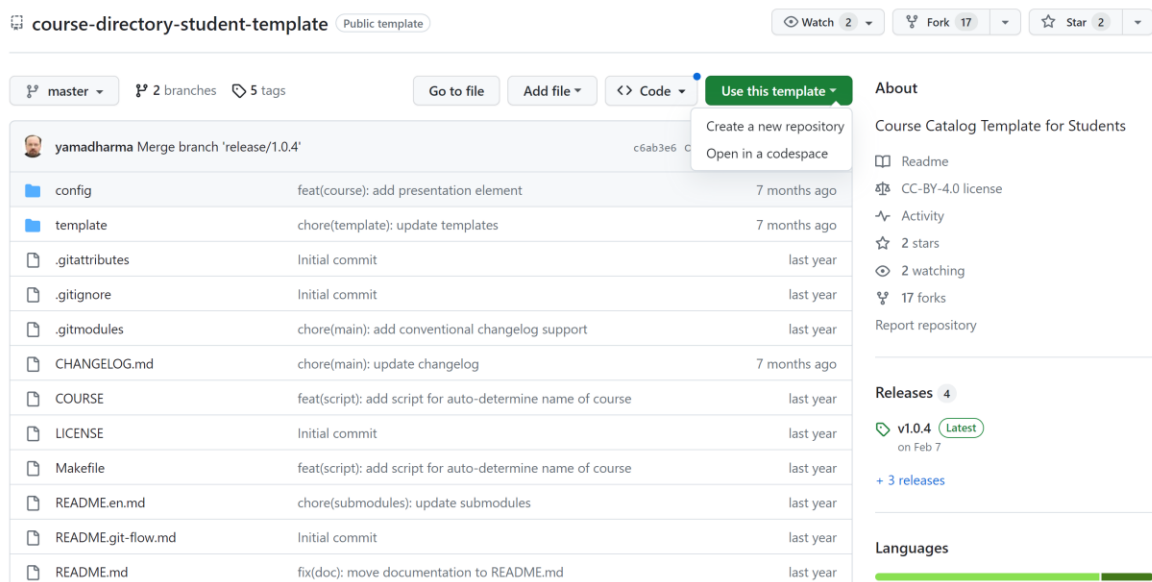


рис. 3.5.1: Шаблон курса

Далее создадим новый репозиторий, нажав кнопку «Create repository», предварительно задав ему имя «study_2023-2024_arhpc» (рис. 3.5.2).

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

Owner * Repository name *

NoisyCake / study_2023-2024_arhpc

✓ study_2023-2024_arhpc is available.

Great repository names are short and memorable. Need inspiration? How about [reimagined-parakeet](#) ?

Description (optional)

- ☒ **Public**
Anyone on the internet can see this repository. You choose who can commit.
- ☐ **Private**
You choose who can see and commit to this repository.

You are creating a public repository in your personal account.

Create repository

рис. 3.5.2: Создание репозитория

Теперь мы можем клонировать созданный репозиторий. Для этого

воспользуемся командой `git clone --recursive`, добавив SSH ссылку на сам репозиторий.

```
[iaberezhnoy@fedora ~]$ cd
[iaberezhnoy@fedora ~]$ cd ~/work/study/2023-2024/"Архитектура компьютера"
[iaberezhnoy@fedora Архитектура компьютера]$ git clone --recursive git@github.com:NoisyCake/study_2023-2024_arhpc.git arch.pc
Клонирование в «arch.pc»...
remote: Enumerating objects: 27, done.
remote: Counting objects: 100% (27/27), done.
```

рис. 3.5.3: Клонирование репозитория

3.6 Настройка каталога курса

Переходим в каталог курса с помощью команды `cd` и удаляем ненужные файлы командой `rm` (рис. 3.6.1).

```
[iaberezhnoy@fedora ~]$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc
[iaberezhnoy@fedora arch-pc]$ rm package.json
```

рис. 3.6.1: Удаление файлов

Теперь создадим необходимые каталоги (рис. 3.6.2) и отправим файлы на сервер (рис. 3.6.3).

```
[iaberezhnoy@fedora arch-pc]$ echo arch-pc > COURSE
[iaberezhnoy@fedora arch-pc]$ make
```

рис. 3.6.2: Создание каталогов

```
[iaberezhnoy@fedora arch-pc]$ git add .
[iaberezhnoy@fedora arch-pc]$ git commit -am 'feat(main): make course structure'
[master db2ceb0] feat(main): make course structure
199 files changed, 54725 insertions(+), 14 deletions(-)
```

рис. 3.6.3: Отправление каталогов на сервер

Проверим, все ли папки и файлы созданы. Сначала откроем файловый менеджер Fedora (рис. 3.6.4), затем проверим GitHub (рис. 3.6.5). Всё получилось.



рис. 3.6.4: Проверка файлового менеджера

NoisyCake feat(main): make course structure			db2ceb0 · 13 minutes ago	History
Name	Last commit message	Last commit date		
..				
lab01	feat(main): make course structure	13 minutes ago		
lab02	feat(main): make course structure	13 minutes ago		
lab03	feat(main): make course structure	13 minutes ago		
lab04	feat(main): make course structure	13 minutes ago		
lab05	feat(main): make course structure	13 minutes ago		
lab06	feat(main): make course structure	13 minutes ago		
lab07	feat(main): make course structure	13 minutes ago		
lab08	feat(main): make course structure	13 minutes ago		
lab09	feat(main): make course structure	13 minutes ago		
lab10	feat(main): make course structure	13 minutes ago		
lab11	feat(main): make course structure	13 minutes ago		
README.md	feat(main): make course structure	13 minutes ago		
README.ru.md	feat(main): make course structure	13 minutes ago		

рис. 3.6.5: Проверка на GitHub

4 Задание для самостоятельной работы

1. Для начала создадим отчёт по лабораторной работе №2. Откроем LibreOffice Writer внутри Fedora, создадим и сохраним пустой файл с именем «Lab2_report» в папке Документы (рис. 4.1.1).

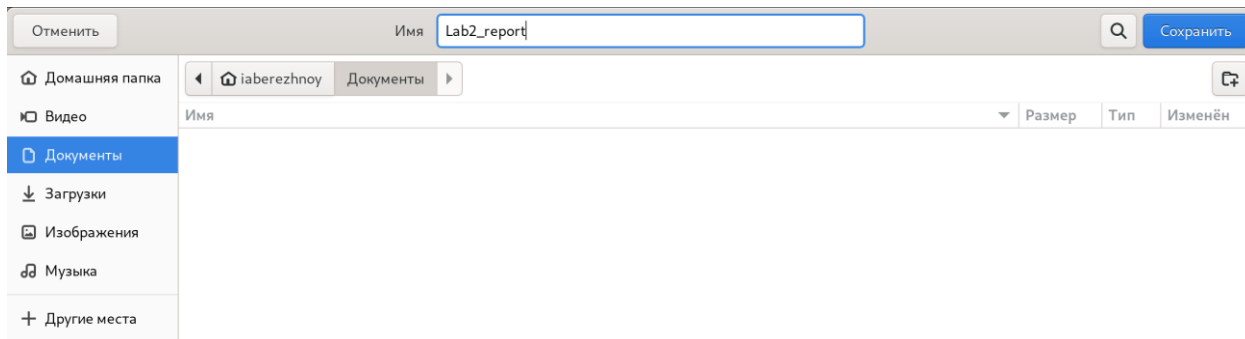


рис. 4.1.1: Создание файла для отчёта

После написания отчёта переместим его в нужную нам папку через терминал командой mv (рис. 4.1.2).

```
[iaberezhnoy@fedora ~]$ mv ~/Документы/Lab2_report.odt ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab02/report
```

рис. 4.1.2: Перемещение отчёта

2. Вторым шагом скопируем отчёт предыдущей лабораторной работы в соответствующую папку командой cp (4.1.3).

```
[iaberezhnoy@fedora ~]$ cp ~/Документы/Lab1_report.odt ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab01/report
```

рис. 4.1.3: Копирование старого отчёта

3. Теперь загрузим файлы на GitHub. С помощью утилиты cd перейдём в папку каждого отчёта, а командой git add добавим файлы отчётов в коммит. Затем сохраним внесённые изменения командой git commit с ключом -am и отправим их в центральный репозиторий (рис. 4.1.4).

```
[iaberezhnoy@fedora report]$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab01/report
[iaberezhnoy@fedora report]$ git add Lab1_report.odt
[iaberezhnoy@fedora report]$ cd ~/work/study/2023-2024/"Архитектура компьютера"/arch-pc/labs/lab02/report
[iaberezhnoy@fedora report]$ git add Lab2_report.odt
[iaberezhnoy@fedora report]$ git commit -am "Lab files were added"
[master 43e3c61] Lab files were added
2 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 labs/lab01/report/Lab1_report.odt
create mode 100644 labs/lab02/report/Lab2_report.odt
[iaberezhnoy@fedora report]$ git push
Перечисление объектов: 15, готово.
Подсчет объектов: 100% (13/13), готово.
При сжатии изменений используется до 5 потоков
Сжатие объектов: 100% (9/9), готово.
Запись объектов: 100% (9/9), 10.00 КиБ | 5.00 МиБ/с, готово.
Всего 9 (изменений 4), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
remote: Resolving deltas: 100% (4/4), completed with 3 local objects.
To github.com:NoisyCake/study_2023-2024_arhpc.git
db2ceb0..43e3c61 master -> master
[iaberezhnoy@fedora report]$
```

рис. 4.1.4: Загрузка отчётов на GitHub

Проверим верность выполнения команд. Откроем репозиторий в GitHub. В папке lab01/report видим наш файл Lab1_report.pdf (рис.4.1.5), а в папке lab01/report файл Lab2_report.pdf (рис. 4.1.6). Всё верно.

NoisyCake Lab files were added09d4ee6 · 3 minutes agoHistory

Name	Last commit message	Last commit date
..		
bib	feat(main): make course structure	1 hour ago
image	feat(main): make course structure	1 hour ago
pandoc	feat(main): make course structure	1 hour ago
Lab1_report.pdf	Lab files were added	3 minutes ago
Makefile	feat(main): make course structure	1 hour ago
report.md	feat(main): make course structure	1 hour ago

рис. 4.1.5: Проверка первого отчёта

NoisyCake Lab files were added5a82c03 · 1 minute agoHistory

Name	Last commit message	Last commit date
..		
bib	feat(main): make course structure	1 hour ago
image	feat(main): make course structure	1 hour ago
pandoc	feat(main): make course structure	1 hour ago
Lab2_report.pdf	Lab files were added	1 minute ago
Makefile	feat(main): make course structure	1 hour ago
report.md	feat(main): make course structure	1 hour ago

рис. 4.1.6: Проверка второго отчёта

5 Выводы

При выполнении лабораторной работы мы изучили идеологию и применение средств контроля версий, а также приобрели практические навыки по работе с системой git.

Список литературы

1. [Архитектура ЭВМ](#)