

Самостоятельная работа №4. Использование Git в других окружениях.

Самостоятельная работа №4.1. Создание нового репозитория в GitHub


Регистрация на GitHub и создание публичного репозитория

После регистрации в GitHub (веб-сервис для хостинга IT-проектов и их совместной разработки, основанной на Git) в выпадающем меню, появляющемся при нажатии на знак «+» в верхнем правом углу, выбирается пункт «New Repository». В новом окне вводится следующая информация:




- **Название репозитория:** «ITMOUniversity-web-developer-Module_5-independent-work_4».
- **Описание репозитория:** «Самостоятельная работа № 4 „Использование Git в других окружениях“ к модулю 5 „Средства организации процесса веб-разработки в команде“ курса „Веб-разработчик 20.35“ от Университета ИТМО».
- **Тип репозитория (Public, Private):** Public.
- **Добавление файла README.md в репозиторий** (и создание первого коммита Initial Commit).
- **Тип лицензии проекта:** MIT Licence

После заполнения всей информации нажимается кнопка «Create Repository», которая создаст репозиторий:

←→↻🔒https://github.com/new🌟🔍Поиск📧📄📁📧🔧⋮

Search or jump to...

PullsIssuesMarketplaceExplore




Create a new repository

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

Owner ^{*}

Repository name ^{*}

InventiveSpark ▾


/


Module_5-independent-work_4 ✓

Great repository names are short and memorable. Need inspiration? How about [fluffy-computing-machine?](#)

Description (optional)

Самостоятельная работа №4. «Использование Git в других окружениях» к модулю 5 «Средства организаци

☒  **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.

Initialize this repository with:


Skip this step if you're importing an existing repository.

☒ **Add a README file**
This is where you can write a long description for your project. [Learn more.](#)

☐ **Add .gitignore**
Choose which files not to track from a list of templates. [Learn more.](#)

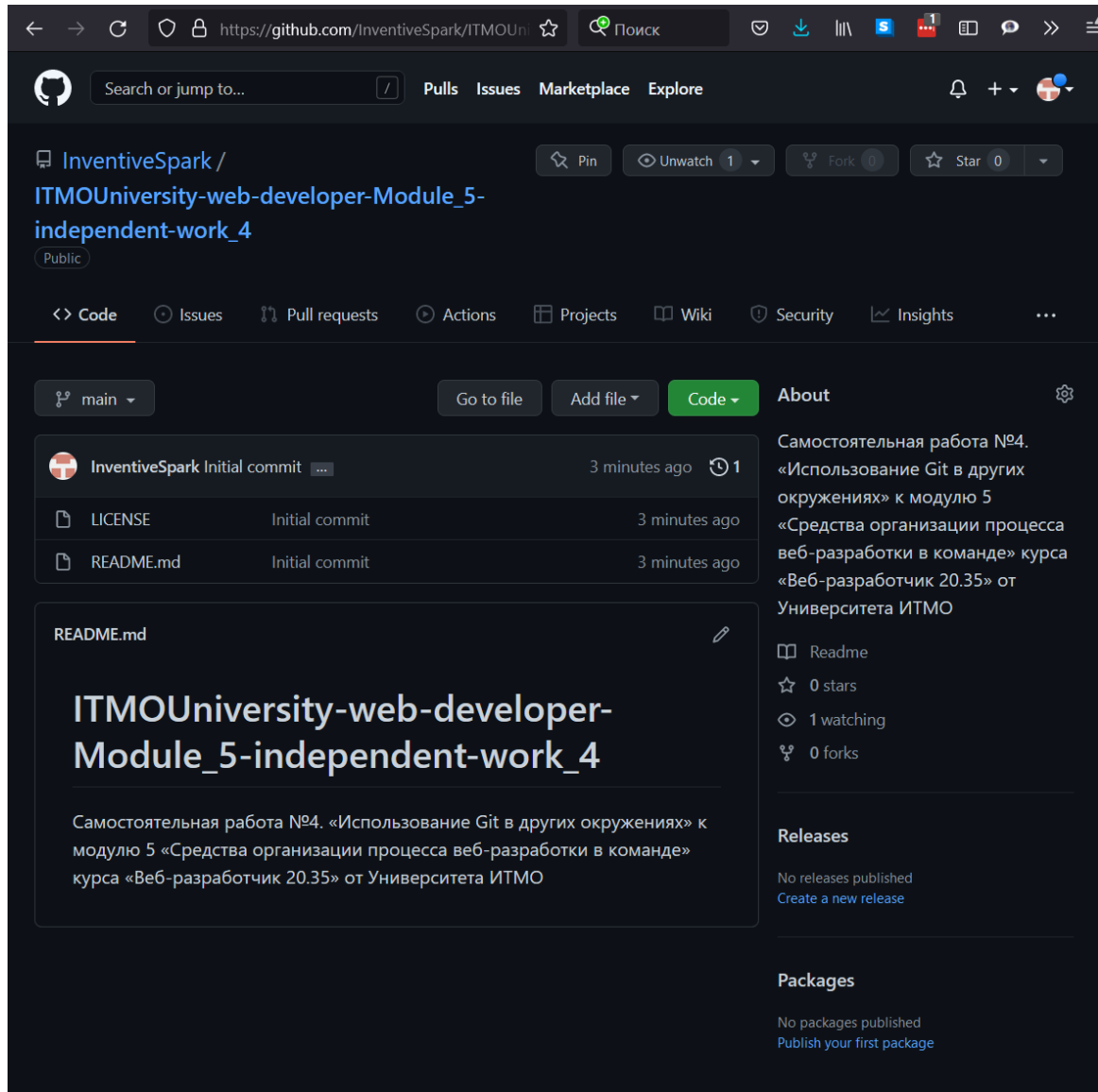
☒ **Choose a license**
A license tells others what they can and can't do with your code. [Learn more.](#)

License: MIT License ▾

This will set  main as the default branch. Change the default name in your [settings](#).

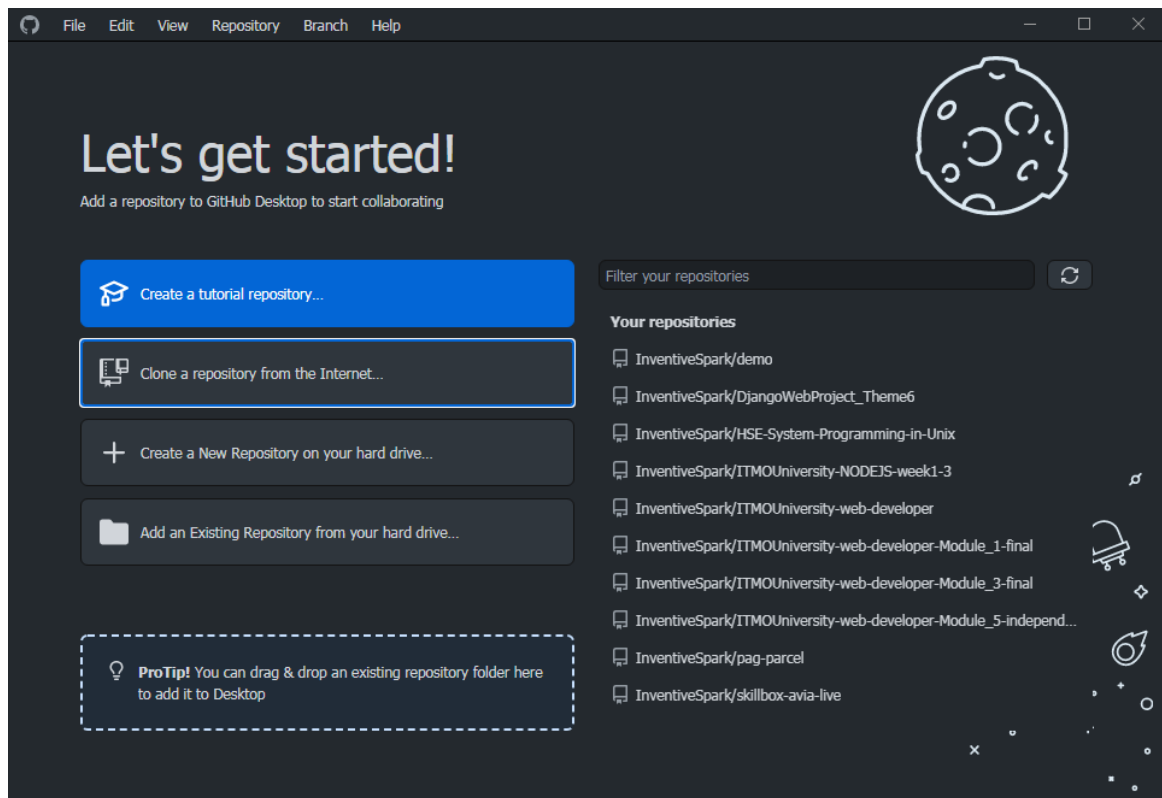
Create repository

После создания репозитория автоматически открывается его страница в GitHub. На странице репозитория отображается первый коммит Initial commit, содержащий файл выбранной лицензии, а также файл README.md с кратким описанием, написанным на языке разметки Markdown:



Клонирование репозитория с помощью утилиты GitHub Desktop

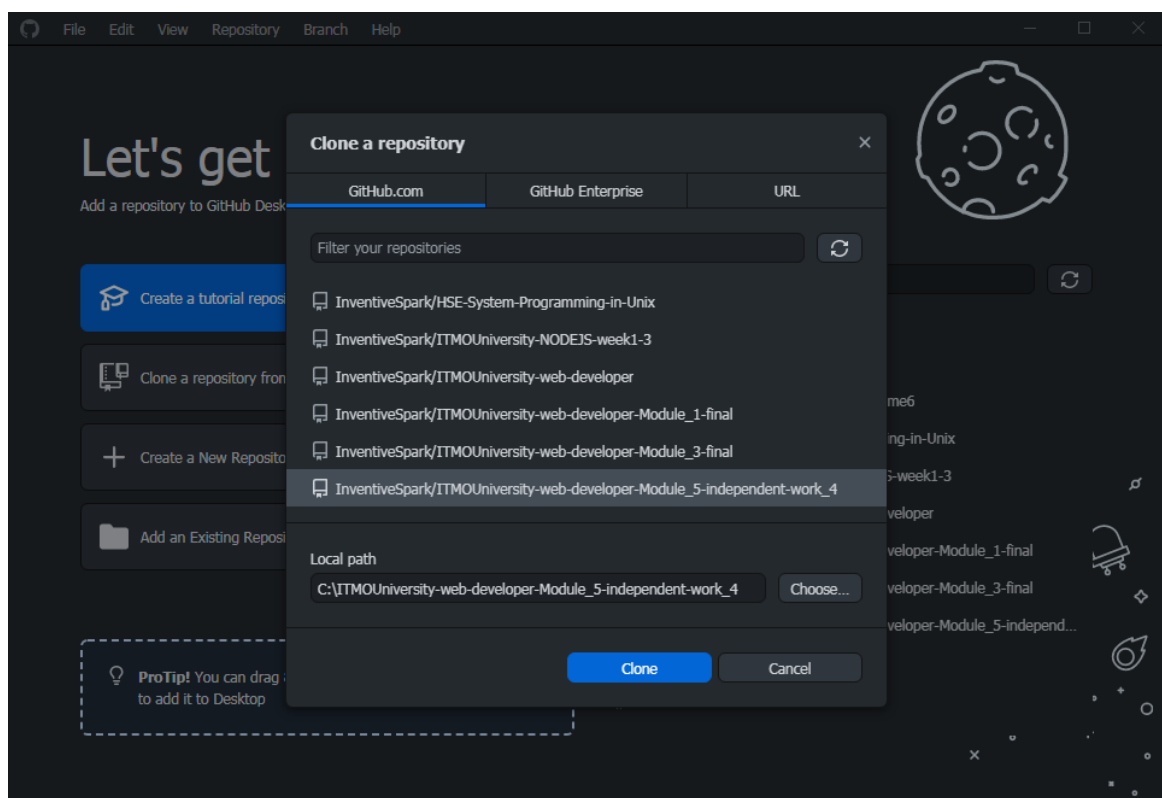
Для клонирования удаленного репозитория выбирается пункт «Clone a repository from the Internet...»:



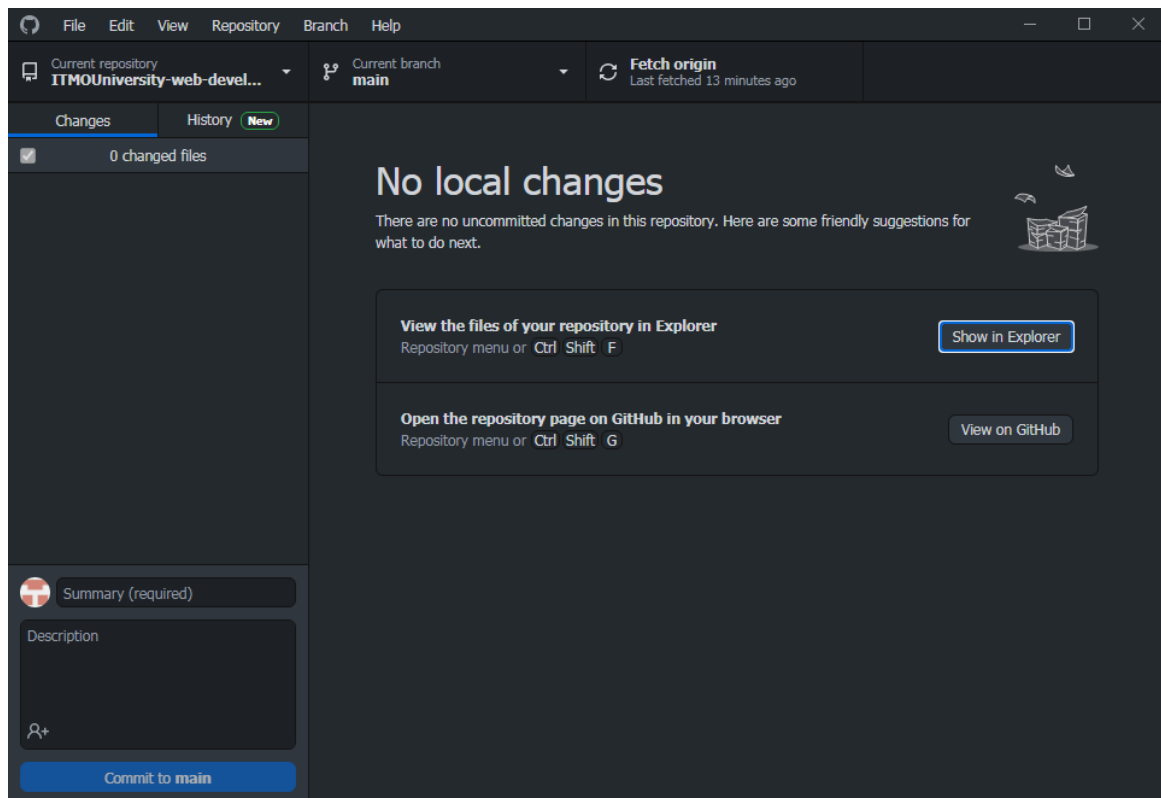
В новом окне во вкладке GitHub.com выбираются:

- репозиторий для клонирования к себе на персональный компьютер;
- каталог для хранения этого репозитория в компьютере.

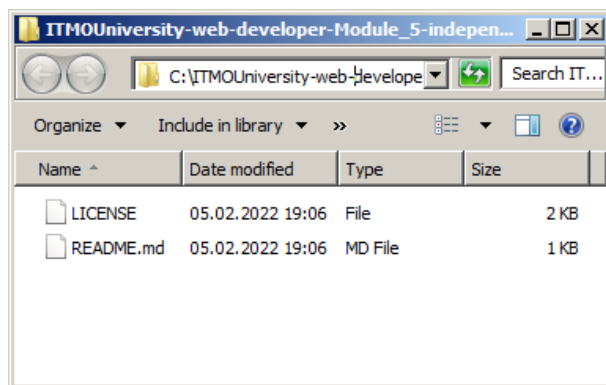
После нажатия на кнопку «Clone» запустится процесс клонирования репозитория:



По завершению клонирования появится окно проекта:



Нажатие на кнопку «Show in Explorer» запустит Проводник Windows, в котором будет открыт каталог клонированного репозитория:



Создание непустого текстового файла с именем hello.txt

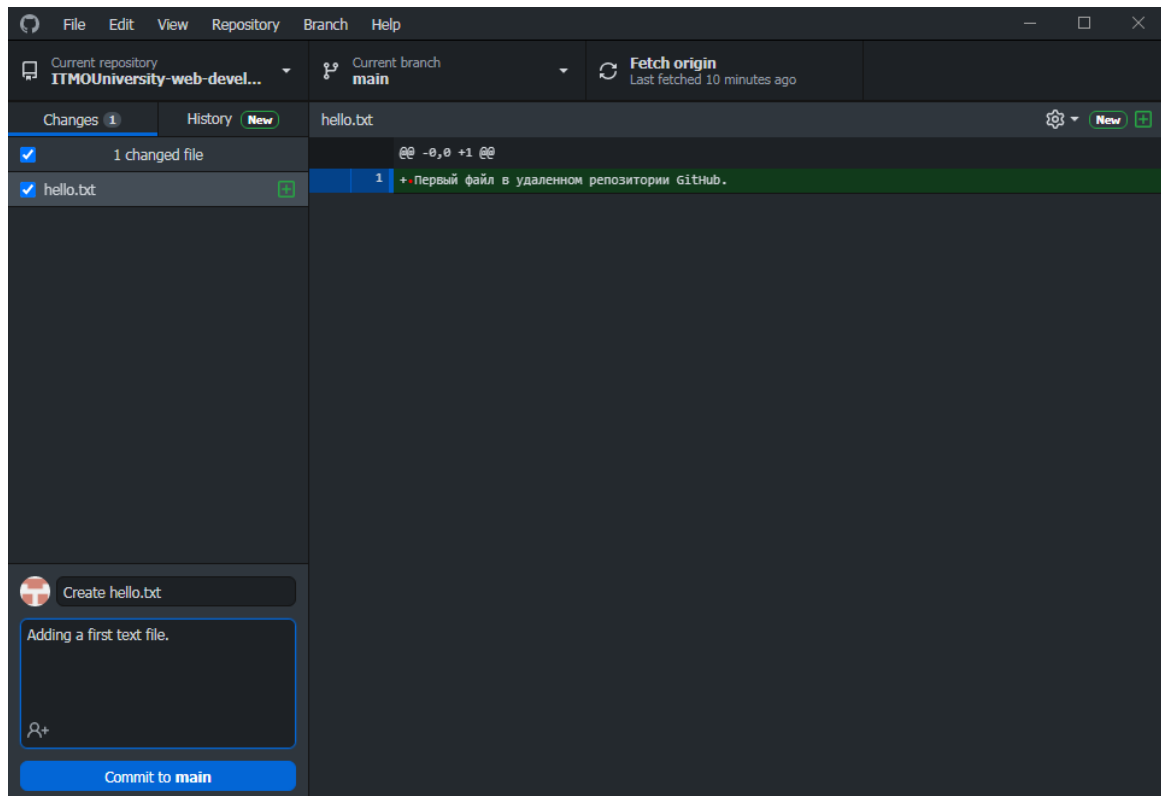
В каталоге репозитория создается файл hello.txt со следующим содержимым: «Первый файл в удаленном репозитории GitHub».

В левой колонке окна программы Github Desktop будут представлены все файлы проекта, которые были изменены. После нажатия на название файла, в правом поле основного окна программы можно увидеть изменения, сделанные в этом файле.

Для создания коммита под именем «Create hello.txt» (фиксация индексированных файлов с комментарием в историю) в форму, расположенную в левом нижнем углу программы, вводится следующая информация:

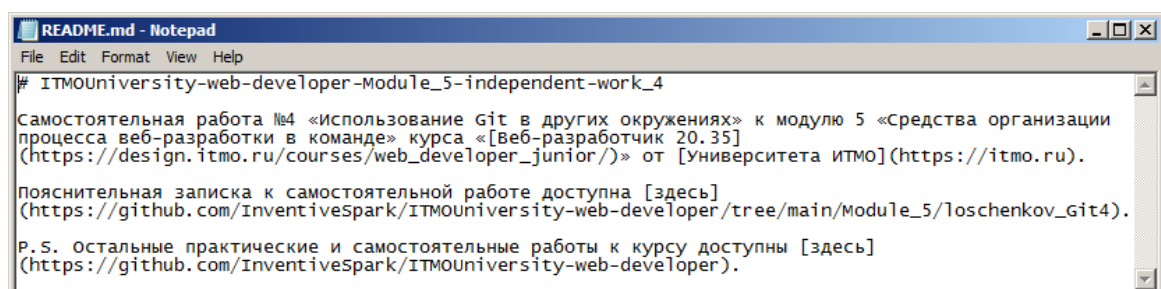
- **Название коммита:** «Create hello.txt».
- **Описание изменений, произведенных в коммите:** «Adding a first text file».

После заполнения всей информации, нажимается кнопка «Commit to main», которая выполнит коммит в ветку main (master) локального репозитория:

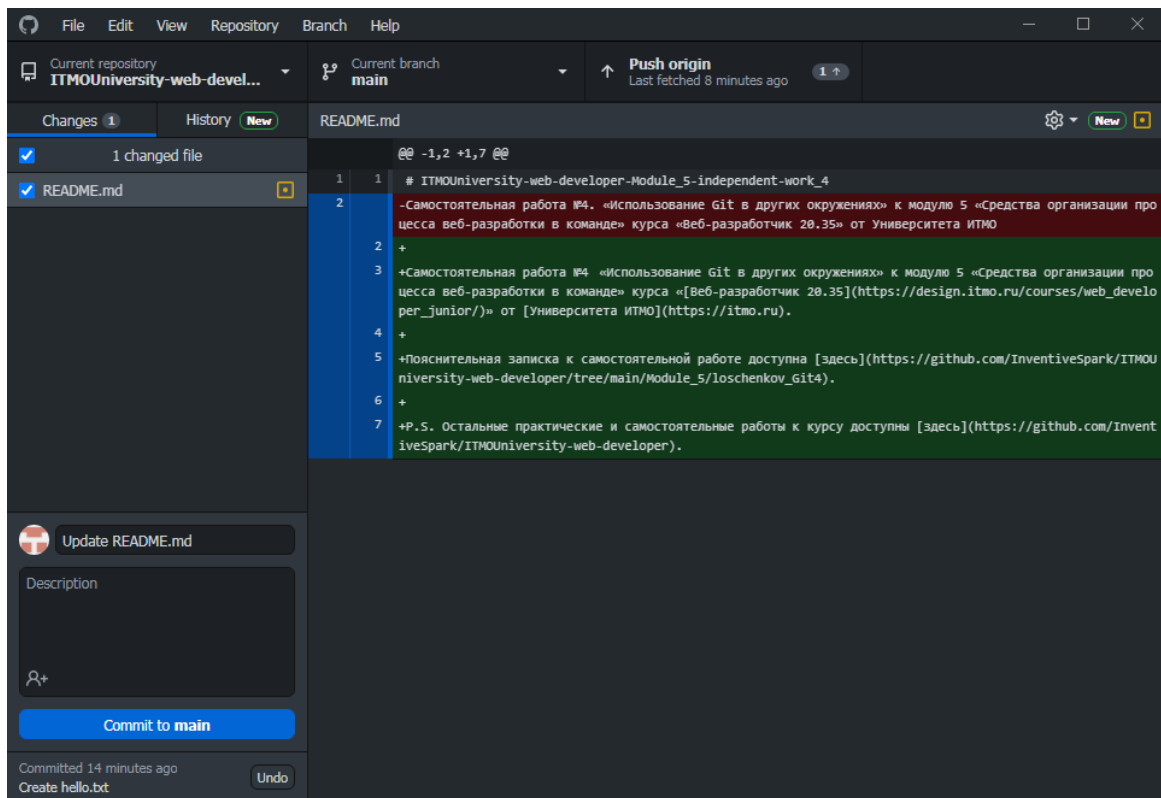


Редактирование файла Readme.md

При помощи текстового редактора файл README.md дополняется подробным описанием репозитория на языке разметки Markdown:

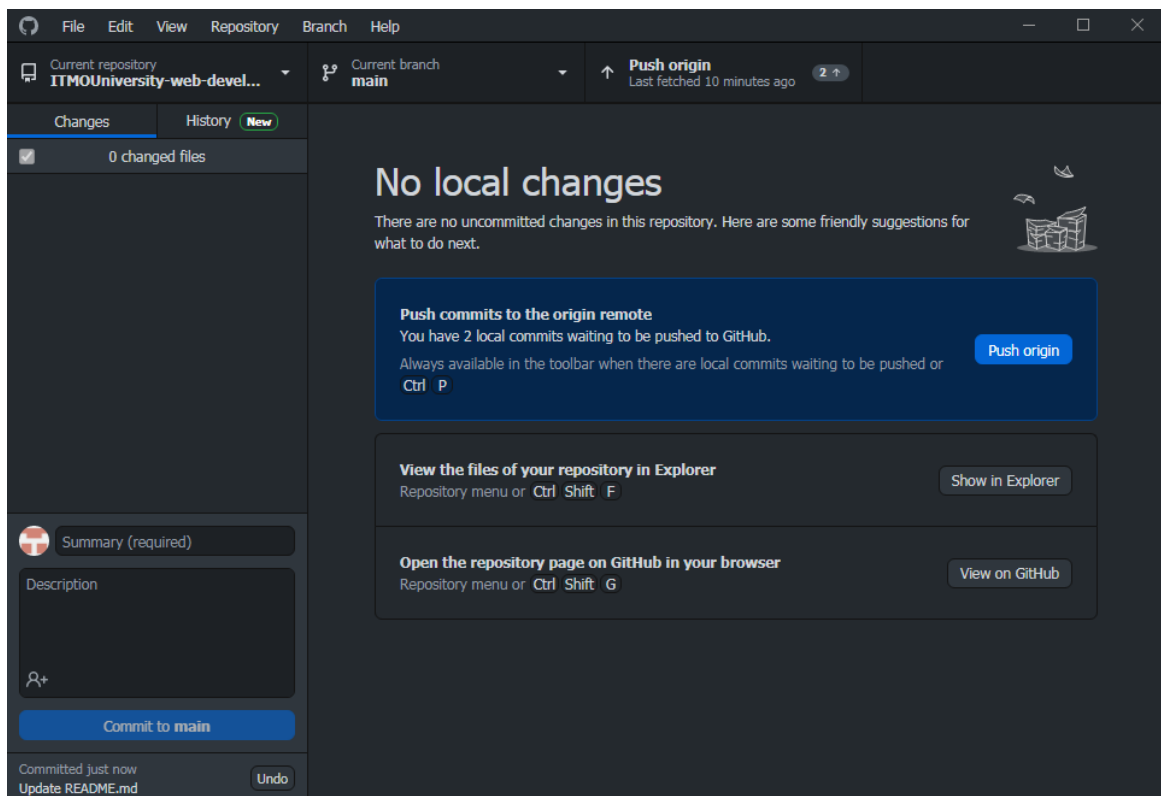


После этого в ветке main (master) локального репозитория создается еще один коммит под именем «Update README.md»:



Отправка изменений на удаленный репозиторий с помощью утилиты GitHub Desktop

Синхронизация с удаленным репозиторием в программе GitHub Desktop производится комбинацией клавиш **Ctrl + P** или нажатием на кнопку «Push Origin»:



Ссылка на созданный публичный репозиторий в GitHub

После отправки изменений на [веб-странице репозитория](#) появится новый файл и подробное описание:

The screenshot shows a GitHub repository page for 'InventiveSpark / ITMOUniversity-web-developer-Module_5-independent-work_4'. The repository is public and has 0 stars and 0 forks. The main branch is 'main'. The repository contains a README.md file, a LICENSE file, and a hello.txt file. The README.md file is the selected file and contains the following text:

ITMOUniversity-web-developer-Module_5-independent-work_4

Самостоятельная работа №4 «Использование Git в других окружениях» к модулю 5 «Средства организации процесса веб-разработки в команде» курса «Веб-разработчик 20.35» от Университета ИТМО.

Пояснительная записка к самостоятельной работе доступна [здесь](#).

P.S. Остальные практические и самостоятельные работы к курсу доступны [здесь](#).

The right sidebar shows the repository's metadata: Readme, MIT License, 0 stars, 1 watching, and 0 forks. There are no releases or packages published.

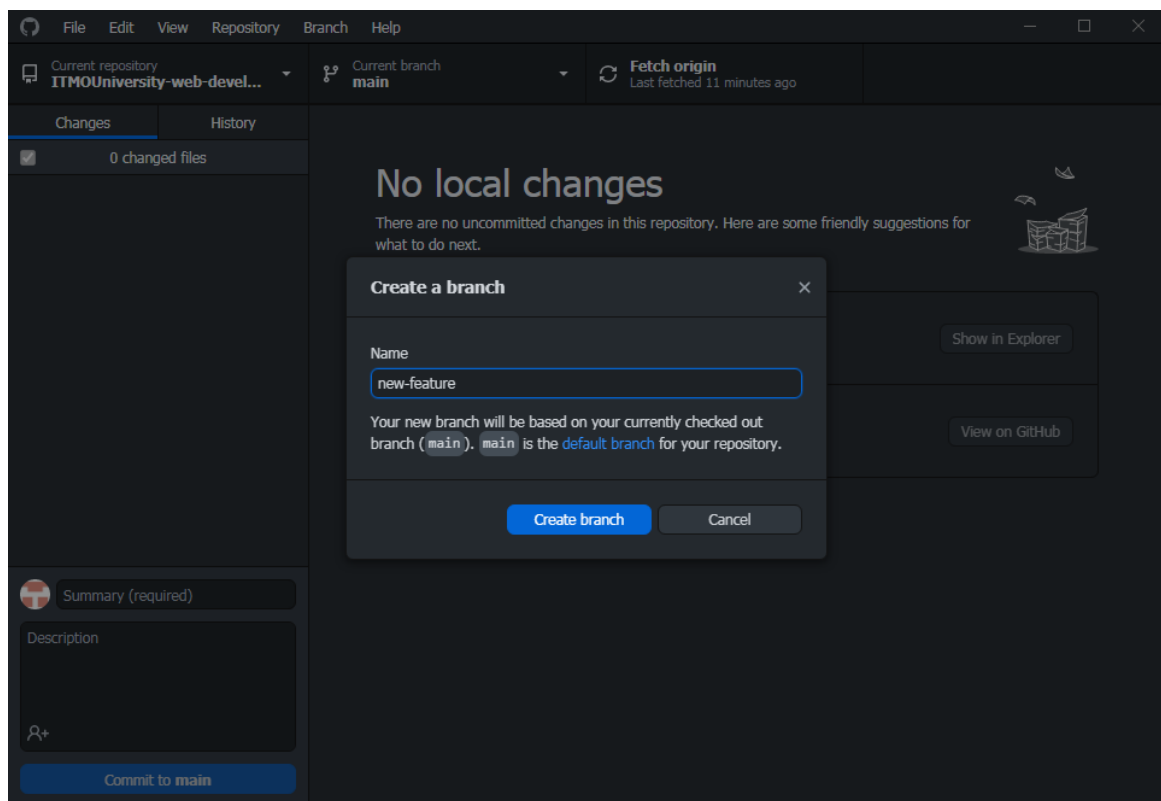
Самостоятельная работа №4.2. Создание новой ветки в GitHub Desktop

Создание новой ветки в репозитории

Новую ветку в репозитории можно создать одним из трех способов:

- Выбрав в главном меню программы «Branch» -> «New Branch».
- Выбрав в верхнем меню пункт «Current branch» и в выпадающем окне щелкнув по кнопке «New branch».
- Нажав комбинацию клавиш Ctrl + Shift + N.

В появившемся окне вводится имя ветки new-feature и нажимается кнопка «Create branch», которая создаст новую ветку с заданным именем:



После создания новой ветки в пункте «Current branch» отображается имя новой ветки. Это означает то, что в нее было осуществлено переключение.

Создание непустого текстового файла с именем new-branch.txt

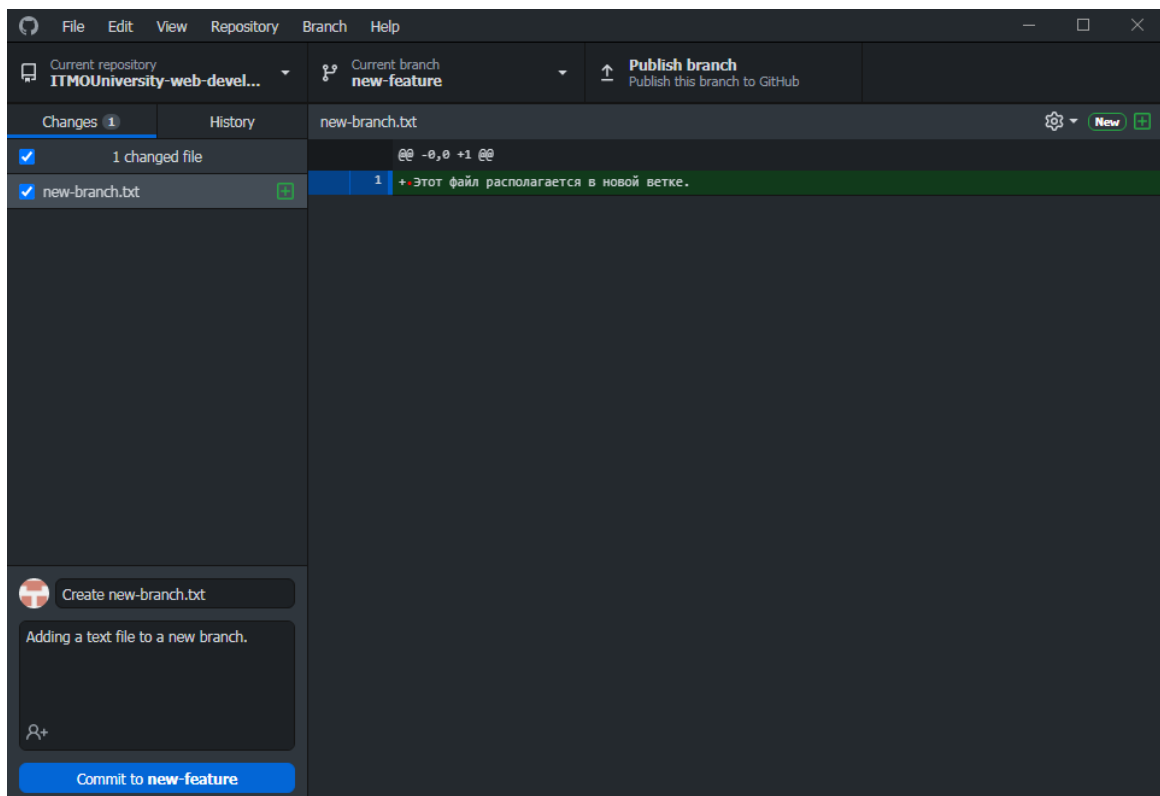
В каталоге репозитория создается файл new-branch.txt со следующим содержимым: «Этот файл располагается в новой ветке».

В левой колонке окна программы Github Desktop будут представлены все файлы проекта, которые были изменены. После нажатия на название файла, в правом поле основного окна программы можно увидеть изменения, сделанные в этом файле.

Для создания коммита под именем «Create new-branch.txt» (фиксация индексированных файлов с комментарием в историю) в форму, расположенную в левом нижнем углу программы, вводится следующая информация:

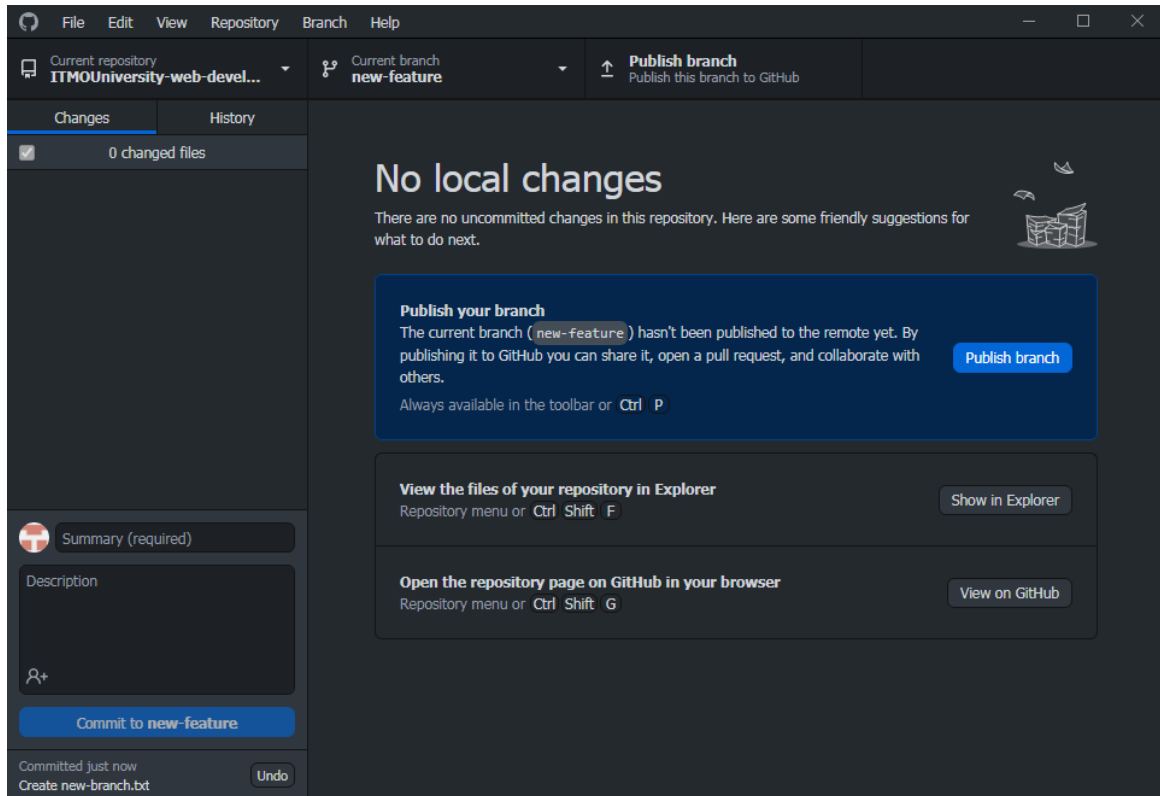
- **Название коммита:** «Create new-branch.txt».
- **Описание изменений, произведенных в коммите:** «Adding a text file to a new branch».

После заполнения всей информации, нажимается кнопка «Commit to new-feature», которая выполнит коммит в ветку new-feature локального репозитория:



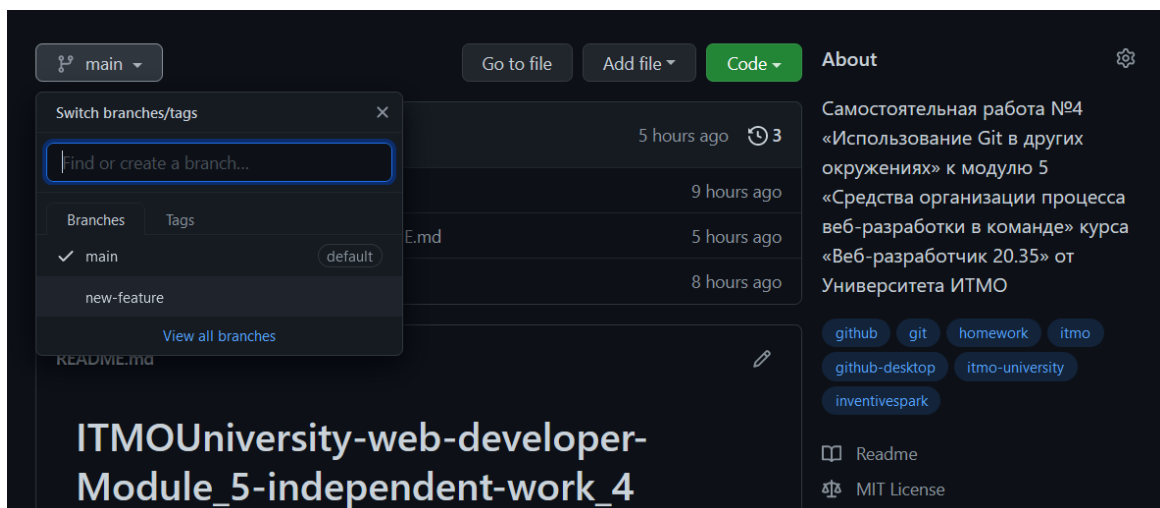
Отправка изменений на удаленный репозиторий с помощью утилиты GitHub Desktop

Синхронизация с удаленным репозиторием в программе GitHub Desktop производится комбинацией клавиш Ctrl + P или нажатием на кнопку «Publish branch»:



Ссылка на публичный репозиторий в GitHub, содержащий созданную ветку

После отправки изменений на [веб-странице репозитория](#) в выпадающем меню появится возможность выбрать новую ветку new-feature:



При переключении на нее появится новый файл new-branch.txt, а также предупреждение, что эта ветка опережает ветку main на 1 коммит. При нажатии на это предупреждение появляется выпадающее окно, в котором предлагается либо сравнить различия между ветками main и new-feature, нажав на кнопку «Compare», либо сразу перейти к созданию запроса на слияние веток (внесение изменений из ветки new-feature в основную ветку), нажав на кнопку «Open pull request»:

