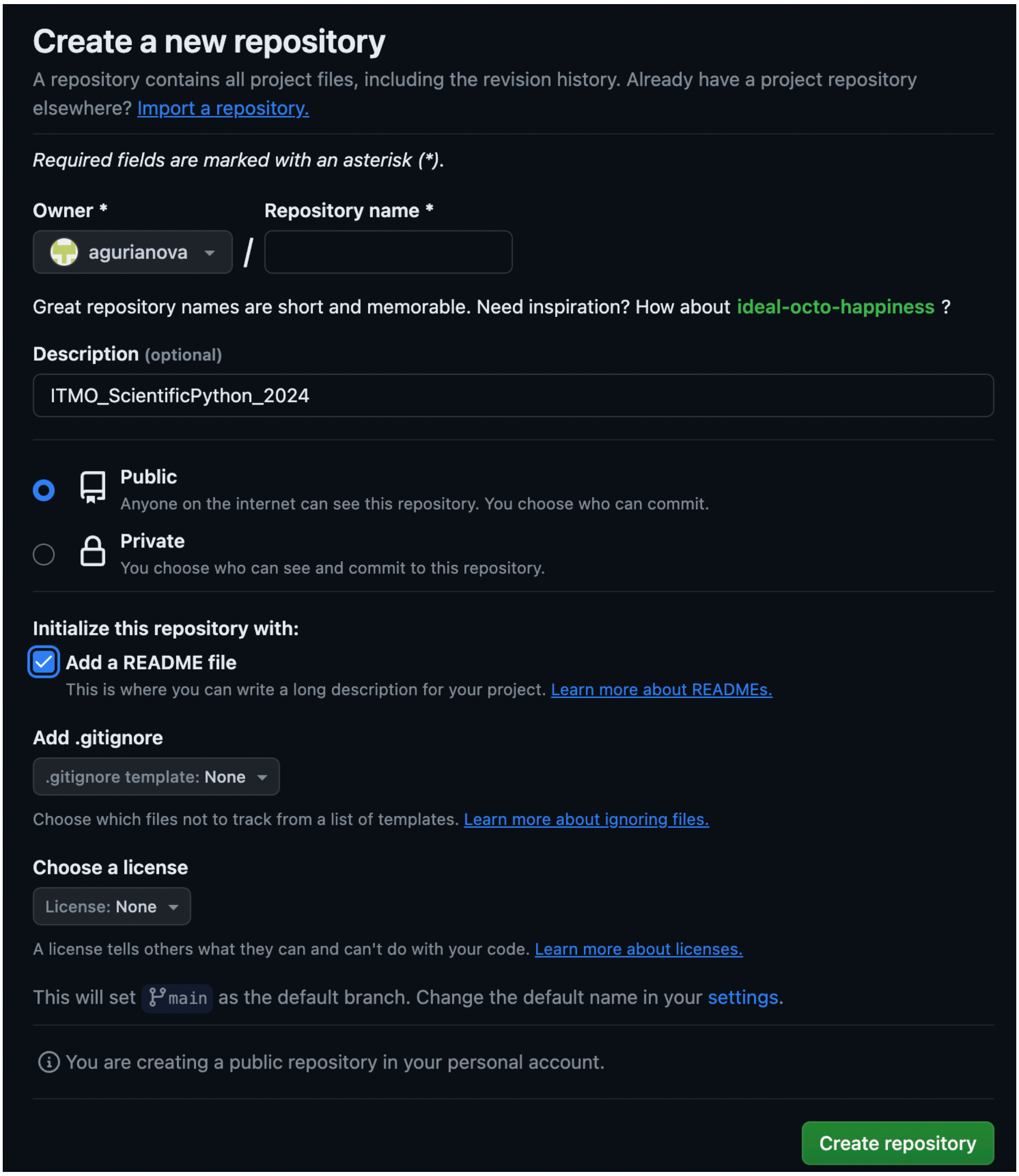
1. Create and clone repo



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

Owner *: agurianova / Repository name *:

Great repository names are short and memorable. Need inspiration? How about ideal-octo-happiness ?

Description (optional): ITMO_ScientificPython_2024

● 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:

☑ Add a README file
This is where you can write a long description for your project. Learn more about READMEs.

Add .gitignore
.gitignore template: None
Choose which files not to track from a list of templates. Learn more about ignoring files.

Choose a license
License: None
A license tells others what they can and can't do with your code. Learn more about licenses.

This will set `main` as the default branch. Change the default name in your settings.

You are creating a public repository in your personal account.

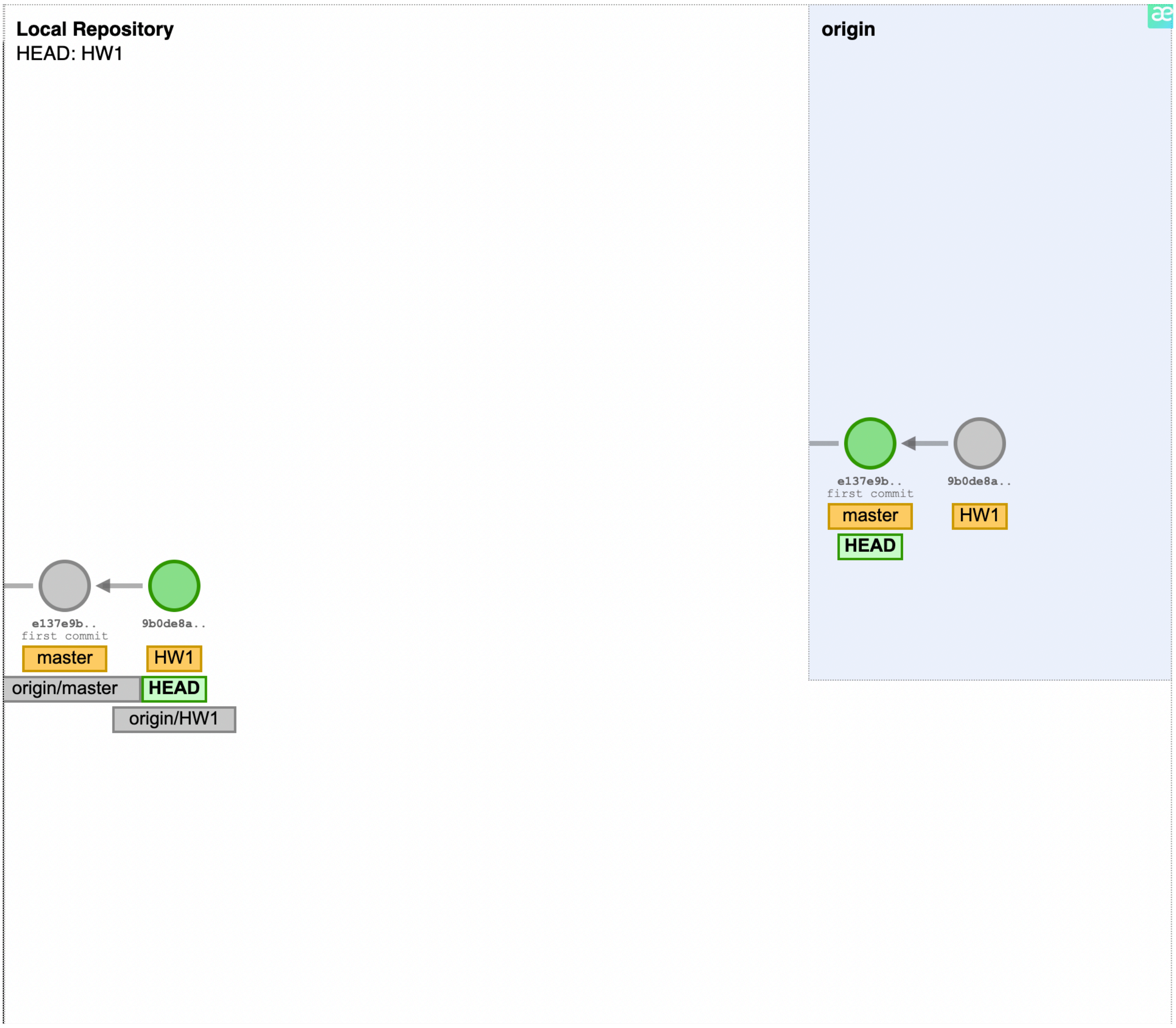
```
git clone https://github.com/agurianova/ITMO_ScientificPython_2024
```

2. Add changes into the HW1 branch

```
git checkout -b HW1
```

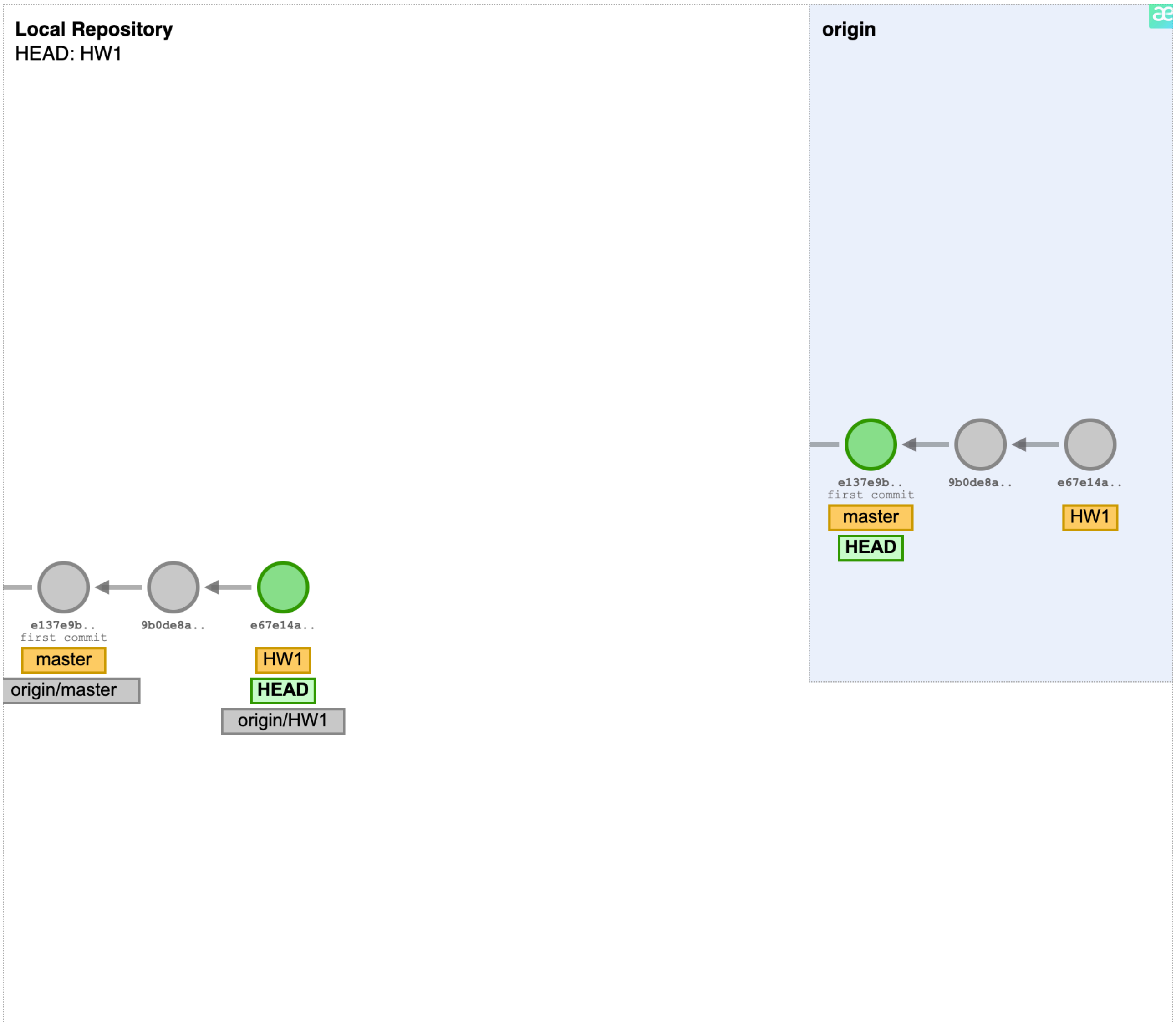
> Put files “hw1.txt”, “test_revert.txt” and “test_revert_merge.txt” into ITMO_ScientificPython_2024 directory on local computer

```
git add .
git status (to check changes)
git commit -m “Three txt files have been added”
git push -u origin HW1
```



> Organised files into HW1 folder

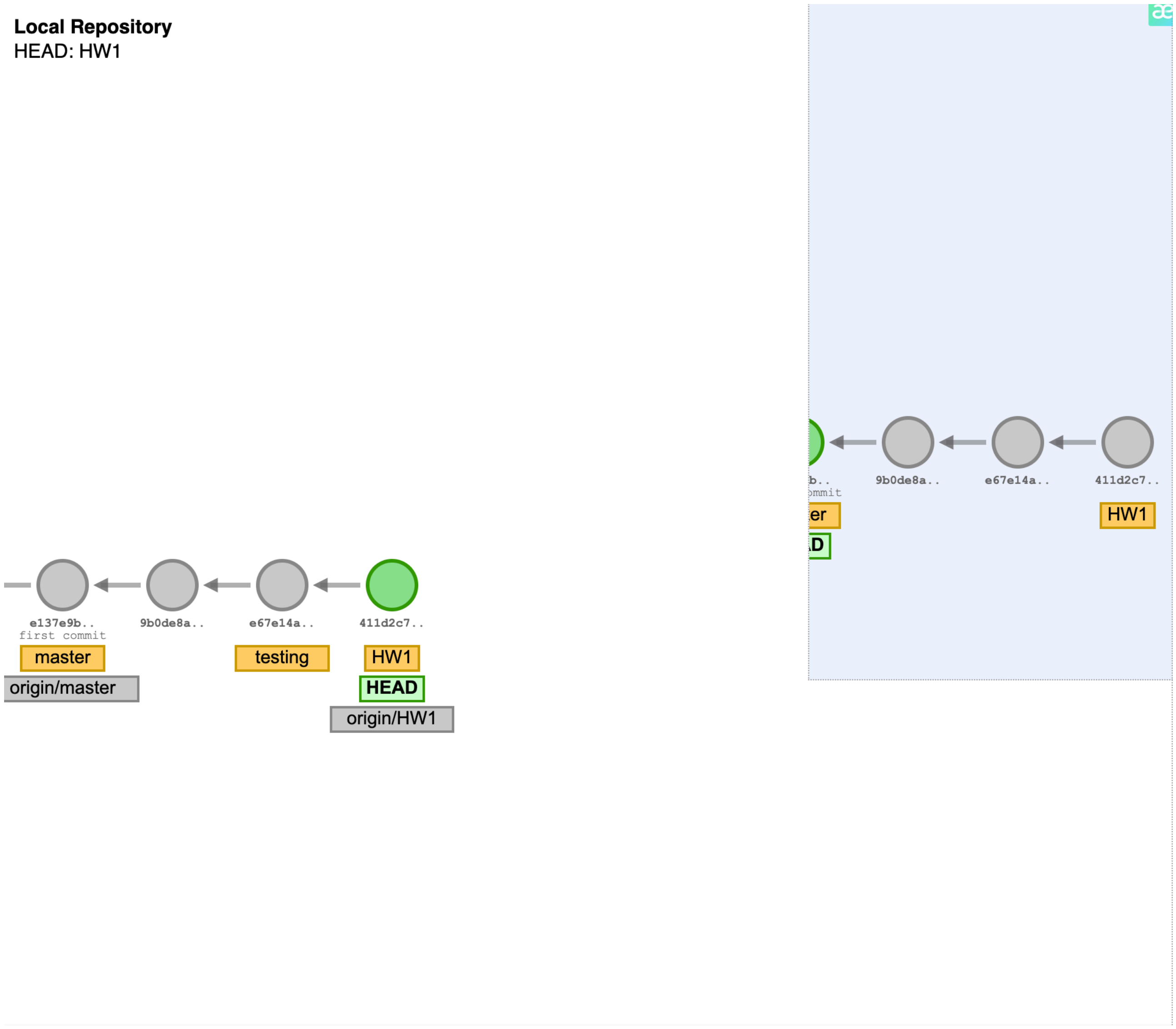
```
git add .
git status
git commit -m “Organized files into a new folder”
git push origin HW1
```



```
git branch testing
```

> Changed hw1.txt

```
git add .
git status
git commit -m “Added changes to the hw1.txt”
git push origin HW1
```



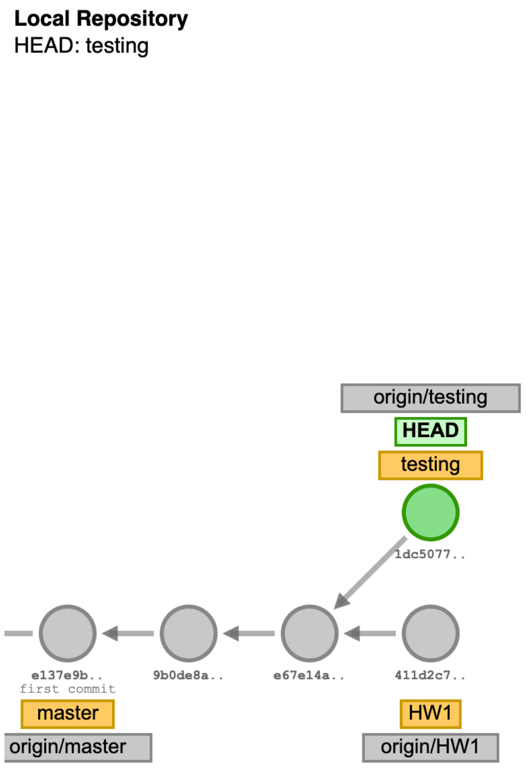
Local Repository
HEAD: HW1

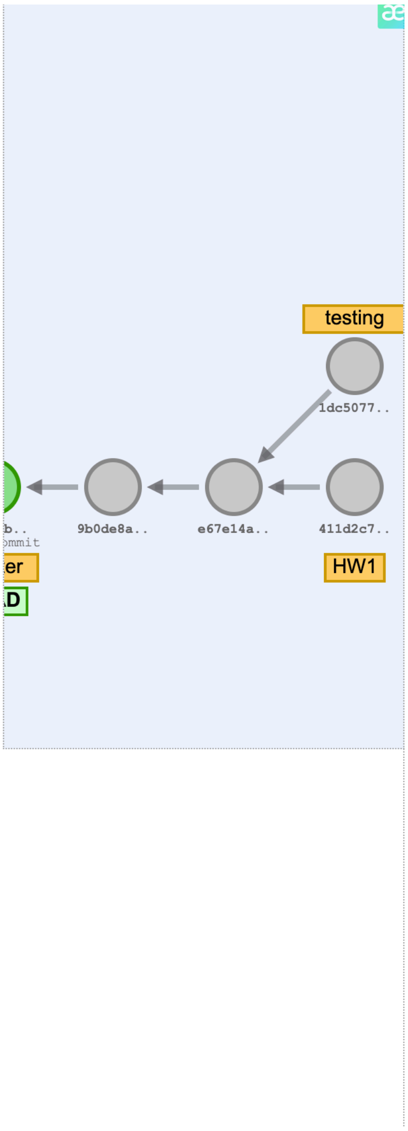
3. Add changes into the testing branch

```
git checkout testing
```

> Changed test_revert.txt

```
git add .
git status
git commit -m “Added changes to the test_revert.txt”
git push -u origin testing
```





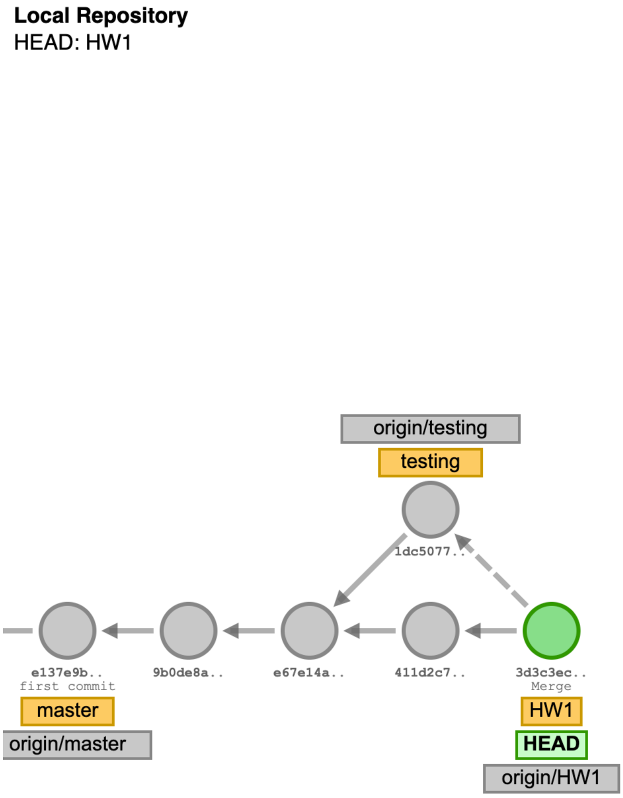
4. Merge branches (testing into the HW1)

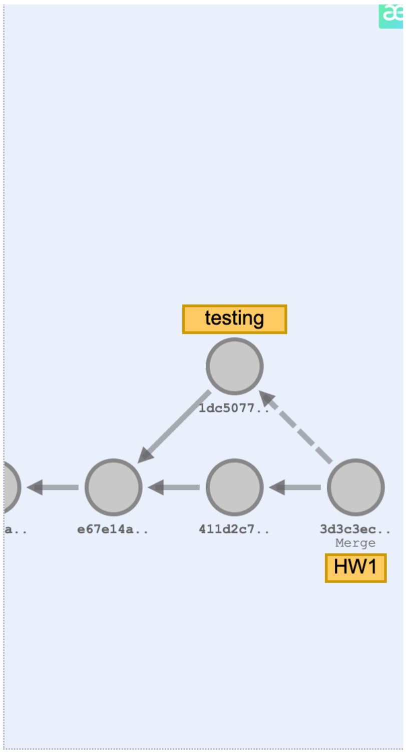
```
git checkout HW1



git merge testing

>> vim
i to insert “Merge branch 'testing' into HW1”
Ctrl+C to escape
:wq to write and quite text editor
```

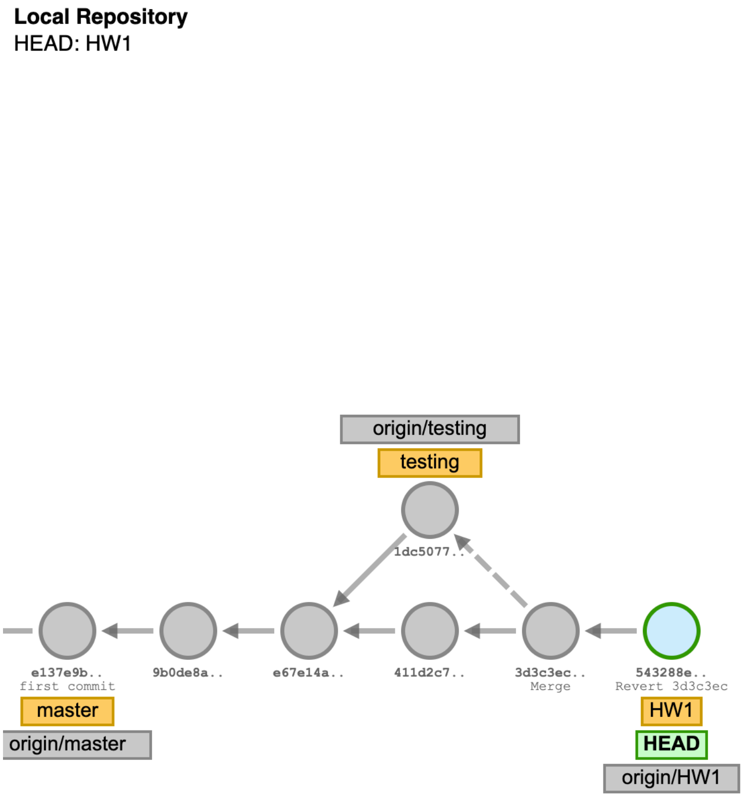


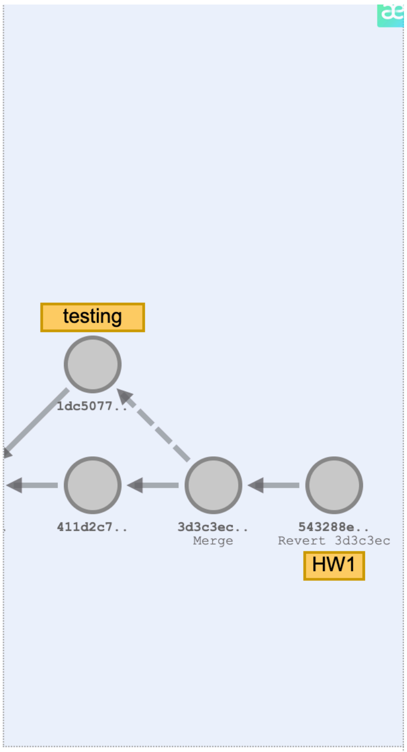


5. Revert merging

```
git revert -m 1 a53054c4b9d7ce4aa2f8d4e704c92297876c765e

>> vim
i to insert “Revert “Merge branch 'testing' into HW1”
Ctrl+C to escape
:wq to write and quite text editor
```



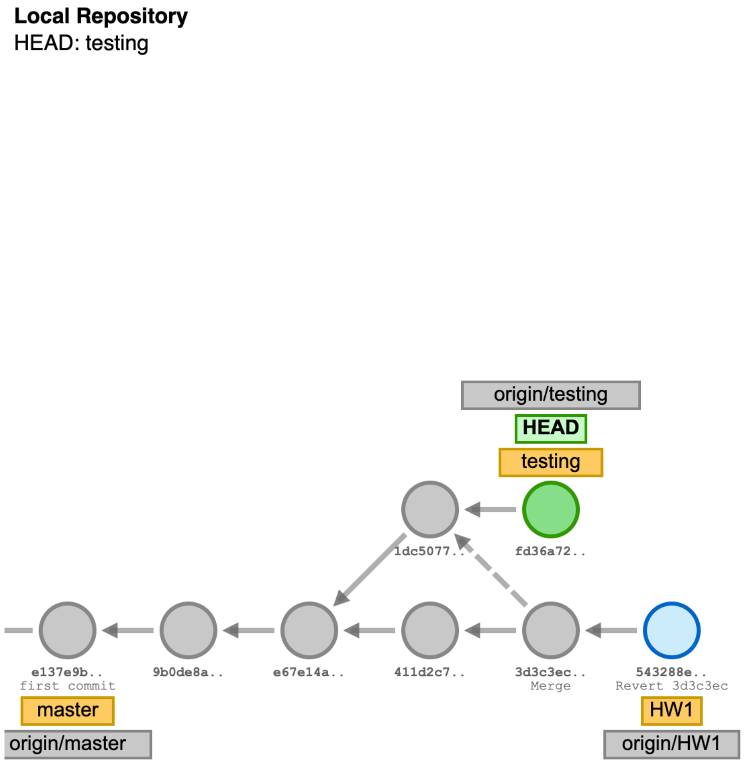


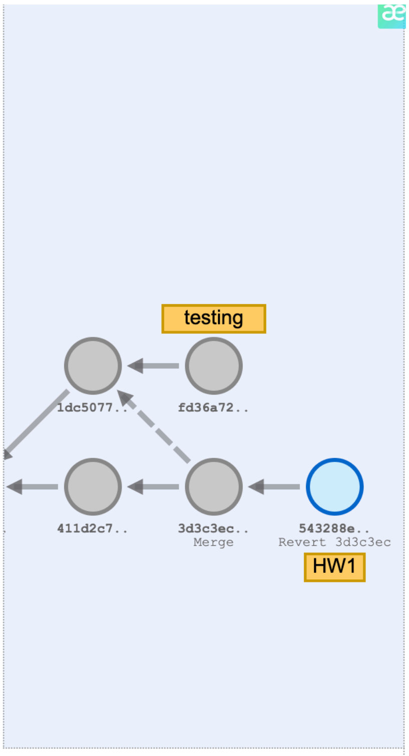
6. Add forgotten changes into the testing branch

```
git checkout testing
```

> Changed test_revert_merge.txt

```
git add .
git status
git commit -m “Added forgotten changes to the test_revert_merge.txt”
git push origin testing
```



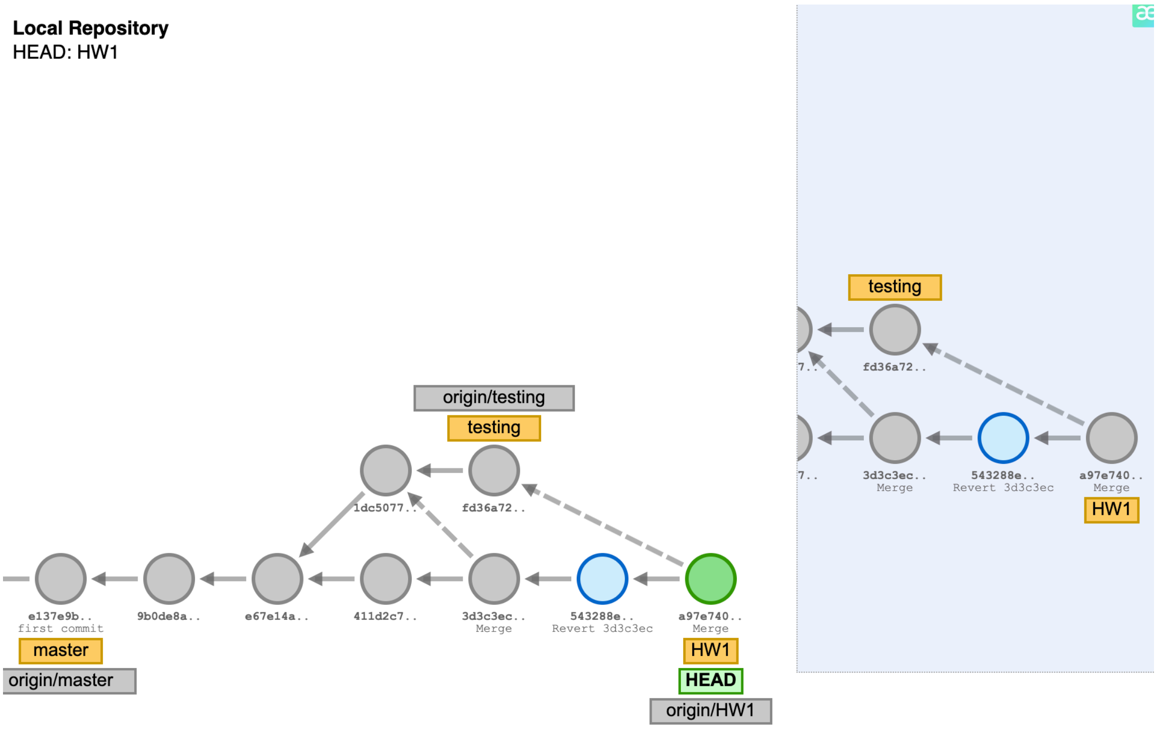


7. Merge branches (testing into the HW1) - last commit only!

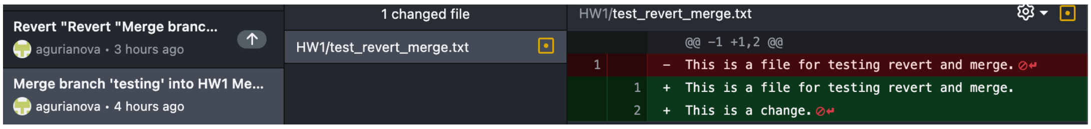
```
git checkout HW1
```

```
git merge testing

>> vim
i to insert and write “Merge branch 'testing' into HW1 x2”
Ctrl+C to escape
```



I’ve expected that both changes in test_revert.txt and test_revert_merge.txt will be added but only the last change in the test_revert_merge.txt file appeared.



Revert "Revert "Merge branc...
agurianova • 3 hours ago

Merge branch 'testing' into HW1 Me...
agurianova • 4 hours ago

1 changed file

HW1/test_revert_merge.txt

HW1/test_revert_merge.txt

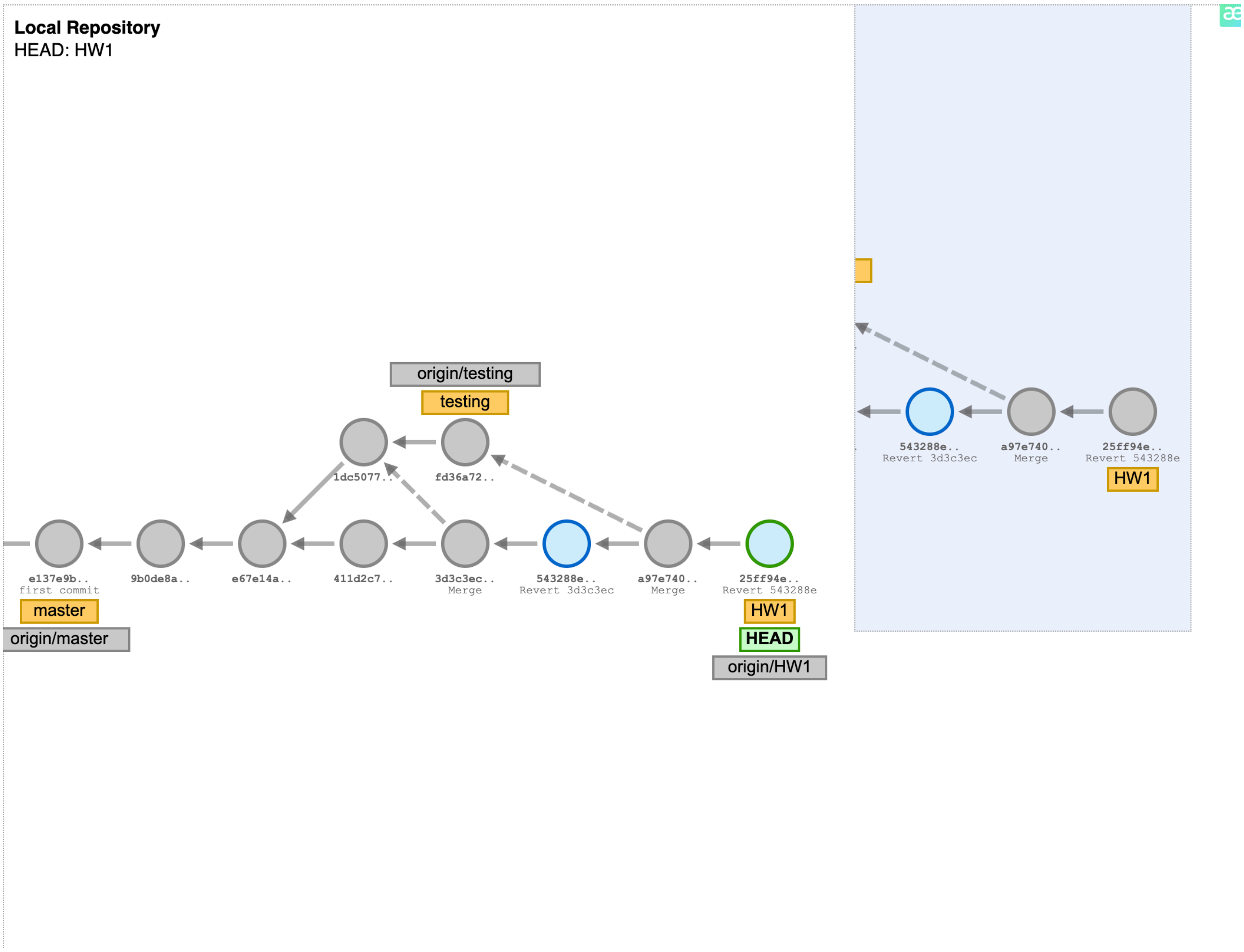
```
@@ -1 +1,2 @@
- This is a file for testing revert and merge.
+ This is a file for testing revert and merge.
+ This is a change.
```

To add changes in test_revert.txt also we should use revert command one more time and re-applies the changes that were initially reverted.

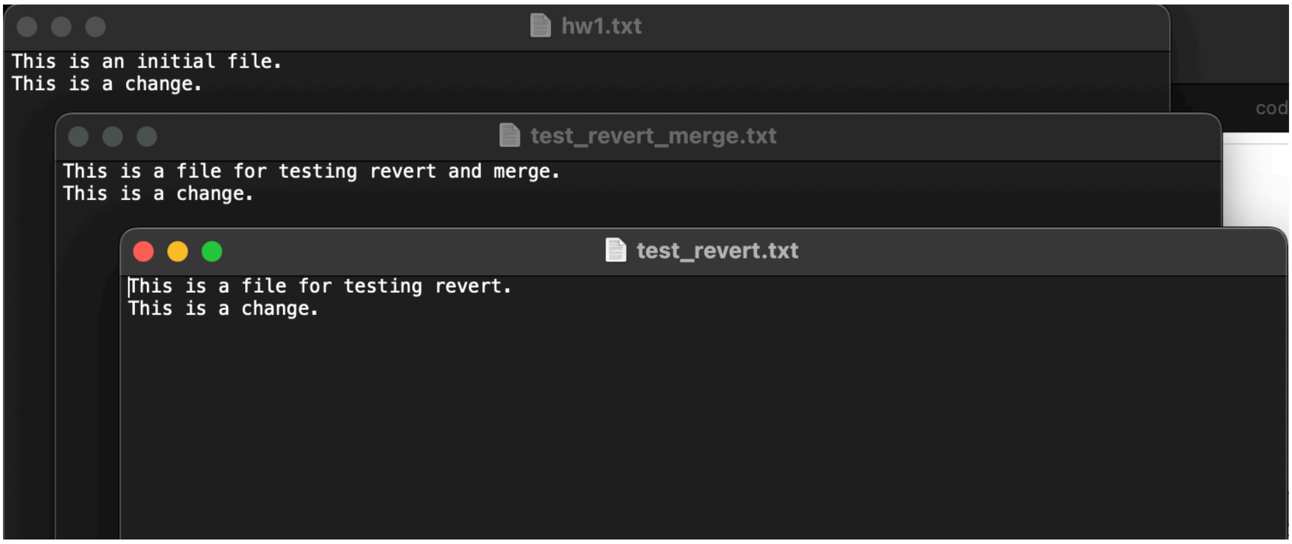
8. Revert revert

```
git revert 22f3334ca3970987b302d189d1424a1d7525a39

>> vim
i to insert “Revert “Revert “Merge branch 'testing' into HW1”””
Ctrl+C to escape
:wq to write and quite text editor
```



Now all files contain changes.



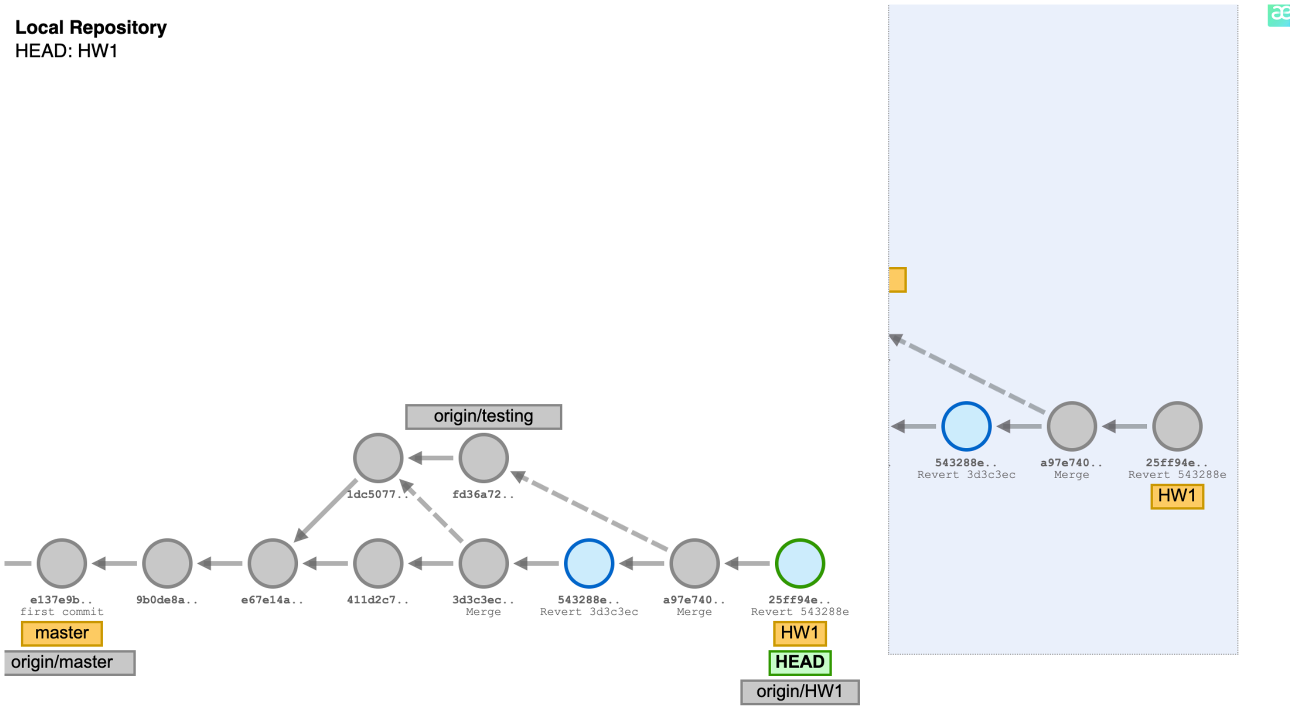
```
This is an initial file.
This is a change.
```

```
This is a file for testing revert and merge.
This is a change.
```

```
This is a file for testing revert.
This is a change.
```

And we can delete testing branch.

```
git branch -d testing
git push origin --delete testing
```



Local Repository
HEAD: HW1
origin/testing
1dc5077..
fd36a72..
e137e9b..
first commit
9b0de8a..
e67e14a..
411d2c7..
3d3c3ec..
Merge
543288e..
Revert 3d3c3ec
a97e740..
Merge
25ff94e..
Revert 543288e
master
origin/master
HW1
HEAD
origin/HW1
543288e..
Revert 3d3c3ec
a97e740..
Merge
25ff94e..
Revert 543288e
HW1