

Отчет по практической работе №1

Основы Git и Github

Цель работы:

1. изучение наиболее распространенных практик в области контроля версий программного обеспечения, его использования в командной разработке ПО и DevOps;
2. изучение концепции Git, основанной на понятиях репозитория и ветвления версий ПО;
3. изучение порядка использования GitHub и его базовых операций.

Ход работы:

Создаем новый каталог репо и переходим в него. Создаем локальный репозиторий.

```
Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол
$ mkdir repo

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол
$ cd repo

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo
$ git init
Initialized empty Git repository in C:/Users/Alex/OneDrive/Рабочий стол/repo/.git/
```

Рис 1.1

Записываем свои данные для git. Создаем файл и добавляем его в ветку master.

```
Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo (master)
$ git config --global user.email "richard.tsapko@gmail.com"

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo (master)
$ git config --global user.name "roventon"

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo (master)
$ touch goodfile
git
Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo (master)
$ git add goodfile
```

Рис 1.2

Фиксируем изменения. Создаем новую ветку и делаем её активной.

```
Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo (master)
$ git commit -m "added new goodfile"
[master (root-commit) 2bf8aca] added new goodfile
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 goodfile

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo (master)
$ git branch newbranch
git
Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo (master)
$ git checkout newbranch
Switched to branch 'newbranch'
```

Рис 1.3

Создаем файл и добавляем его в новую ветку. Фиксируем изменения.

```
Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/геро (newbranch)
$ touch newbranchfile

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/геро (newbranch)
$ git add newbranchfile

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/геро (newbranch)
$ git commit -m "added newbranchfile"
[newbranch 540468a] added newbranchfile
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 newbranchfile
```

Рис 1.4

Отменяем последний commit. Создаем новый файл и добавляем его в newbranch.

```
Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/геро (newbranch)
$ git revert HEAD --no-edit
[newbranch 3e6f53a] Revert "added newbranchfile"
Date: Sun Feb 16 13:51:37 2025 +0300
1 file changed, 0 insertions(+), 0 deletions(-)
delete mode 100644 newbranchfile

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/геро (newbranch)
$ touch newgoodfile

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/геро (newbranch)
$ git add newgoodfile
```

Рис 1.5

Фиксируем изменения. Делаем активной ветку master и объединяем изменения в новой ветке с основной.

```
Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/геро (newbranch)
$ git commit -m "added newgoodfile"
[newbranch 13dd5c3] added newgoodfile
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 newgoodfile

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/геро (newbranch)
$ git checkout master
Switched to branch 'master'

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/геро (master)
$ git merge newbranch
Updating caae3e9..13dd5c3
Fast-forward
 newgoodfile | 0
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 newgoodfile
```

Рис 1.6

Репозиторий другого студента.

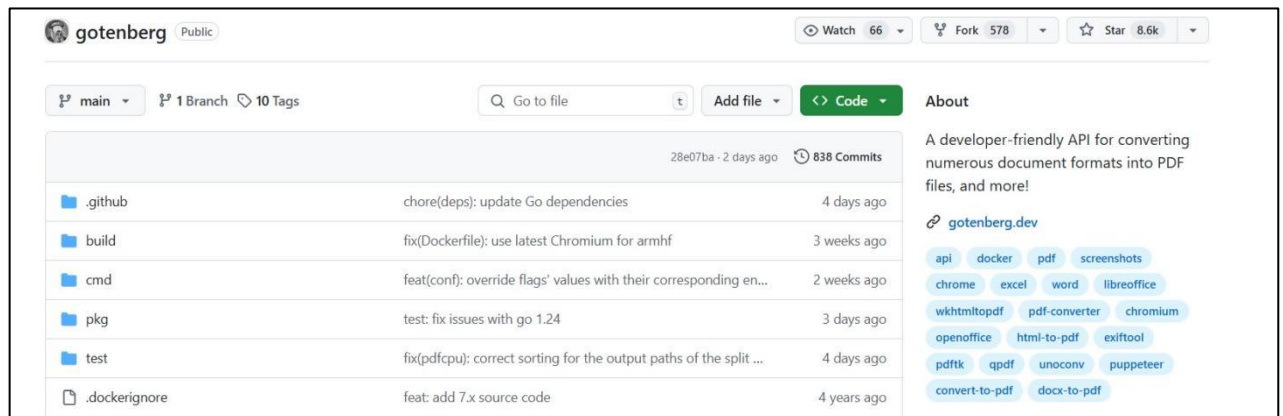


Рис 1.7

Fork репозитория другого студента.

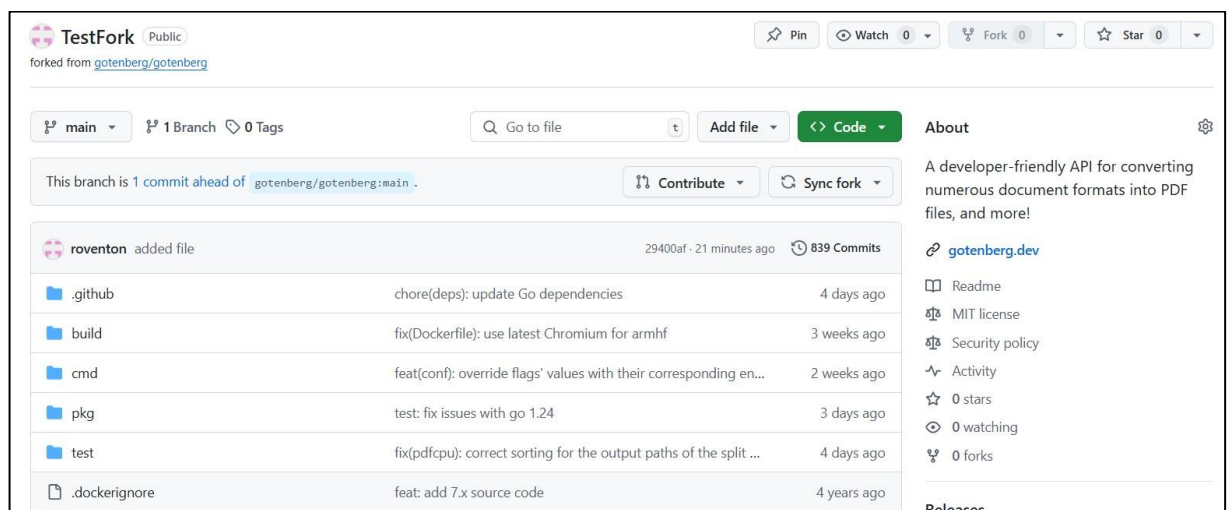


Рис 1.8

Clone проекта в локальный репозиторий.

```
Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/репо (master)
$ git clone https://github.com/roventon/TestFork
Cloning into 'TestFork'...
remote: Enumerating objects: 7793, done.
remote: Counting objects: 100% (215/215), done.
remote: Compressing objects: 100% (127/127), done.
remote: Total 7793 (delta 117), reused 99 (delta 88), pack-reused 7578 (from 2)
Receiving objects: 100% (7793/7793), 20.09 MiB | 2.70 MiB/s, done.
Resolving deltas: 100% (4478/4478), done.
```

Рис 1.9

Переход в папку клонированного проекта, создание в нем нового файла и фиксирование изменений.

```
Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo (master)
$ cd TestFork

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo/TestFork (main)
$ git status
On branch main
Your branch is up to date with 'origin/main'.

nothing to commit, working tree clean

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo/TestFork (main)
$ touch file

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo/TestFork (main)
$ git add file

Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo/TestFork (main)
$ git commit -m "added file"
[main 29400af] added file
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 file
```

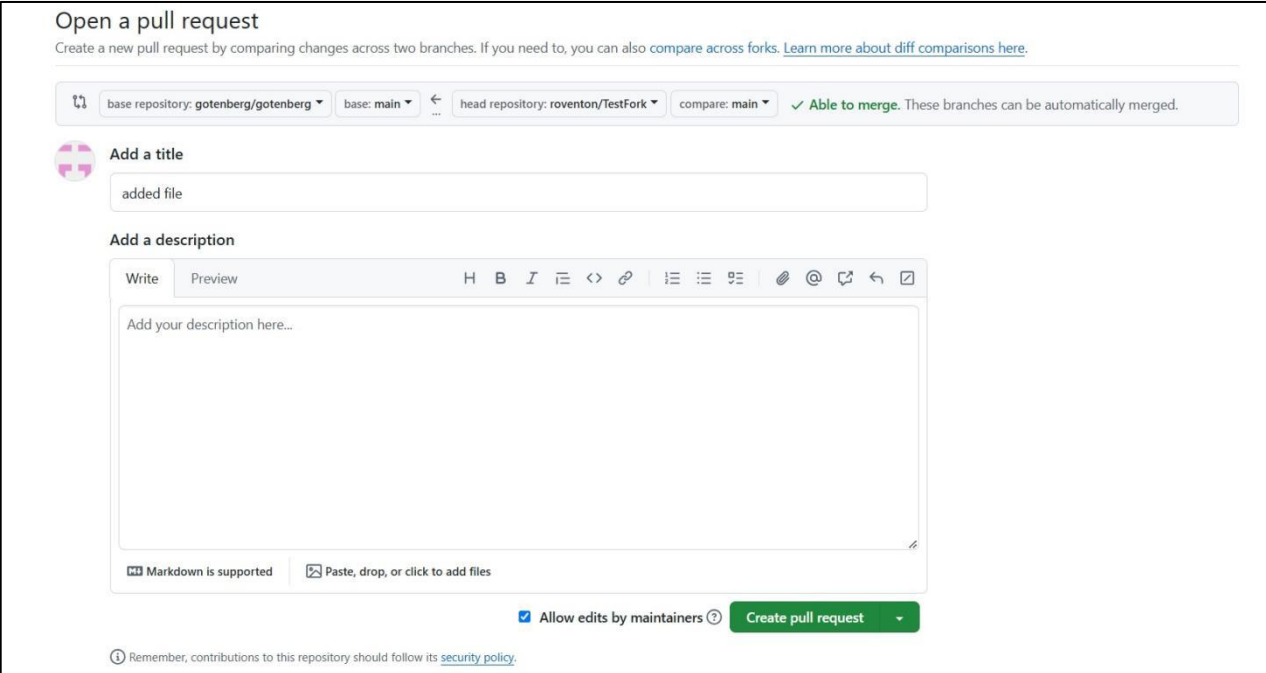
Рис 1.10

Синхронизация с fork-репозиторием.

```
Alex@LAPTOP-5RALGHPH MINGW64 ~/OneDrive/Рабочий стол/repo/TestFork (main)
$ git push origin main
info: please complete authentication in your browser...
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 263 bytes | 11.00 KiB/s, done.
Total 3 (delta 1), reused 1 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/roventon/TestFork
28e07ba..29400af main -> main
```

Рис 1.11

Формирование pull request к origin проекту на прием изменения.



Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#). [Learn more about diff comparisons here](#).

base repository: gutenberg/gutenberg base: main head repository: roventon/TestFork compare: main ✓ Able to merge. These branches can be automatically merged.

Add a title

added file

Add a description

Write Preview H B I @

Add your description here...

Markdown is supported Paste, drop, or click to add files

☒ Allow edits by maintainers **Create pull request**

Remember, contributions to this repository should follow its [security policy](#).

Рис 1.16

Вывод:Было изучено создание локального репозитория, новых веток и файлов, а также их последующее фиксированное изменение. Проведен fork репозитория другого студента, клонирование его проекта, добавление в него файлов и формирование pull request к origin проекту.