

ГУАП

КАФЕДРА № 32

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

старший преподаватель

должность, уч. степень, звание



подпись, дата

В.Е. Белай

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ

ФОРМИРОВАНИЕ ЛОКАЛЬНОГО И УДАЛЁННОГО РЕПОЗИТОРИЯ РАЗРАБОТОК

по курсу: УПРАВЛЕНИЕ РОБОТАМИ И РОБОТОТЕХНИЧЕСКИМИ СИСТЕМАМИ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №

3021



подпись, дата

В.Д. Тарасов

инициалы, фамилия

Санкт-Петербург 2024

Цель работы

Создать локальный репозиторий. Создать и заполнить удалённый репозиторий.

Ход работы

Для создания репозитория воспользуемся системой контроля версий Git.

Взаимодействовать с Git можно множеством различных версий. При работе под ОС Windows доступна программная оболочка Git BASH, добавляющая функционал командной строки/терминала из UNIX-подобных операционных систем в Windows. Кроме того, можно воспользоваться специальными графическими приложениями, упрощающими работу с Git, такими как: GitKraken, GitHub Desktop и т.д. Воспользуемся в данном случае средой Git Bash и GitKraken.

Для создания локального репозитория необходимо создать директорию, которая будет отведена под репозиторий, а затем инициировать её при помощи команды `git init`. После этого система Git создаст свой собственный скрытый каталог внутри избранной директории, который будет исполнять роль своего рода файловой системы.

На рисунке 1 приведён вид интерфейса эмулятора терминала Git Bash под управлением ОС Windows. Здесь последовательно показан путь перехода к директории, которая используется как репозиторий. Опущено только использование команды `mkdir` для создания директории.

```
PCMINSTOR-PMHARD MINGW64 ~
$ ls
3D Objects/ Downloads/ Links/ OneDrive/ SendTo/ Install.log Выбрано
adaline.ipynb Favorites/ Local Settings\0 Perceptron.py ntuser.dat.LOG1 "Главное меню"
AppData/ IntelGraphicsProfiles/ Music/ Pictures/ Useful_Functions.py ntuser.dat.LOG2
Application Data\ Intel/ NTUSER.DAT PrintSpooler/ project/
Contacts/ LR_2.ipynb NTUSER.DAT{3b39e88-18c4-11ea-ab11-000d3aa4692b}.TM.btf Recent/ _jupyter_
Cookbook/ LR_3.ipynb NTUSER.DAT{3b39e88-18c4-11ea-ab11-000d3aa4692b}.TMCContainer000000000000000000021.regtrans-ms 'Kosennett perceptronishe.ipynb'
Desktop/ LR_4.ipynb NTUSER.DAT{3b39e88-18c4-11ea-ab11-000d3aa4692b}.TMCContainer000000000000000000021.regtrans-ms 'Saved Games/'
Documents/ LR_4.ipynb NetHood/ Searches/ env/ 'Виртуал 21 книга.png'
'Mои документы'

PCMINSTOR-PMHARD MINGW64 ~
$ cd f:/
PCMINSTOR-PMHARD MINGW64 /f
$ ls
SECURITY.BIN/ GUAP/ 'PCB projects/' Scans/ Web-pages/ 'Всякие драйва' Матюшны/ Тексты/
20F4ad4168488073af18eef70784feedc72f4_s2_x1.png Games/ 'QU_20_47-0000-111--2826-17_02_141742.xml' Shave/ character/ 'Все для PSP' Мобильники/ Торренты/
'arduino projects' / ISO/ 'QU_20_47-0000-111--2826-17_02_141742.xml' 'Shineyi - DOLC Pack' desktop.ini Found.000/ 'Все для Sonya' Музыка/ 'Валер ЕГЭ'
'fluphill projects' / Inkscape tutorial/ 'QU_20_47-0000-111--2826-17_02_141742.xml.sig' SmilSeelG/ 'Диски с приложения' 'Мусора телефона' Фиксы-Латчи/
'CNC mill backup' / LibreOffice/ RDPs/ Souper10.jpg mdown16.tmp/ Миксеры/ 'Полный нос прима' Прогн/ 'Чужие файлы'
'CNC Projects' / MSOCache/ Riffs/ SD card dump 18.03.2019/ 'System Volume Information' receipt_SNA7W9XFKF2531ZH.pdf Картоночки/ Работы эпохи напечатана/ IP3/
Coding/ Newy.dokument_v_formatе_RTF.docx TCE/ 'VK arch' Аргументы/ 'Работы эпохи напечатана' ТКУМ/
DEMON/ NTOPib/ Saves/ 'VK arch' Видео/ 'Машина телефон' ТКУМ/

PCMINSTOR-PMHARD MINGW64 /f
$ cd /GUAP
bash: cd: /GUAP: No such file or directory
PCMINSTOR-PMHARD MINGW64 /f
$ cd GUAP
PCMINSTOR-PMHARD MINGW64 /f/GUAP
$ ls
GUAP III/ 'GUAP IV' / drobovik.xls 'ГУАП I куст' / 'ГУАП II куст' / Демон/
PCMINSTOR-PMHARD MINGW64 /f/GUAP
$ cd GUAP IV
bash: cd: too many arguments
PCMINSTOR-PMHARD MINGW64 /f/GUAP
$ cd "GUAP IV"
PCMINSTOR-PMHARD MINGW64 /f/GUAP/GUAP IV
$ ls
3021_Homework/ 'Part I' / 'Part II' /
PCMINSTOR-PMHARD MINGW64 /f/GUAP/GUAP IV
$ cd 3021_Homework/Tarasov_3021_repo
PCMINSTOR-PMHARD MINGW64 /f/GUAP/GUAP IV/3021_Homework/Tarasov_3021_repo (master)
$ ls
LR_1.ipynb LR_2.ipynb LR_3.ipynb LR_4.ipynb 'PR2_URI5(1).docx' 'PR2_URI5(1).docx' README.md '-31_URI5(1).docx' 'RP1 (Тарасов 3021).pdf' 'RP2 (Тарасов 3021).pdf' 'RP3 Тарасов 3021.docx' 'RP3 Тарасов 3021.pdf'
PCMINSTOR-PMHARD MINGW64 /f/GUAP/GUAP IV/3021_Homework/Tarasov_3021_repo (master)
$
```

Рисунок 1 – Интерфейс Git Bash

Здесь используются основные команды эмулятора терминала Bash, такие как cd (Change Directory – смена директории) и ls (LiSt – отобразить список файлов в директории).

После перехода в нужную директорию нужно ввести команду git init для создания локального репозитория. Репозиторий после этого можно наполнять файлами.

Для создания удалённого репозитория воспользуемся сервисом GitHub. После регистрации в сервисе, в веб-интерфейсе можно создать ещё один репозиторий, который впоследствии будет привязан к локальному репозиторию на рабочей машине. Интерфейс сервиса GitHub приведён на рисунке 2.

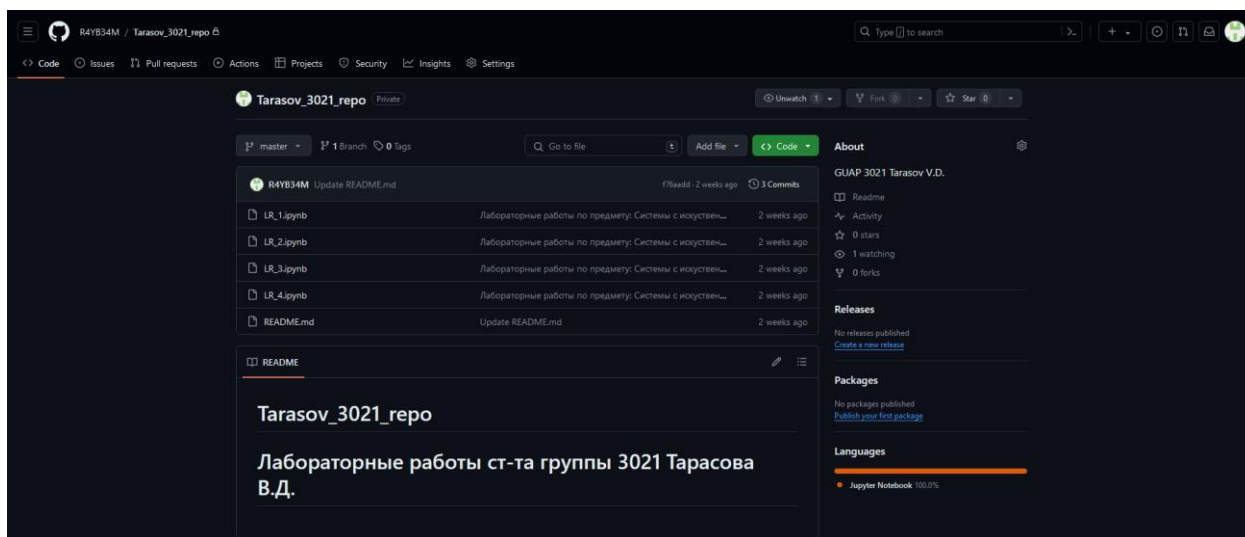


Рисунок 2 – Интерфейс GitHub

После создания репозитория внутри сервиса GitHub его можно привязать к локальному репозиторию, для чего выполняется команда `git remote add origin`. После аргумента `origin` следует ввести адрес репозитория, который был создан внутри сервиса GitHub. Это отображено на рисунке 3.

```
JCD@DESKTOP-FMDMARD MINGW64 /f/GUAP/GUAP IV/3021_Homework/Tarasov_3021_repo (master)
$ git remote add origin https://github.com/R4YB34M/repo-github|
```

Рисунок 3 – Связывание репозитория

На рисунке 4 приведена команда, позволяющая убедиться, что репозиторий доступен для загрузки и скачивания файлов.

```
JCD@DESKTOP-FMDMARD MINGW64 /f/GUAP/GUAP IV/3021_Homework/Tarasov_3021_repo (master)
$ git remote -v
Tarasov_3021_Homework https://github.com/R4YB34M/Tarasov_3021_repo.git (fetch)
Tarasov_3021_Homework https://github.com/R4YB34M/Tarasov_3021_repo.git (push)
```

Рисунок 4 – Подтверждение готовности репозитория к работе

Далее воспользуемся функцией загрузки файлов на удалённый репозиторий при помощи клиента GitHub Desktop. На рисунке 5 приведён вид интерфейса данного клиента.

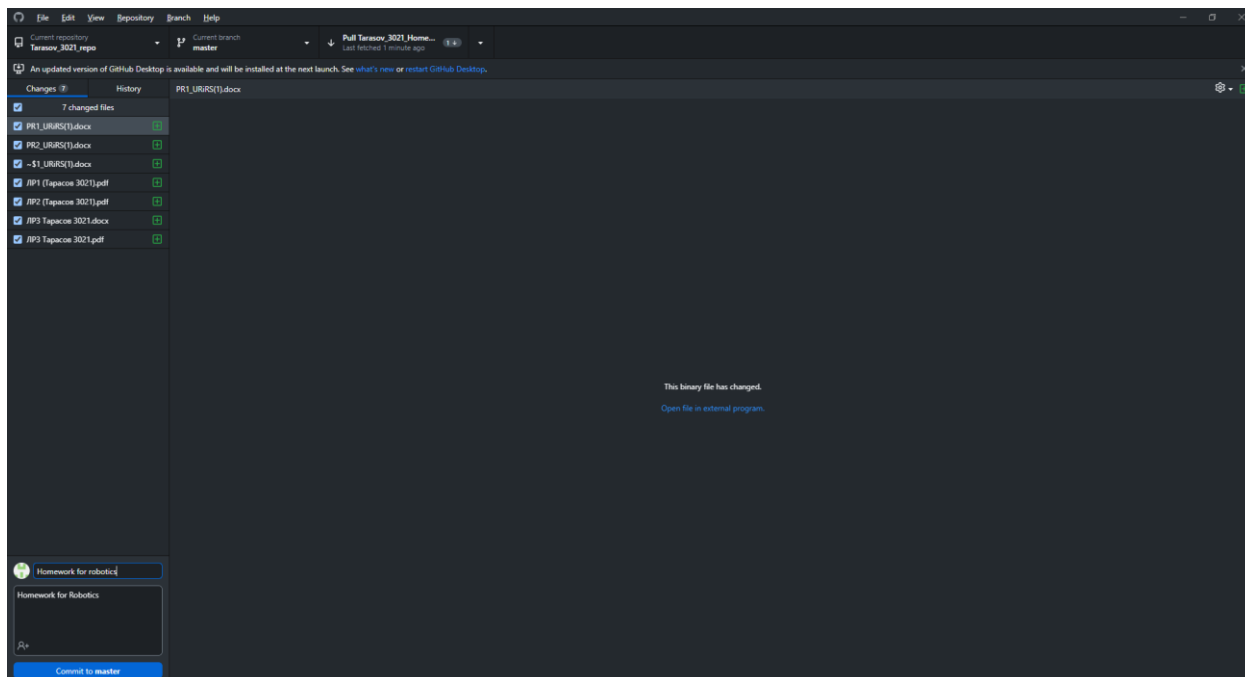


Рисунок 5 – Интерфейс GitHub Desktop

В локальном репозитории присутствуют файлы, которые ещё не были добавлены в удалённый репозиторий. На рисунке 6 показано добавление файлов в удалённый репозиторий через клиент GitHub Desktop.

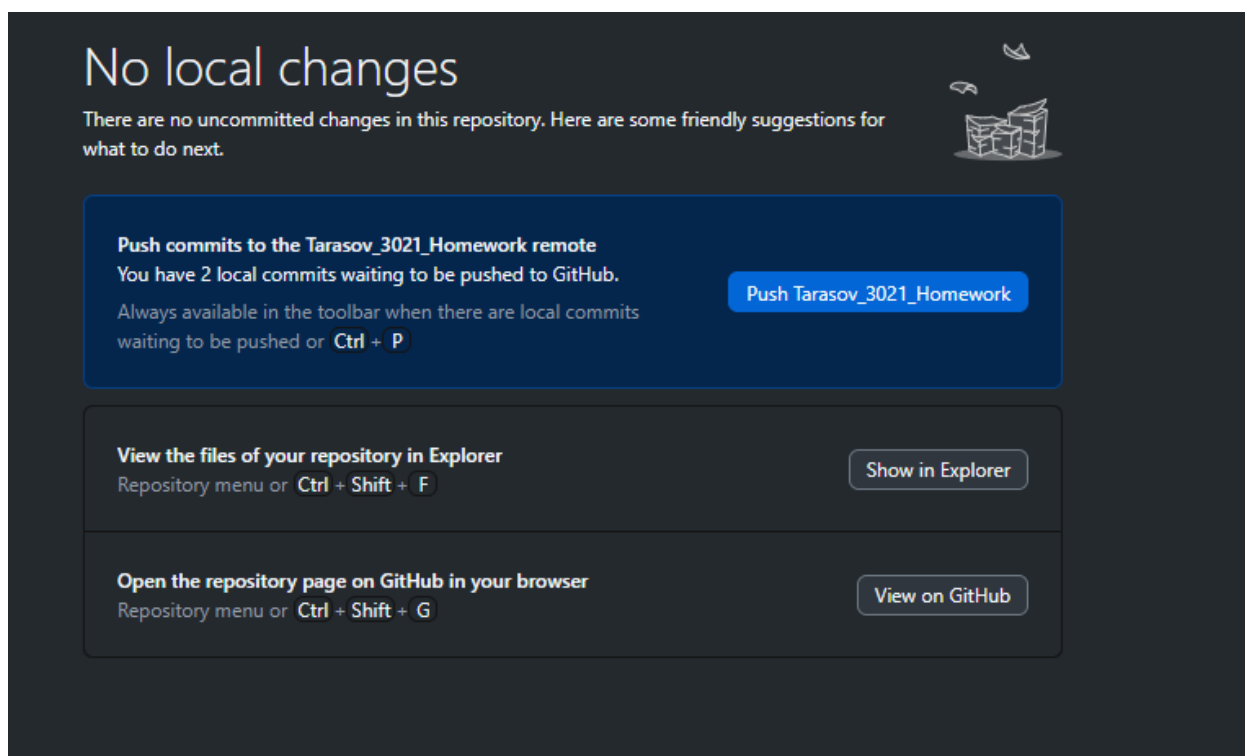


Рисунок 6 – Добавление файлов в удалённый репозиторий

После добавления файлов в графе «история» можно видеть, что в репозитории произошли изменения, то есть были добавлены новые файлы. Это отображено на рисунке 7.

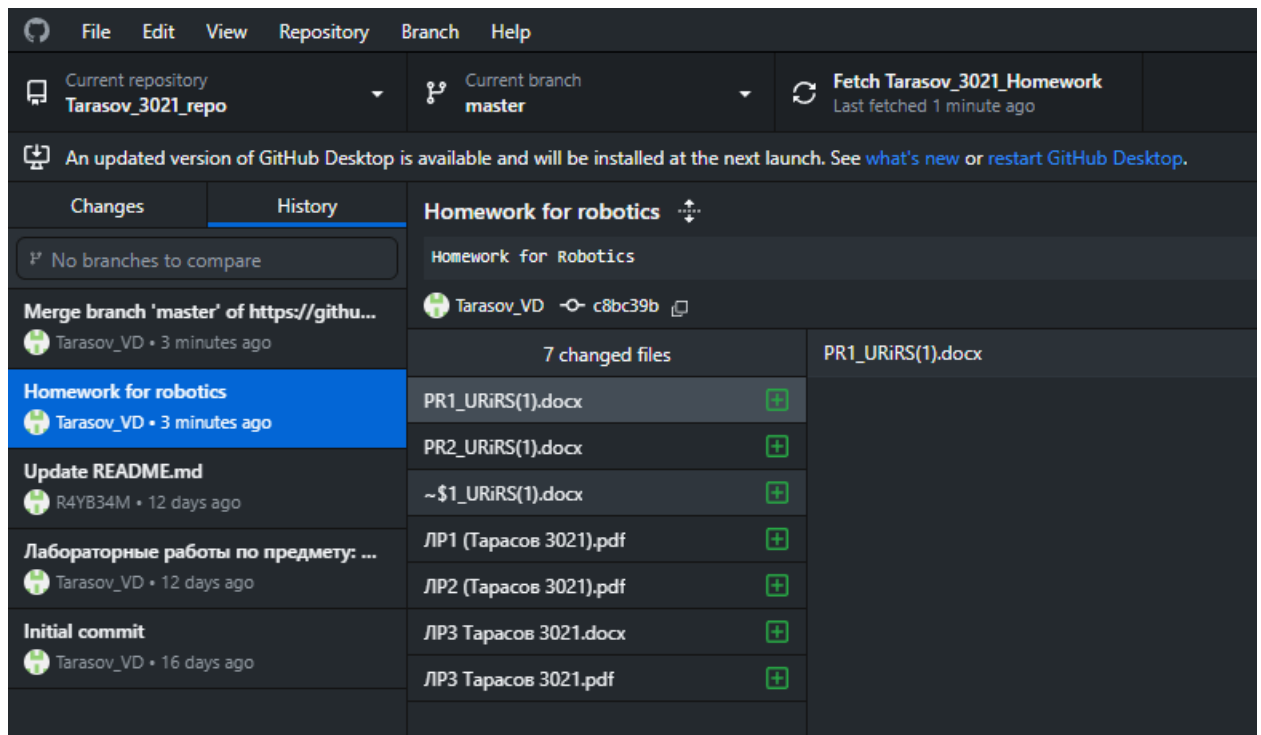


Рисунок 7 – История изменений репозитория

Проверим наличие изменений в веб-интерфейсе GitHub. Откроем страницу репозитория. Видно, что появились новые файлы, что отображено на рисунке 8.

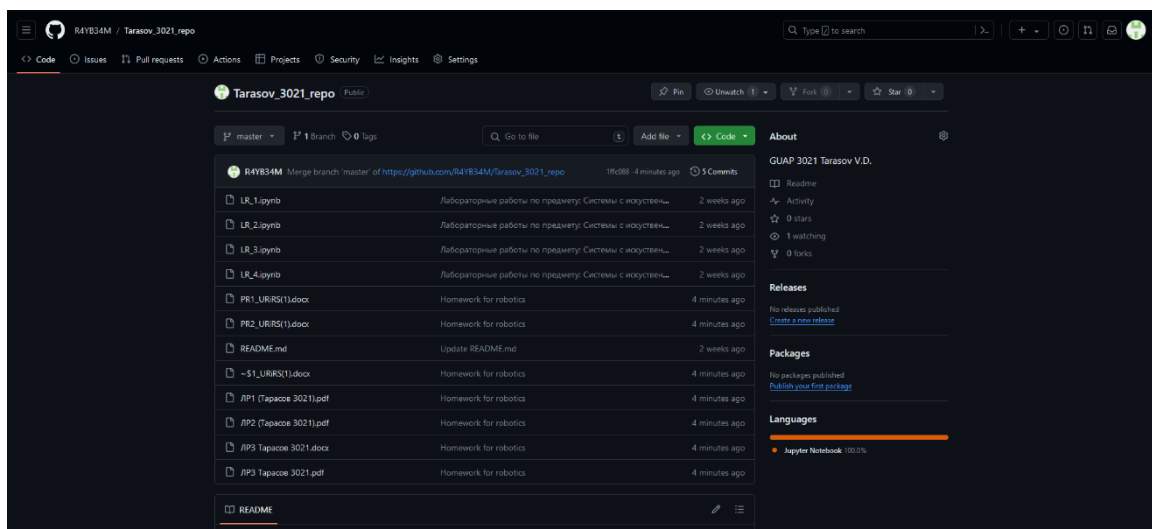


Рисунок 8 – Новые файлы в репозитории

Вывод

. В ходе выполнения работы был создан локальный репозиторий при помощи программного пакета Git Bash.

При помощи графического клиента GitHub Desktop была осуществлена синхронизация локального репозитория с удалённым и добавление новых файлов.

Ссылка на репозиторий: https://github.com/R4YB34M/Tarasov_3021_repo