

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: Гадаборшев Заур Закреевич

Группа: НПИбд-01-23

МОСКВА

2023 г.

Цель работы

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

Ход работы

Регистрируюсь на гитхабе (Рис. 1)

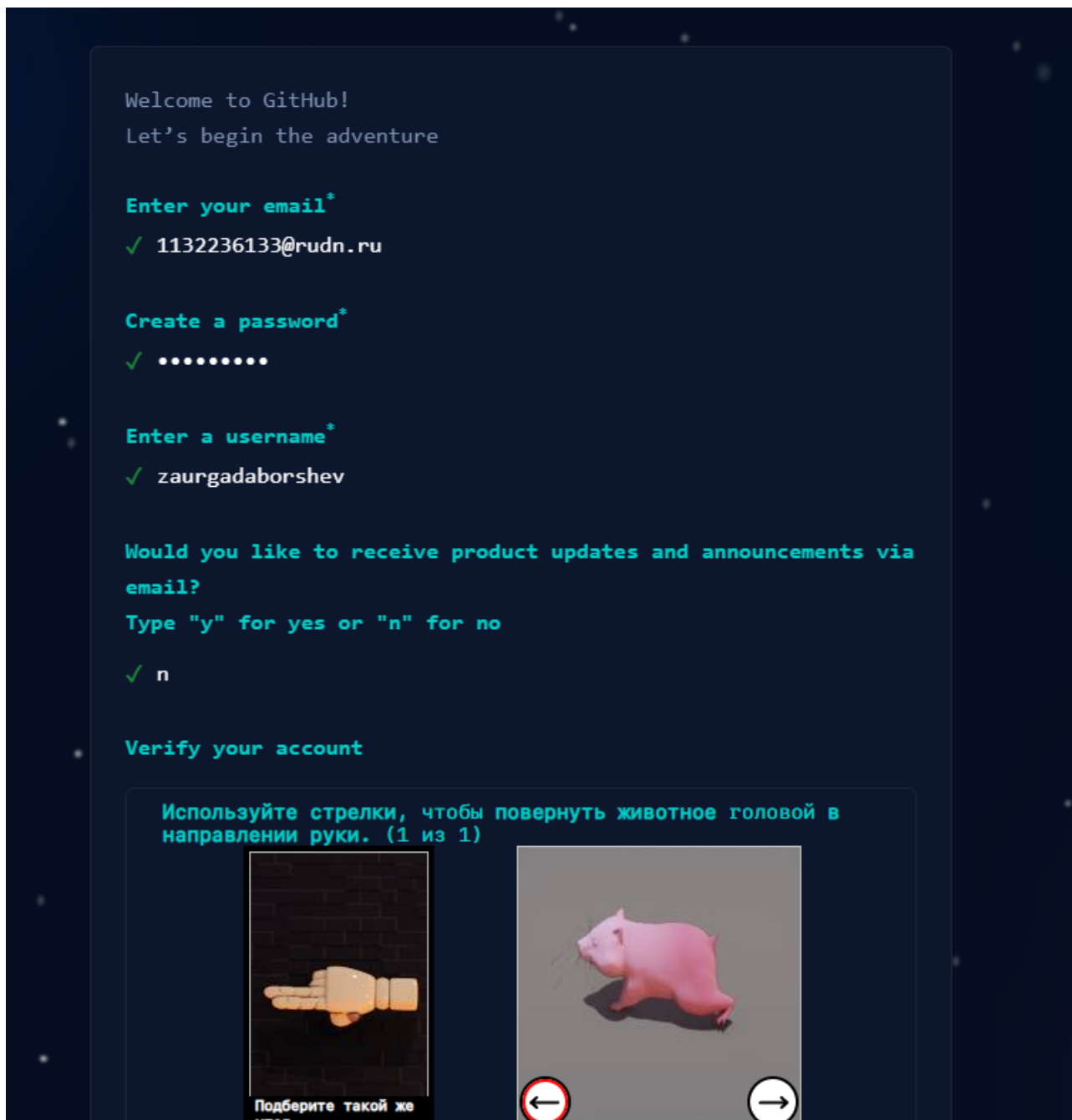


Рисунок 1: регистрация на гитхабе

Нахожу шаблонный репозиторий и создаю из него свой. (Рис. 1.1)

course-directory-student-template

Public template

Watch 2

master

2 branches

5 tags

Go to file

Add file

<> Code

Use this template

yamadharma

Merge branch 'release/1.0.4'

c6ab3e6 on Feb 7

34 commits

config	feat(course): add presentation element	7 months ago
template	chore(template): update templates	7 months ago
.gitattributes	Initial commit	last year
.gitignore	Initial commit	last year
.qitmodules	chore(main): add conventional changelog support	last year

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 *

zaurgadaborshev

/ arch-pc

arch-pc is available.

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

Description (optional)

Архитектура компьютеров

☒ Public

☐ Private

Anyone on the internet can see this repository. You choose who can commit.

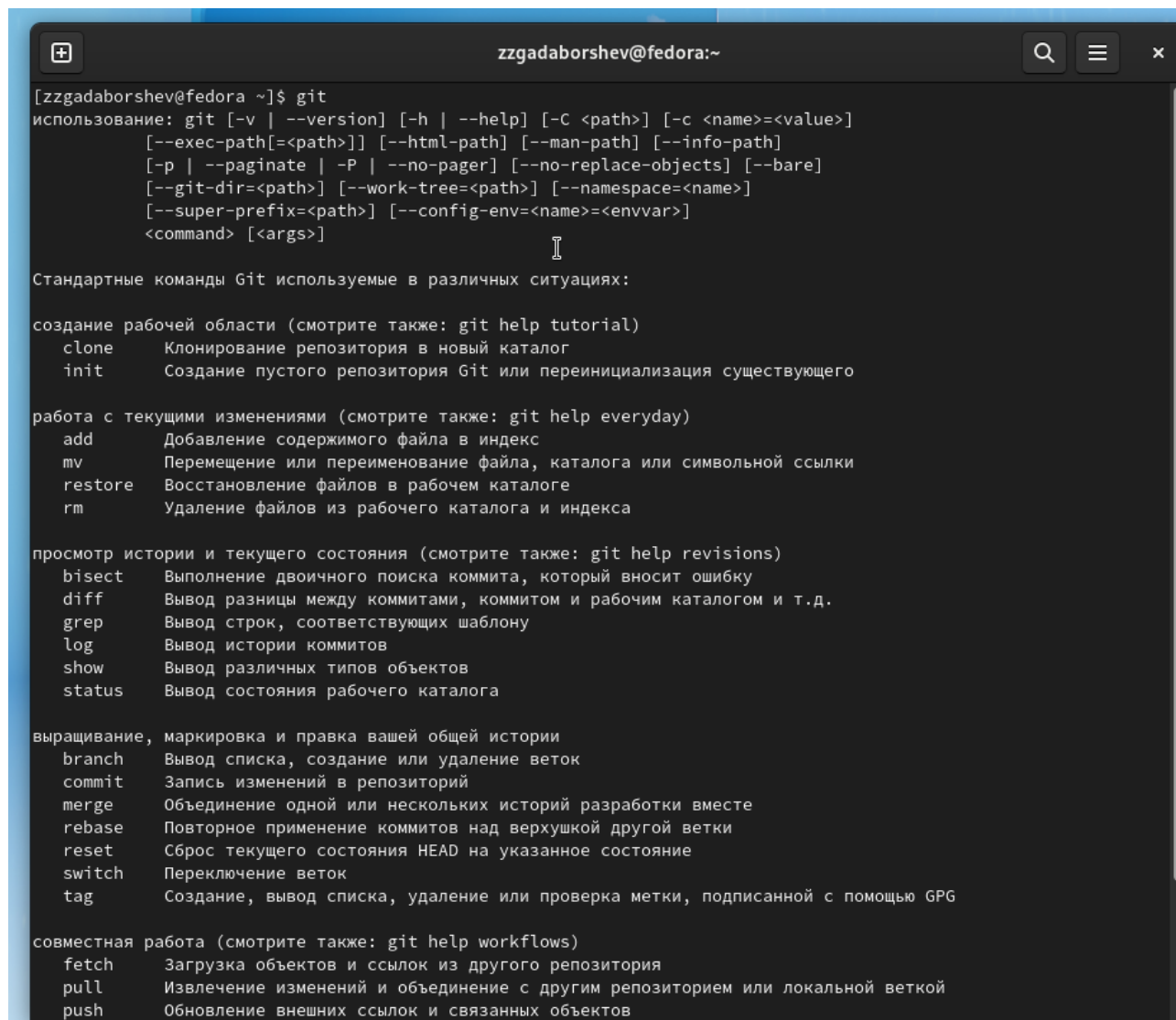
You choose who can see and commit to this repository.

You are creating a public repository in your personal account.

Create repository

Рисунок 1.1: создаю свой репозиторий

Установил программу гит (Рис 1.2)



```
[zzgadaborshev@fedora ~]$ git
использование: git [-v | --version] [-h | --help] [-C <path>] [-c <name>=<value>]
                [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
                [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
                [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
                [--super-prefix=<path>] [--config-env=<name>=<envvar>]
                <command> [<args>]

Стандартные команды Git используемые в различных ситуациях:

создание рабочей области (смотрите также: git help tutorial)
  clone      Клонирование репозитория в новый каталог
  init       Создание пустого репозитория Git или переинициализация существующего

работа с текущими изменениями (смотрите также: git help everyday)
  add        Добавление содержимого файла в индекс
  mv         Перемещение или переименование файла, каталога или символической ссылки
  restore    Восстановление файлов в рабочем каталоге
  rm         Удаление файлов из рабочего каталога и индекса

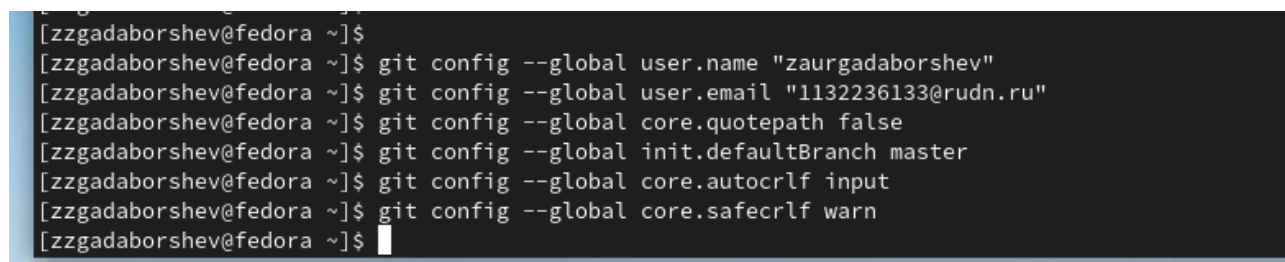
просмотр истории и текущего состояния (смотрите также: git help revisions)
  bisect     Выполнение двоичного поиска коммита, который вносит ошибку
  diff       Вывод разницы между коммитами, коммитом и рабочим каталогом и т.д.
  grep       Вывод строк, соответствующих шаблону
  log        Вывод истории коммитов
  show       Вывод различных типов объектов
  status     Вывод состояния рабочего каталога

выращивание, маркировка и правка вашей общей истории
  branch     Вывод списка, создание или удаление веток
  commit     Запись изменений в репозиторий
  merge      Объединение одной или нескольких историй разработки вместе
  rebase     Повторное применение коммитов над верхушкой другой ветки
  reset      Сброс текущего состояния HEAD на указанное состояние
  switch     Переключение веток
  tag        Создание, вывод списка, удаление или проверка метки, подписанной с помощью GPG

совместная работа (смотрите также: git help workflows)
  fetch      Загрузка объектов и ссылок из другого репозитория
  pull       Извлечение изменений и объединение с другим репозиторием или локальной веткой
  push       Обновление внешних ссылок и связанных объектов
```

Рисунок 1.2: программа гит

Сначала сделаем предварительную конфигурацию git, создаю пользователя и ставлю параметры. (Рис. 1.3)



```
[zzgadaborshev@fedora ~]$
[zzgadaborshev@fedora ~]$ git config --global user.name "zaurgadaborshev"
[zzgadaborshev@fedora ~]$ git config --global user.email "1132236133@rudn.ru"
[zzgadaborshev@fedora ~]$ git config --global core.quotepath false
[zzgadaborshev@fedora ~]$ git config --global init.defaultBranch master
[zzgadaborshev@fedora ~]$ git config --global core.autocrlf input
[zzgadaborshev@fedora ~]$ git config --global core.safecrlf warn
[zzgadaborshev@fedora ~]$
```

Рисунок 1.3: конфигурация гита

Далее создаю ключи для идентификации. (Рис 1.4)

```
[zzgadaborshev@fedora ~]$  
[zzgadaborshev@fedora ~]$ ssh-keygen -C "zaurgadaborshev 1132236133@rudn.ru"  
Generating public/private rsa key pair.  
Enter file in which to save the key (/home/zzgadaborshev/.ssh/id_rsa):  
Created directory '/home/zzgadaborshev/.ssh'.  
Enter passphrase (empty for no passphrase):  
Enter same passphrase again:  
Your identification has been saved in /home/zzgadaborshev/.ssh/id_rsa  
Your public key has been saved in /home/zzgadaborshev/.ssh/id_rsa.pub  
The key fingerprint is:  
SHA256:VaLOCTlunEojDiNd/cfZsIpiHnRFX8Se2YK8C1QL/0 zaurgadaborshev 1132236133@rudn.ru  
The key's randomart image is:  
+---[RSA 3072]-----+  
|      o.o o .      |  
|      . = B =      |  
|      . = B X =      |  
|    . . + O B &    |  
| + o = * S * E      |  
| .+ * = . o        |  
|   + = . .         |  
|    o .            |  
+-----[SHA256]-----+  
[zzgadaborshev@fedora ~]$
```

Рисунок 1.4:Ключи идентификации

И добавляю ключ в профиль на гитхабе (Рис. 1.5)

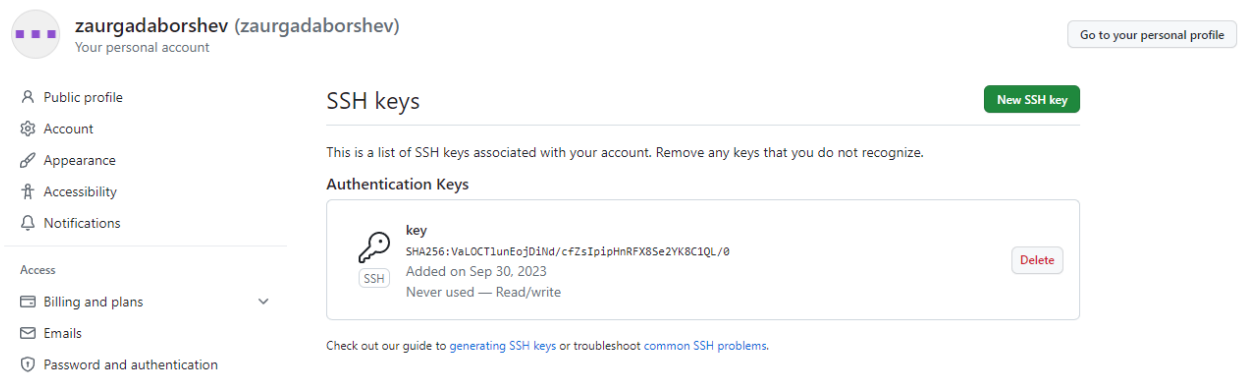


Рисунок 1.5:добавление ключа в гит

Теперь я создаю рабочий каталог и клонирую туда репозиторий с гитхаба.(Рис. 2)

```
Подмодуль «template/report» (https://github.com/yamadharma/academic-laboratory-report-template.git) зарегистриро  
ван по пути «template/report»  
Клонирование в «/home/zzgadaborshev/work/study/2023-2024/Архитектура компьютера/arch-pc/template/presentation»..  
.  
remote: Enumerating objects: 82, done.  
remote: Counting objects: 100% (82/82), done.  
remote: Compressing objects: 100% (57/57), done.  
remote: Total 82 (delta 28), reused 77 (delta 23), pack-reused 0  
Получение объектов: 100% (82/82), 92.90 КиБ | 889.00 КиБ/с, готово.  
Определение изменений: 100% (28/28), готово.  
Клонирование в «/home/zzgadaborshev/work/study/2023-2024/Архитектура компьютера/arch-pc/template/report»...  
remote: Enumerating objects: 101, done.  
remote: Counting objects: 100% (101/101), done.  
remote: Compressing objects: 100% (70/70), done.  
remote: Total 101 (delta 40), reused 88 (delta 27), pack-reused 0  
Получение объектов: 100% (101/101), 327.25 КиБ | 2.14 МиБ/с, готово.  
Определение изменений: 100% (40/40), готово.  
Submodule path 'template/presentation': checked out 'b1be3800ee91f5809264cb755d316174540b753e'  
Submodule path 'template/report': checked out '1d1b61dcac9c287a83917b82e3aef11a33b1e3b2'  
[zzgadaborshev@fedora Архитектура компьютера]$
```

Рисунок 2: клонирование репозитория

Создаю курс (Рис. 2.1)

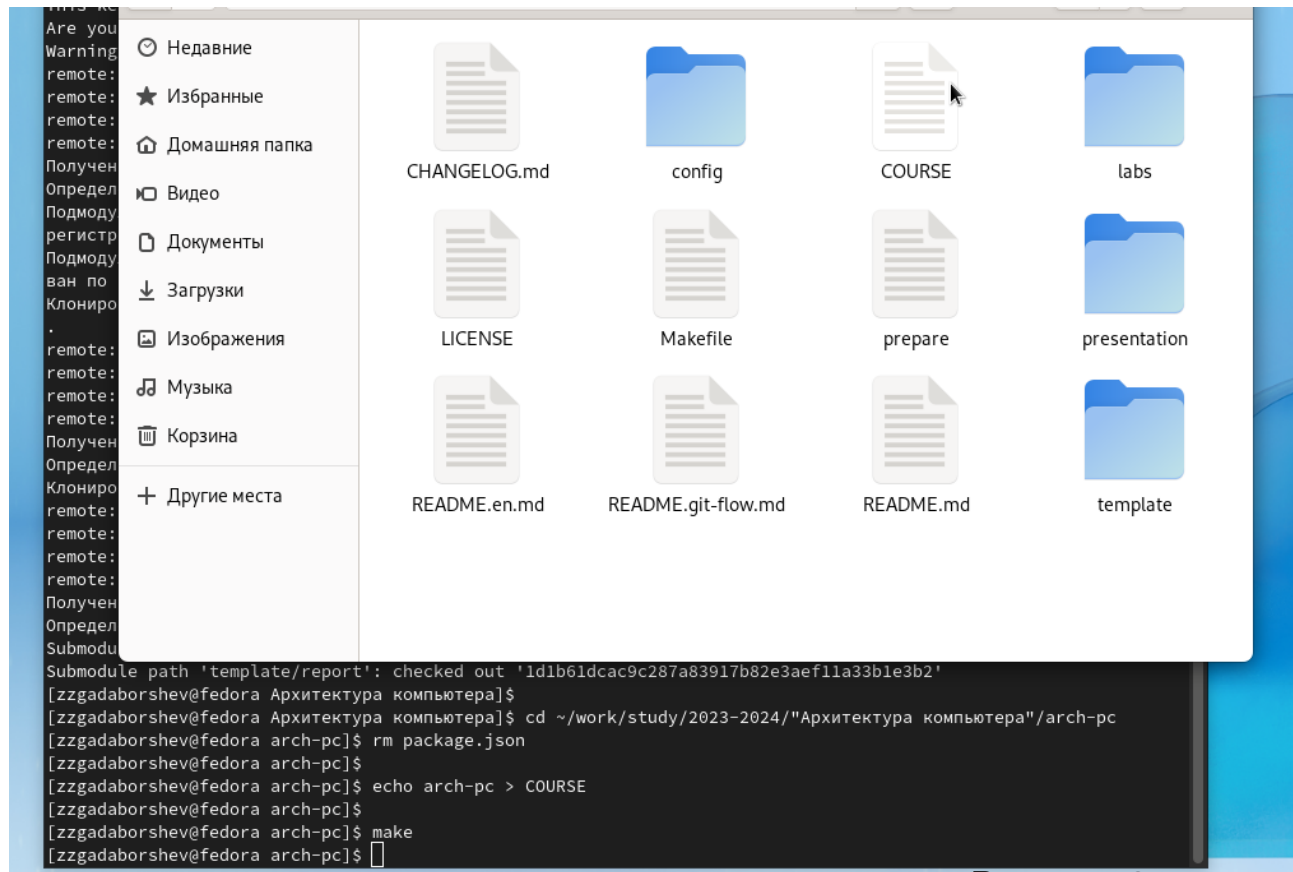


Рисунок 2.1: курс

Отправляю в гитхаб (Рис. 2.2)

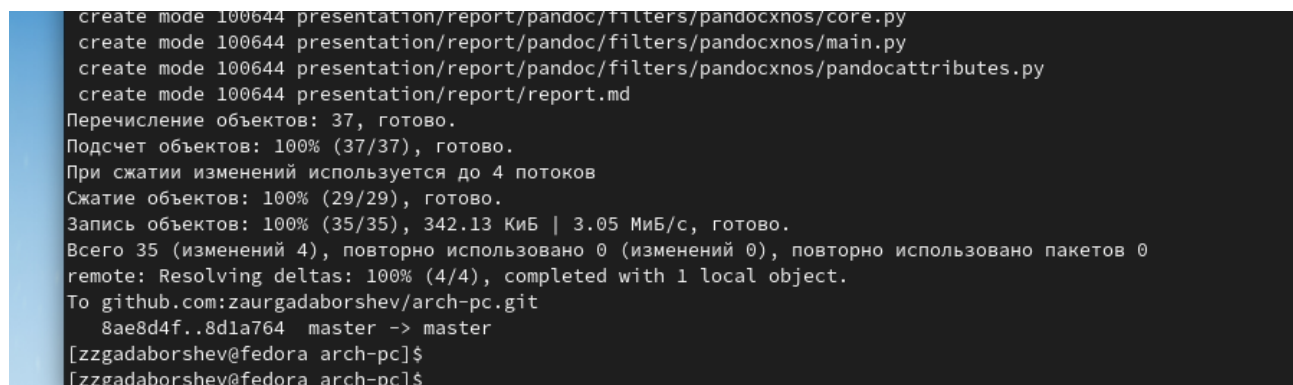


Рисунок 2.2: отправлен в гитхаб

Вывод:

В ходе выполнения работы изучили работу с GitHub.