Homework assignment 3a

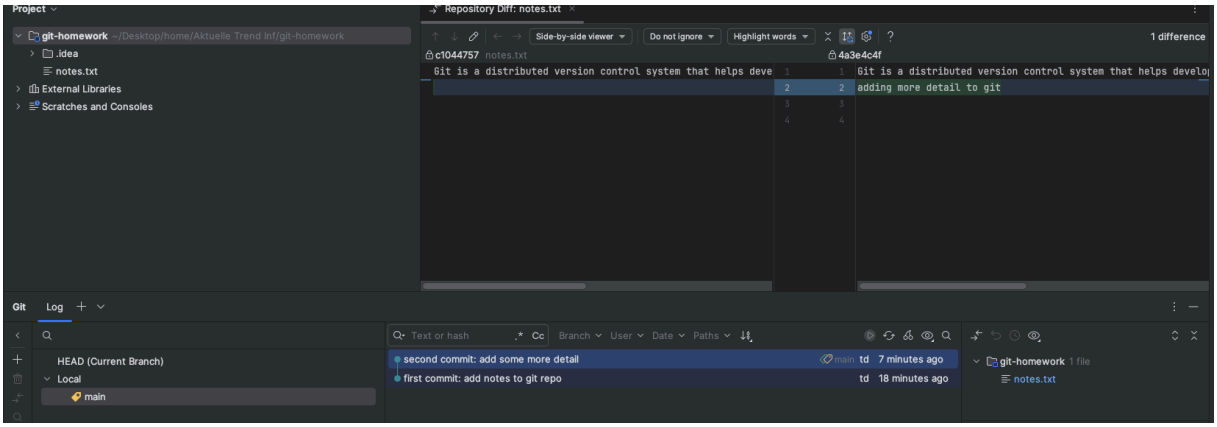
Exercise 1 Practice Git Basics

```
❯ git log
❯ clear
commit 4a3e4c4f532acbcb04e09d6ffadfb24bdb39e46c (HEAD -> main)
Author: td <td@mac.speedport.ip>
Date:   Sat Aug 23 14:37:04 2025 +0200

    second commit: add some more detail

commit c104475752fb63ad0a03499fef86d5c6e59b5251
Author: td <td@mac.speedport.ip>
Date:   Sat Aug 23 14:25:55 2025 +0200

    first commit: add notes to git repo
(END)
```



Exercise 2 Branching and Merging

```
  feature
* main
(END)
```

```
❯ clear
* a44f574 (HEAD -> main, feature) feature: add feature.txt and .gitignore
* 4a3e4c4 second commit: add some more detail
* c104475 first commit: add notes to git repo
(END)
```

```
❯ cat feature.txt
A feature branch is a separate line of development in Git used to work on a specific feature or task
without affecting the main codebase (usually called main or master).
Think of it as a sandbox where you can build, test, and tweak your feature safely.
❯~/Desktop/home/Aktuelle Trend Inf/git-homework main ❯
```

3-Way Merge with conflict:

```
  UW PICO 5.09                                                File: notes.txt

<<<<<<< HEAD
change from main
=======
expermiment change sentence
>>>>>>> experiment
Git is a distributed version control system that helps developers track changes in source code during software development.
adding more detail to git
```

Resolution:

```
change from main
expermiment change sentence
Git is a distributed version control system that helps developers track changes in source code during software development.
adding more detail to git
```

```
*   c148d47 (HEAD -> main) main: resolve merge conflict
|\
| * 1666a45 (experiment) experiment: change notes in experiment
* | ded6954 main: change notes in main
|/
* a44f574 (feature) feature: add feature.txt and .gitignore
* 4a3e4c4 second commit: add some more detail
* c104475 first commit: add notes to git repo
(END)
```

Exercise 3 Reflect on Questions

- Fast-Forward Merge vs. 3-Way Merge
  - A fast-forward merge is like moving forward in a straight line, Git just moves the main branch ahead because there are no changes in between.
  - A 3-way merge happens when both branches have changed, so Git has to look at both versions and their common starting point to combine them.
- What Is Version Control
  Version control helps you keep track of changes in your project, like saving different versions of a file. It's useful when many people work together, so no one overwrites each other's work and you can go back if something breaks(revert commit etc..).
- How Git Stores Data

Git saves project like a series of pictures (snapshots) of all files at different times. Each snapshot has a special code called a SHA hash-like a fingerprint-so Git knows exactly what changed and when.

- What Is Branching in Git
  A branch is like a separate path where developer can work on something new without touching the main project. It's great for:
  - Adding new features
  - Fixing bugs
  - Trying out ideas or experiments