

Введение

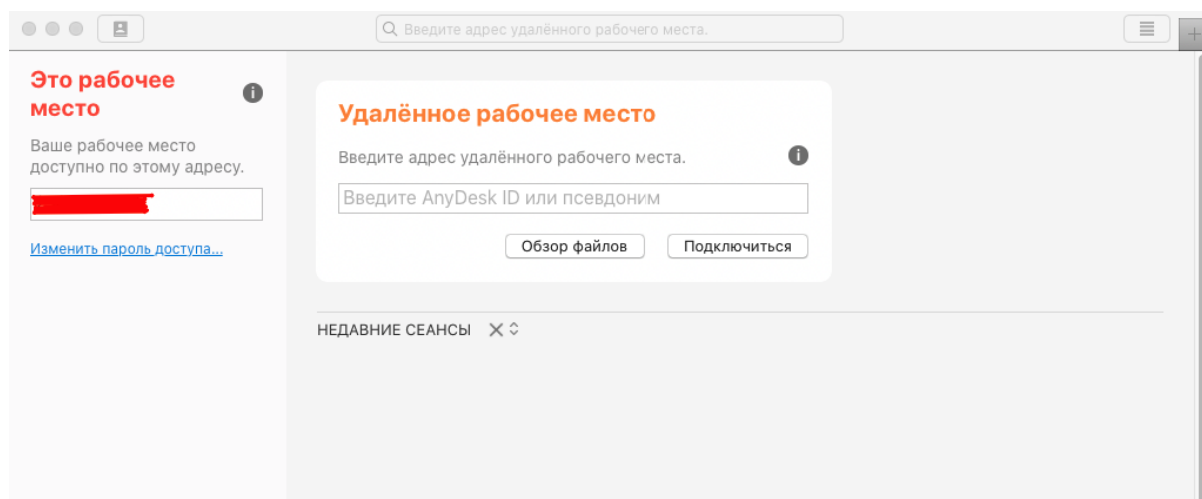
Для выполнения этой и последующих лабораторных работ нам потребуется программа для удаленного доступа AnyDesk. В нашем случае мы не будем ей пользоваться по прямому назначению, а выступать она будет в роли tcp-туннеля ([вики](#)).

Пока мы забудем о том, что мы занимаемся devops, т.к. там для начала необходимо все подготовить (регистрация в удаленном gitlab, jenkins, создание проекта и написания для него тестов).

1. Настройка Anydesk и своего компьютера

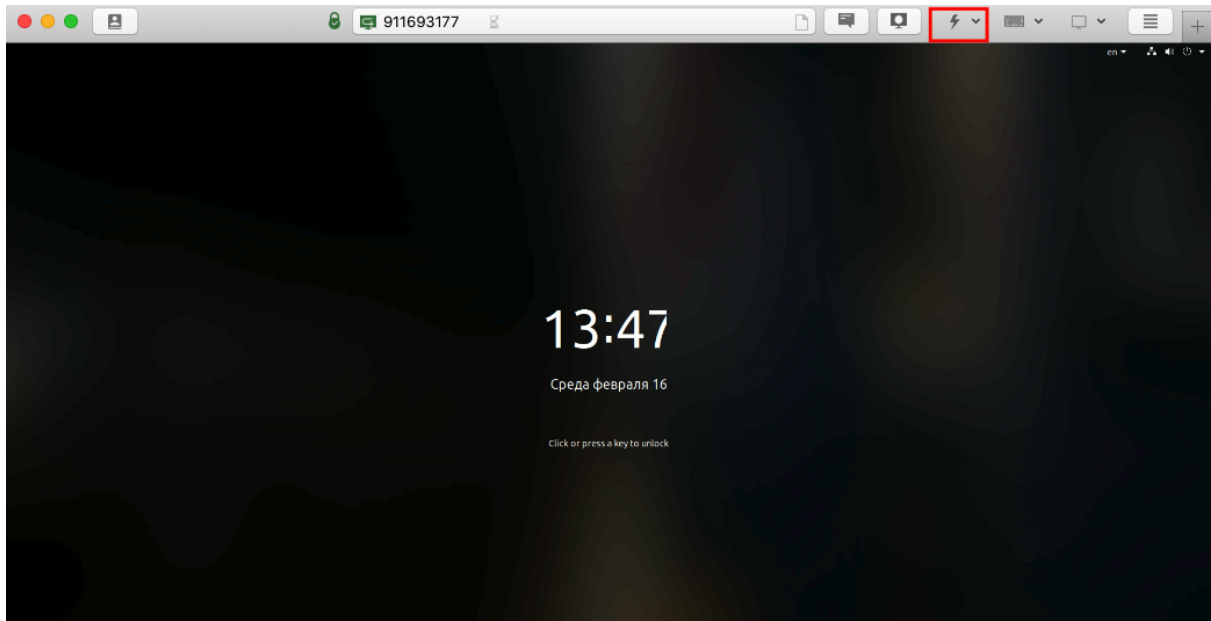
1.1 AnyDesk

Скачиваем AnyDesk на свой компьютер ([оф. сайт](#)) и устанавливаем его. Далее открываем его. Видим следующий интерфейс:

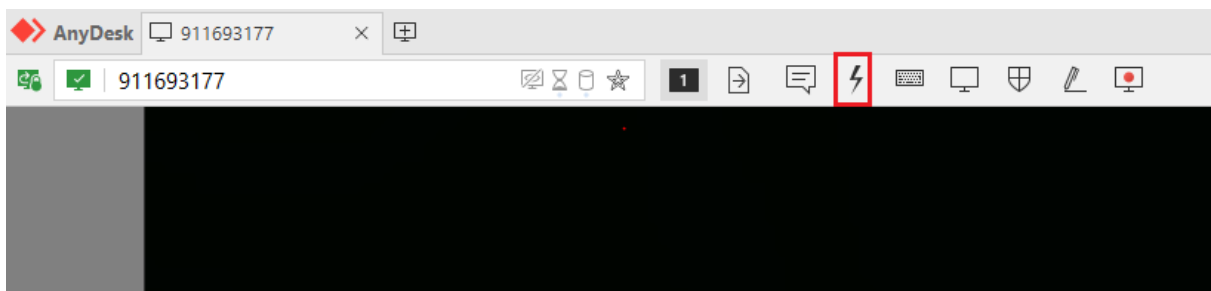


Далее в строку удаленного рабочего места вводим id, далее пароль, предоставленные вам. С AnyDesk мы практически закончили, однако нам необходимо выполнить самую главную вещь - настройка tcp-туннеля.

MacOS:

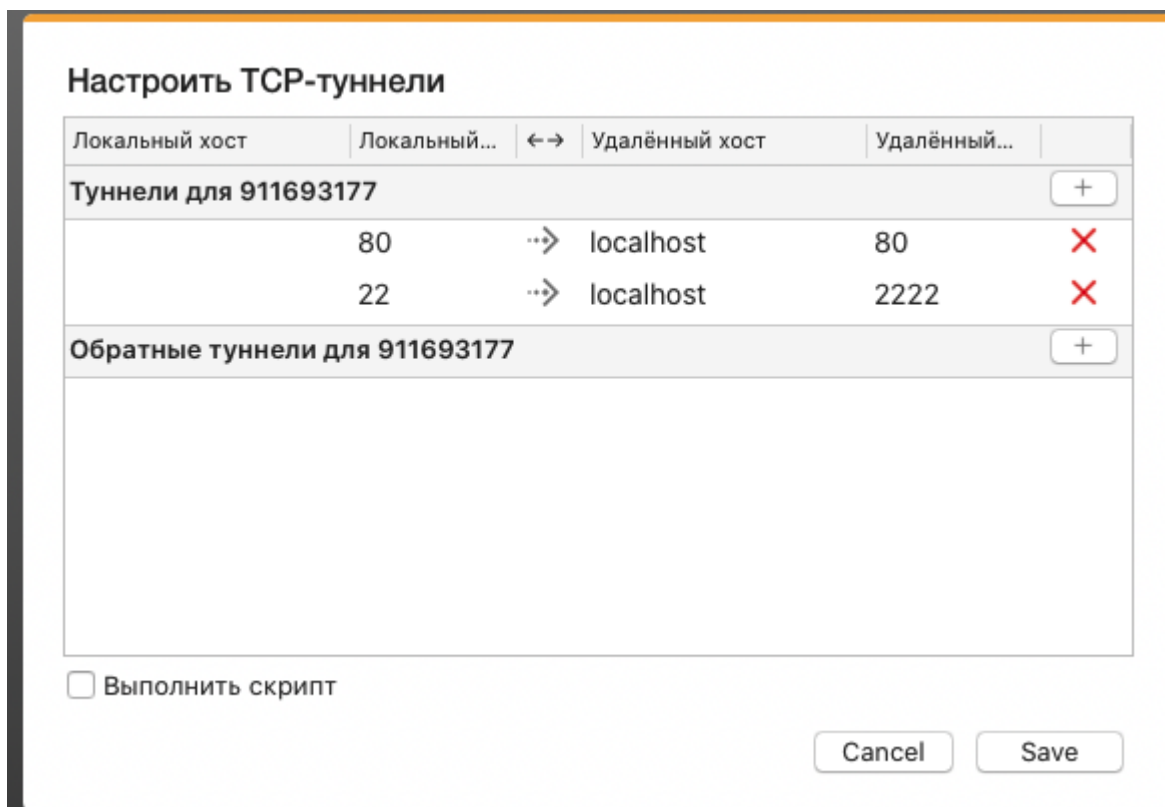


Windows:



В появившемся меню выбираем пункт: *“настроить TCP-туннель”*.

Прописываем порты следующим образом:



С AnyDesk мы закончили, для выполнения лабораторных работ **НЕ ЗАБЫВАЕМ ПОДКЛЮЧАТЬСЯ** к этому компьютеру.

1.2 Настройка предустановленных доменных имен.

Linux и MacOS:

Открываем с помощью терминала файл по пути `/etc/hosts` и добавляем все выделенные записи.

```
nikolay@Nikolays-iMac ~/Desktop> sudo vim /etc/hosts
```

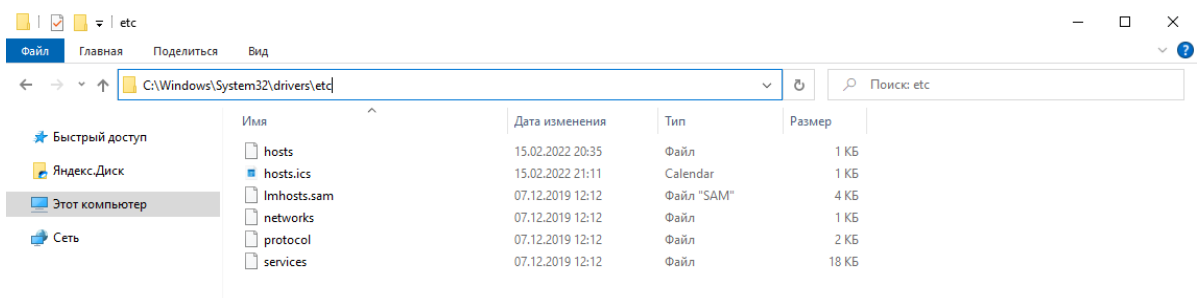
```
##
# Host Database
#
# localhost is used to configure the loopback interface
# when the system is booting. Do not change this entry.
##
127.0.0.1    localhost
255.255.255.255 broadcasthost
::1         localhost

127.0.0.1    devops.ru
127.0.0.1    gitlab.devops.ru
127.0.0.1    jenkins.devops.ru

# Added by Docker Desktop
# To allow the same kube context to work on the host and the container:
127.0.0.1    kubernetes.docker.internal
# End of section
```

Windows:

Переходим по пути “C:\Windows\System32\drivers\etc”.



Открываем в блокноте файл hosts в режиме администратора и добавляем запись по аналогии с Linux.

```
hosts - Блокнот
Файл  Правка  Формат  Вид  Справка
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com           # source server
#       38.25.63.10       x.acme.com               # x client host
#
# localhost name resolution is handled within DNS itself.
#       127.0.0.1         localhost
#       ::1               localhost
127.0.0.1                 devops.ru
127.0.0.1                 gitlab.devops.ru
127.0.0.1                 jenkins.devops.ru
```

Теперь все готово!

2. Регистрация в удаленном GitLab.

Переходим по ссылке: gitlab.devops.ru и мы попадаем в веб интерфейс развернутого на удаленном компьютере, к которому мы подключились с помощью AnyDesk, GitLab.

Далее, как в первой работе, регистрируемся.

Username or email

Password

☐ Remember me

[Forgot your password?](#)

Sign in

Don't have an account yet?

[Register now](#)

После регистрации нас опять перебросит на страницу авторизации. Внимательно читаем появившееся предупреждение. Сначала нам будет необходимо подтвердить вашу учетную запись, после этого вы сможете авторизоваться.

You have signed up successfully. However, we could not sign you in because your account is awaiting approval from your GitLab administrator.

×

GitLab

A complete DevOps platform

GitLab is a single application for the entire software development lifecycle. From project planning and source code management to CI/CD, monitoring, and security.

This is a self-managed instance of GitLab.

Username or email

Password

☐ Remember me

[Forgot your password?](#)

Sign in

Don't have an account yet?

[Register now](#)

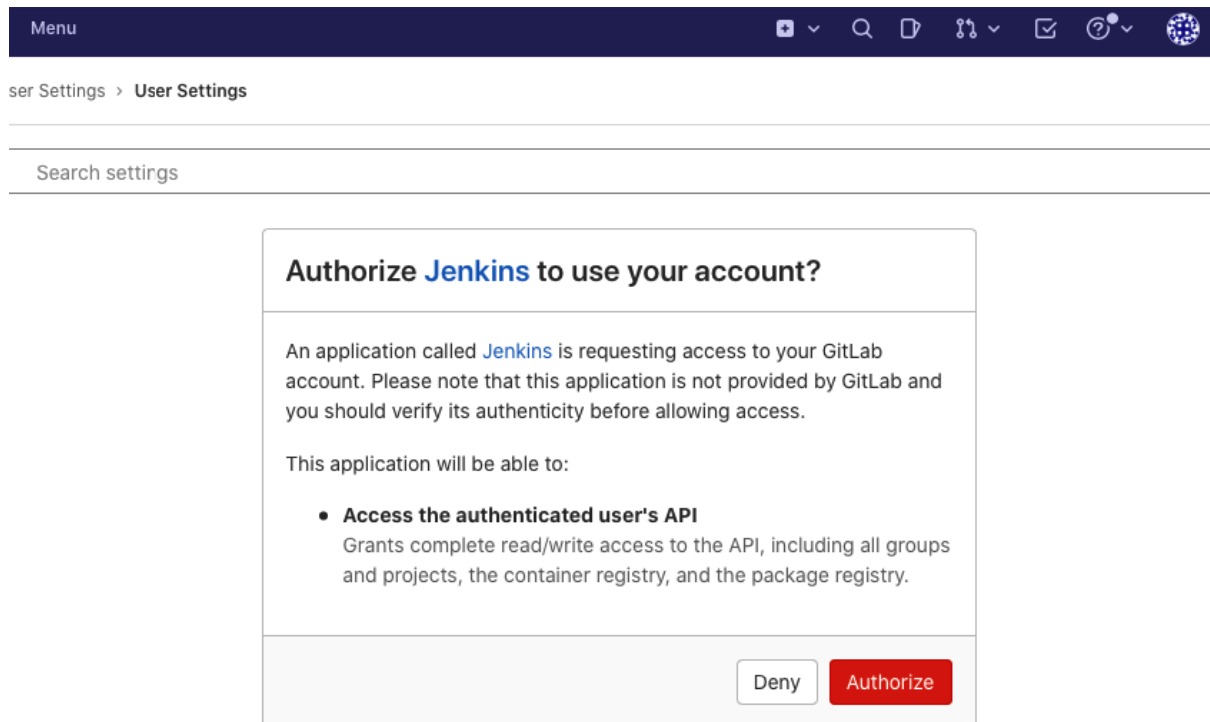
To personalize your GitLab experience, we'd like to know a bit more about you. Don't worry, this information isn't shared outside of your self-managed GitLab instance.

Теперь мы попадаем на стартовую страницу GitLab.



3. Jenkins

Переходим по ссылке: jenkins.devops.ru. Однако вместо привычного интерфейса у нас снова открывается GitLab, но с предложением привязать свою учетную запись GitLab к Jenkins



Нажимаем на “*Authorize*”
Jenkins готов к работе.

Задание

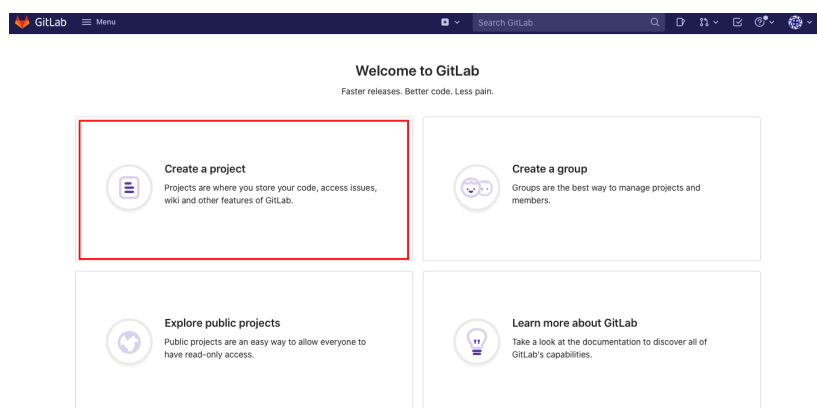
Необходимо будет создать простой REST сервис (без GUI) и написать к нему тесты (желательно воспользоваться встроенным в фреймворк инструментами, либо `pytest + requests`, что не очень правильно).

Выбор серверной библиотеки и задания осуществляется по номеру в журнале в следующей таблице (10 в списке снова берет 1 и т.д.):


№	Фреймворк	Задание
1	FastAPI	Калькулятор
2	FastAPI	Форматирование текста
3	FastAPI	N-ое число ряда простых чисел или ряда Фибоначчи
4	Flask	Калькулятор
5	Flask	Форматирование текста
6	Flask	N-ое число ряда простых чисел или ряда Фибоначчи
7	Tornado	Калькулятор
8	Tornado	Форматирование текста
9	Tornado	N-ое число ряда простых чисел или ряда Фибоначчи

Далее необходимо будет сделать запустить полученный код в репозиторий в гитлаб.


Пример:




Create new project



Create blank project
Create a blank project to house your files, plan your work, and collaborate on code, among other things.



Create from template
Create a project pre-populated with the necessary files to get you started quickly.



Import project
Migrate your data from an external source like GitHub, Bitbucket, or another instance of GitLab.

You can also create a project from the command line. [Show command](#)

New project › **Create blank project**

Project name




Project URL

Project slug

Want to house several dependent projects under the same namespace? [Create a group](#).

Project description (optional)

Visibility Level

- ☒  **Private**
Project access must be granted explicitly to each user. If this project is part of a group, access will be granted to members of the group.
- ☐  **Internal**
The project can be accessed by any logged in user except external users.
- ☐  **Public**
The project can be accessed without any authentication.

Project Configuration

- ☒ **Initialize repository with a README**
Allows you to immediately clone this project's repository. Skip this if you plan to push up an existing repository.

Project 'Example Project' was successfully created.

Example Project

Project ID: 4

1 Commit 1 Branch 0 Tags 61 KB Files 61 KB Storage

master example-project / +

Initial commit
Nikolay Vedernikov authored 1 minute ago

Upload File README Auto DevOps enabled Add LICENSE Add C

Add Kubernetes cluster Configure Integrations

Name	Last commit
README.md	Initial commit

README.md

Example Project

Clone with SSH

git@gitlab.devops.ru:immobilize

Clone with HTTP

http://gitlab.devops.ru/immobilize

Open in your IDE

Visual Studio Code (SSH)

Visual Studio Code (HTTPS)

Пример подготовки кода.

```

nikolay@Nikolays-iMac ~/Desktop> git clone http://gitlab.devops.ru/immobilize/example-project.git
Клонирование в «example-project»...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Получение объектов: 100% (3/3), готово.
nikolay@Nikolays-iMac ~/Desktop> cd example-project/
nikolay@Nikolays-iMac ~/D/example-project (master)> ls
README.md
nikolay@Nikolays-iMac ~/D/example-project (master)> vim main.py
nikolay@Nikolays-iMac ~/D/example-project (master)> pip install pytest
Requirement already satisfied: pytest in /Users/nikolay/opt/anaconda3/lib/python3.8/site-packages (0.0.0)
Requirement already satisfied: attrs>=17.4.0 in /Users/nikolay/opt/anaconda3/lib/python3.8/site-packages (from pytest) (20.3.0)
Requirement already satisfied: iniconfig in /Users/nikolay/opt/anaconda3/lib/python3.8/site-packages (from pytest) (1.1.1)
Requirement already satisfied: packaging in /Users/nikolay/opt/anaconda3/lib/python3.8/site-packages (from pytest) (20.4)
Requirement already satisfied: pluggy<1.0,>=0.12 in /Users/nikolay/opt/anaconda3/lib/python3.8/site-packages (from pytest) (0.13.1)
Requirement already satisfied: py>=1.8.2 in /Users/nikolay/opt/anaconda3/lib/python3.8/site-packages (from pytest) (1.9.0)
Requirement already satisfied: toml in /Users/nikolay/opt/anaconda3/lib/python3.8/site-packages (from pytest) (0.10.1)
Requirement already satisfied: pyparsing>=2.0.2 in /Users/nikolay/opt/anaconda3/lib/python3.8/site-packages (from packaging->pytest) (2.4.7)
Requirement already satisfied: six in /Users/nikolay/opt/anaconda3/lib/python3.8/site-packages (from packaging->pytest) (1.15.0)
WARNING: You are using pip version 21.2.2; however, version 22.0.3 is available.
You should consider upgrading via the '/Users/nikolay/opt/anaconda3/bin/python3 -m pip install --upgrade pip' command.
nikolay@Nikolays-iMac ~/D/example-project (master)> vim test_main.py
nikolay@Nikolays-iMac ~/D/example-project (master)> vim test_main.py
nikolay@Nikolays-iMac ~/D/example-project (master)> pytest test_main.py
===== test session starts =====
platform darwin -- Python 3.8.5, pytest-6.1.1, py-1.9.0, pluggy-0.13.1
rootdir: /Users/nikolay/Desktop/example-project
plugins: typeguard-2.12.0
collected 1 item

test_main.py .

===== 1 passed in 0.02s =====
nikolay@Nikolays-iMac ~/D/example-project (master)>

```

Файл main.py:

```
def sum_fn(fst_arg, snd_arg):  
    return fst_arg + snd_arg
```

Файл test_main.py:

```
from main import sum_fn  
  
def test_sum_fn():  
    assert sum_fn(2, 2) == 4  
    assert sum_fn(0, -1) == -1  
    assert sum_fn(100, -100) == 0
```

Пушим получившийся результат в созданный репозиторий.

```
nikolay@Nikolays-iMac ~/D/example-project (master)> ls  
README.md  __pycache__  main.py      test_main.py  
nikolay@Nikolays-iMac ~/D/example-project (master)> git add main.py  
nikolay@Nikolays-iMac ~/D/example-project (master)> git commit -m "add main file"  
[master 82ee590] add main file  
1 file changed, 2 insertions(+)  
create mode 100644 main.py  
nikolay@Nikolays-iMac ~/D/example-project (master)> git add test_main.py  
nikolay@Nikolays-iMac ~/D/example-project (master)> git commit -m "add test for main file"  
[master d32da73] add test for main file  
1 file changed, 7 insertions(+)  
create mode 100644 test_main.py  
nikolay@Nikolays-iMac ~/D/example-project (master)> git push  
Перечисление объектов: 7, готово.  
Подсчет объектов: 100% (7/7), готово.  
При сжатии изменений используется до 12 потоков  
Сжатие объектов: 100% (6/6), готово.  
Запись объектов: 100% (6/6), 611 байтов | 611.00 КиБ/с, готово.  
Всего 6 (изменений 1), повторно использовано 0 (изменений 0), повторно использовано пакетов 0  
To http://gitlab.devops.ru/immobilize/example-project.git  
0bf0221..d32da73  master -> master  
nikolay@Nikolays-iMac ~/D/example-project (master)>
```

Ссылки:

1. [What is a REST API?](#)
2. [Введение в REST API — RESTful веб-сервисы](#)
3. [Общение микросервисов: REST, JSON, GraphQL или gRPC \(Их история появления и зачем они нужны\) видео](#)
4. [Документация по FastAPI](#)
5. [Документация по Flask](#)
6. [Документация по Tornado](#)