МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»

Факультет «Радіоелектроніки, комп’ютерних систем та інфокомунікацій» Кафедра «Аерокосмічних радіоелектронних систем»

**Лабораторна робота №1**

з дисципліни «Інформаційно-комунікаційні мережі » на тему

«Основи роботи з Git»

Виконав: студент 4 курсу групи № 546 напряму підготовки (спеціальності) 172 «Телекомунікації та радіотехніка»

Мельник І.

Чуніхін В.

Прийняв: ас. каф. 501 Перетятько М. С.

Національна шкала: Кількість балів:

Оцінка: ECTS

Харків 2021

**Ціль роботи:** Навчитися створювати репозиторії на GitHub, працювати з ssh ключами, оформити профіль.

Теоретична довідка

Основи Git – Отримання репотизорія Git

Зазвичай репозиторій Git можна отримати одним з двох способів:

Ви можете взяти локальний каталог, який в даний час не перебуває під контролем версій, і перетворити його в репозиторій Git, або

Ви можете клонувати існуючий репозиторій Git з будь-якого місця.

У будь-якому випадку ви отримаєте репозиторій Git на вашому локальному комп'ютері, готовий до роботи.

Ініціалізація репотизорія в існуючому каталозі

Якщо у вас є каталог проекту, який в даний час не перебуває під контролем версій, і ви хочете почати керувати ним за допомогою Git, вам спочатку потрібно перейти в каталог цього проекту. Якщо ви ніколи цього не робили, це виглядає трохи по-різному в залежності від того, в якій системі ви працюєте:

для Windows:

$ cd C:/Users/user/my\_project

и введите:

$ git init

Клонування існуючого репозіторія

Якщо ви хочете отримати копію існуючого сховища Git - наприклад, проекту, в який ви хочете внести свій внесок, - вам потрібна команда git clone. Якщо ви знайомі з іншими VCS, такими як Subversion, ви помітите, що це команда «clone», а не «checkout». Це важлива відмінність - замість того, щоб отримати просто робочу копію, Git отримує повну копію майже всіх даних, які є на сервері. Кожна версія кожного файлу історії проекту позамовчуванням видаляється при запуску git clone. Фактично, якщо ваш серверний диск пошкоджений, ви часто можете використовувати майже будь-який з клонів на будь-якого клієнта, щоб повернути сервер в той стан, в якому він був, коли він був клонований (ви можете втратити деякі серверні перехоплювачі тощо, але все версійні дані будуть там - см. Отримання Git на сервері Більше подробиць).

Ви клонуєте репозиторій за допомогою git clone <url>. Наприклад, якщо ви хочете клонувати звану підключається бібліотеку Git libgit2, ви можете зробити це в такий спосіб:

$ git clone https://github.com/libgit2/libgit2

Це створює каталог з ім'ям libgit2, инициализирует .gitкаталог всередині нього, витягує всі дані для цього сховища та перевіряє робочу копію останньої версії. Якщо ви ввійдете в новий libgit2каталог, який був тільки що створений, ви побачите там файли проекту, готові до роботи або використання.

Якщо ви хочете клонувати репозиторій в каталог з ім'ям, відмінним від libgit2, ви можете вказати нове ім'я каталогу в якості додаткового аргументу:

$ git clone https://github.com/libgit2/libgit2 mylibgit

Ця команда робить те ж саме, що і попередня, але викликається цільової каталог mylibgit.

У Git є кілька різних протоколів передачі, які ви можете використовувати. У попередньому прикладі використовується https: // протокол, але ви також можете побачити git: // або user @ server: path / to / repo.git, який використовує протокол передачі SSH. При установці Git на сервері будуть представлені всі доступні параметри, які сервер може налаштувати для доступу до вашого сховища Git, а також плюси і мінуси кожного з них.

Основи Git - Додавання тегів

Як і більшість VCS, Git має можливість позначати певні точки в історії сховища як важливі. Як правило, люди використовують цю функцію, щоб відзначити пункти випуску (v1.0, v2.0і так далі). У цьому розділі ви дізнаєтеся, як скласти список існуючих тегів, як створювати і видаляти теги і які бувають типи тегів.

Розміщення ваших тегів

Перерахувати існуючі теги в Git просто. Просто введіть git tag (необов'язково -lілі --list):

$ Git tag

v1.0

v2.0

Ця команда перераховує теги в алфавітному порядку; порядок, в якому вони відображаються, не має реального значення.

Ви також можете шукати теги, що відповідають певним шаблоном. Наприклад, репозиторій вихідного коду Git містить понад 500 ключових слів. Якщо вас цікавить тільки серія 1.8.5, ви можете запустити це:

$ git tag -l "v1.8.5\*"

v1.8.5

v1.8.5-rc0

v1.8.5-rc1

v1.8.5-rc2

v1.8.5-rc3

v1.8.5.1

v1.8.5.2

v1.8.5.3

v1.8.5.4

v1.8.5.5

ХІД РОБОТИ

1. Створіть обліковий запис GitHub

<https://github.com/join?ref_cta=Sign+up&ref_loc=header+logged+out&ref_page=%2F&source=header-home>

1. Встановіть CHOCOLATEY:

* запустить powershell відімені адміністратора
* виконайте команду:

Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))

1. Встановить git на ваш комп'ютер за допомогою Chocolatey:

choco install git -y

Можно использовать автоматический git bash для роботов с git, а затем переварить ConEmu. Это программа підтримує гарячі клавіші Ctrl + c і Ctrl + v та приємніша у користуванні. Для того, чтобы установить ConEmu необязательно виконати команду:

choco install conemu

1. Налаштуйте свій git:

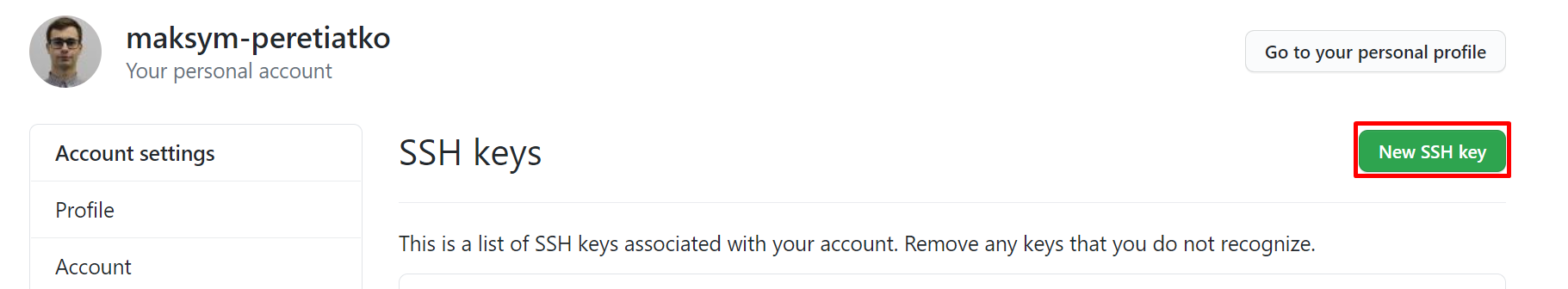
git config --global user.name "John Doe"

git config --global user.email johndoe@example.com

1. Згенеруйте новый SSH ключ:

ssh-keygen -t ed25519 -C "your\_email@example.com"

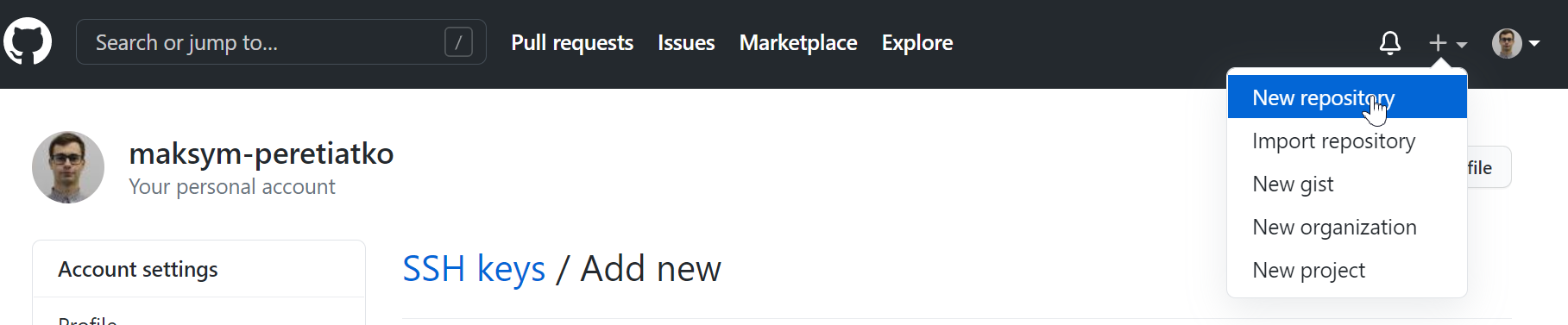
1. Додайте ключ до облікового запису Github:

* зкопіюйте /c/Users/you/.ssh/id\_ed25519.pub
* перейдіть за посылку <https://github.com/settings/keys>
* натисніть зелену [](https://github.com/maksym-peretiatko/labs-itm/blob/labs-ukr/lab1/Screenshot_2.png)
* Вставьте зкопійований ключ та натисніть відповідну кнопку

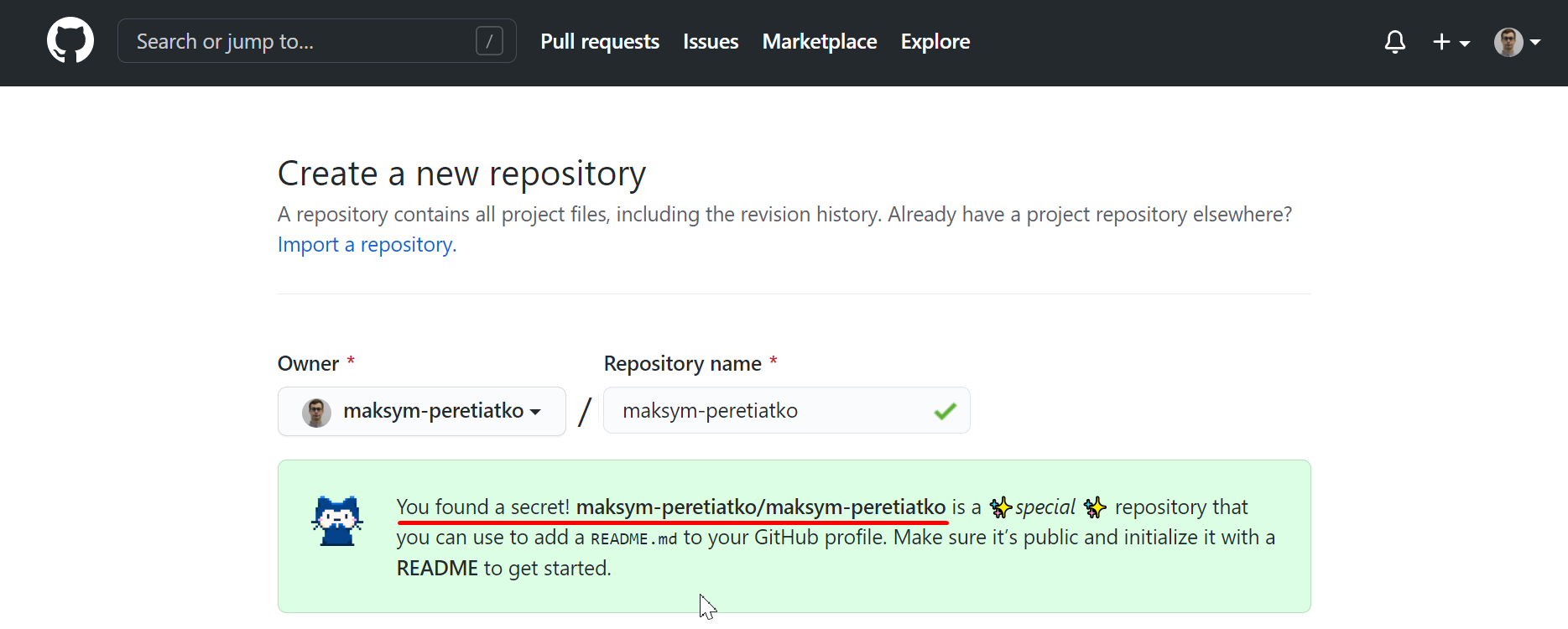
Подробнее інформації: <https://docs.github.com/en/github/authenticating-to-github/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent>

Тепер використовати GitHub

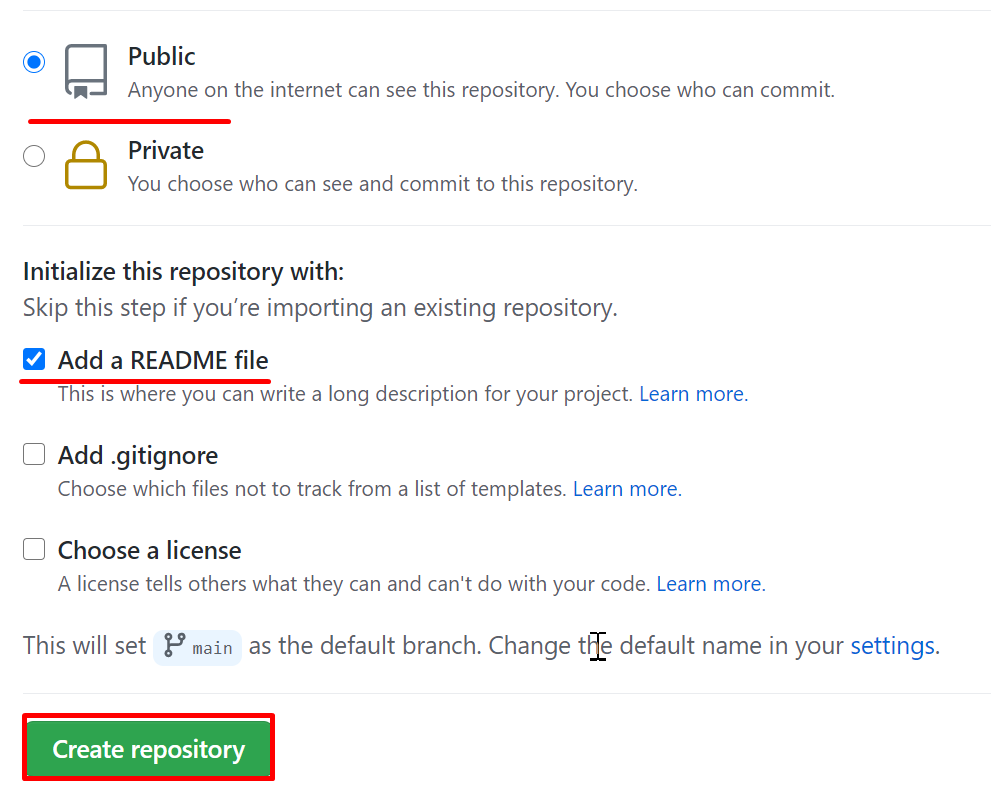
1. Створіть новий репозиторій

[](https://github.com/maksym-peretiatko/labs-itm/blob/labs-ukr/lab1/Screenshot_4.png)

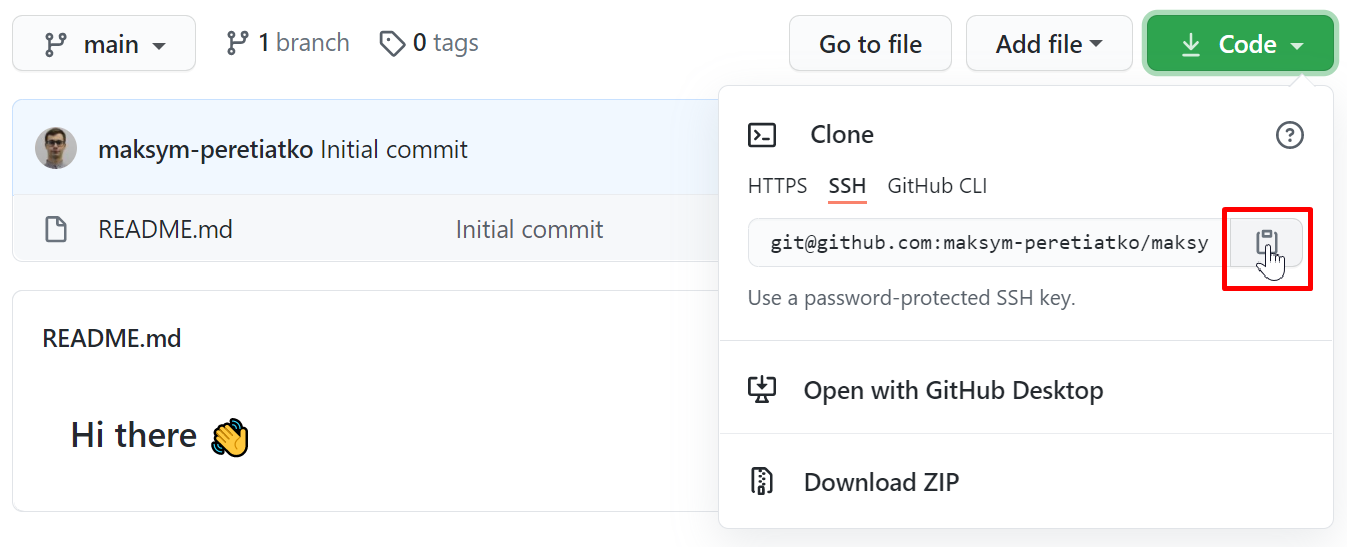
Назва репозиторію має бути такою як ваш нікнейм

[](https://github.com/maksym-peretiatko/labs-itm/blob/labs-ukr/lab1/Screenshot_5.png)

Переконайтеся, репозиторий публикаций и прохождений из readme.md файлом

[](https://github.com/maksym-peretiatko/labs-itm/blob/labs-ukr/lab1/Screenshot_6.png)

1. Забереть копію репозитория на локальный комп'ютер:

* зкопіюйте посилання [](https://github.com/maksym-peretiatko/labs-itm/blob/labs-ukr/lab1/Screenshot_1.png)
* запустить команду для изменения репозиторію:

git clone

У моєму випадку це:

git clone git@github.com:maksym-peretiatko/maksym-peretiatko.git

1. Відкрийте README.md вашим улюбленим текстовим редактором.

**максым-перетятко / максым-перетятко** - это✨ специальный ✨репозиторий, потому что его README.md(этот файл) отображается в вашем профиле GitHub.

Вот несколько идей для начала:

* 🔭 Я сейчас работаю над ...
* 🌱 Я сейчас учусь ...
* 👯 Я хочу сотрудничать над ...
* 🤔 Мне нужна помощь с ...
* 💬 Спроси меня о ...
* 📫 Как до меня добраться: ...
* 😄 Местоимения: ...
* ⚡ Интересный факт: ...

Вы можете скористатися цим шаблоном или створити власний.

1. Напишіть про себе

* З чим хотіли б попрацювати;
* Чим захоплюєтесь;
* Дайте посылку на своей социальной мережі та ресурсы, які вважаєте необхідними или бажаними

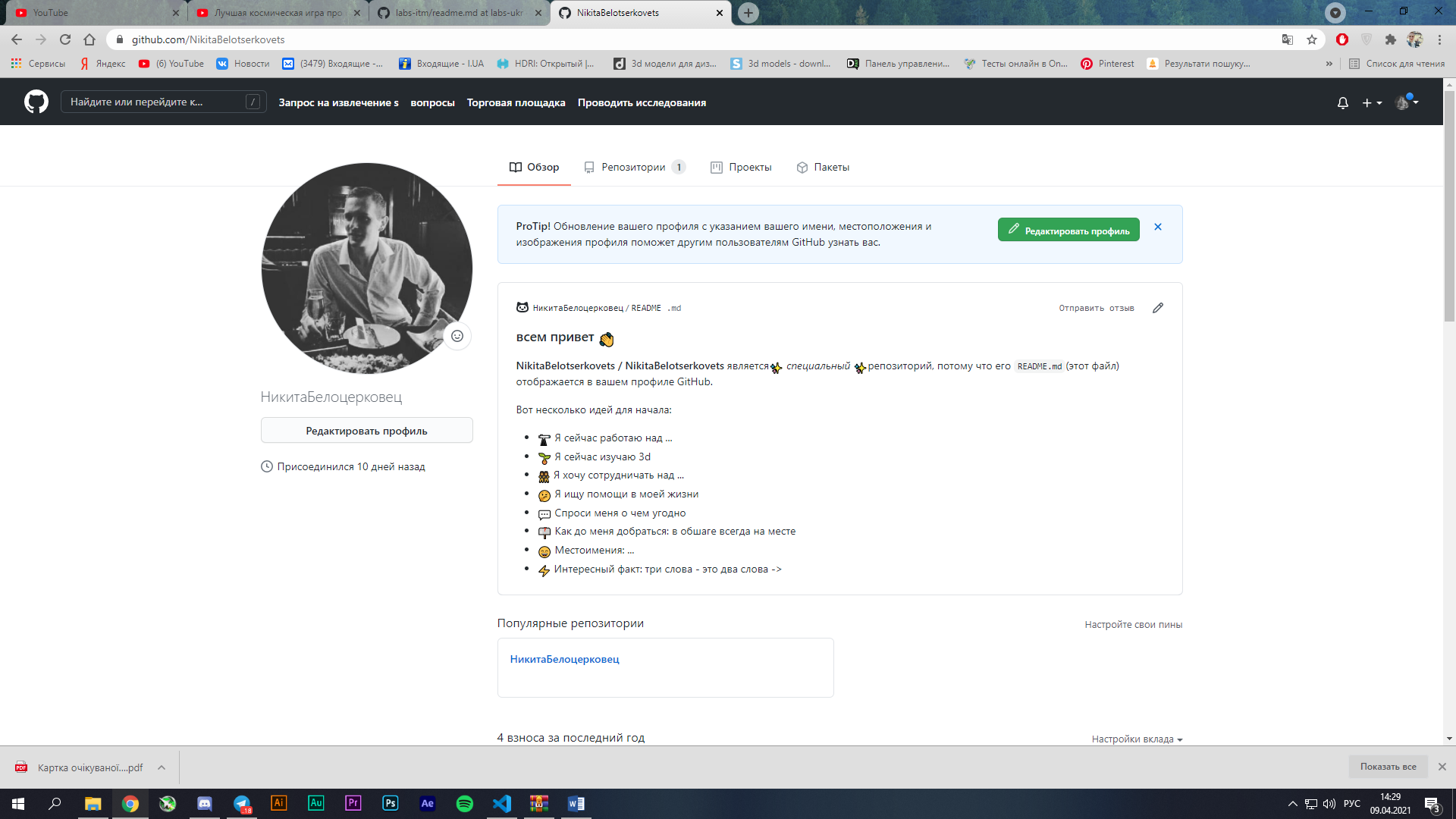
1. Загрузить опис вашого профілю:

git add .

git commit -m "Add your comment"

git push

ВИСНОВКИ



Зі скриншота ми бачимо результат роботи. В ході лабораторної роботи розібрався з GIT. Також освоїв додаткові інструменти для більш поглибленої роботи, такі як ConEmu, GitBash і інші. Навчився створювати репозиторії на GitHub, працювати з ssh ключами, оформляти профіль