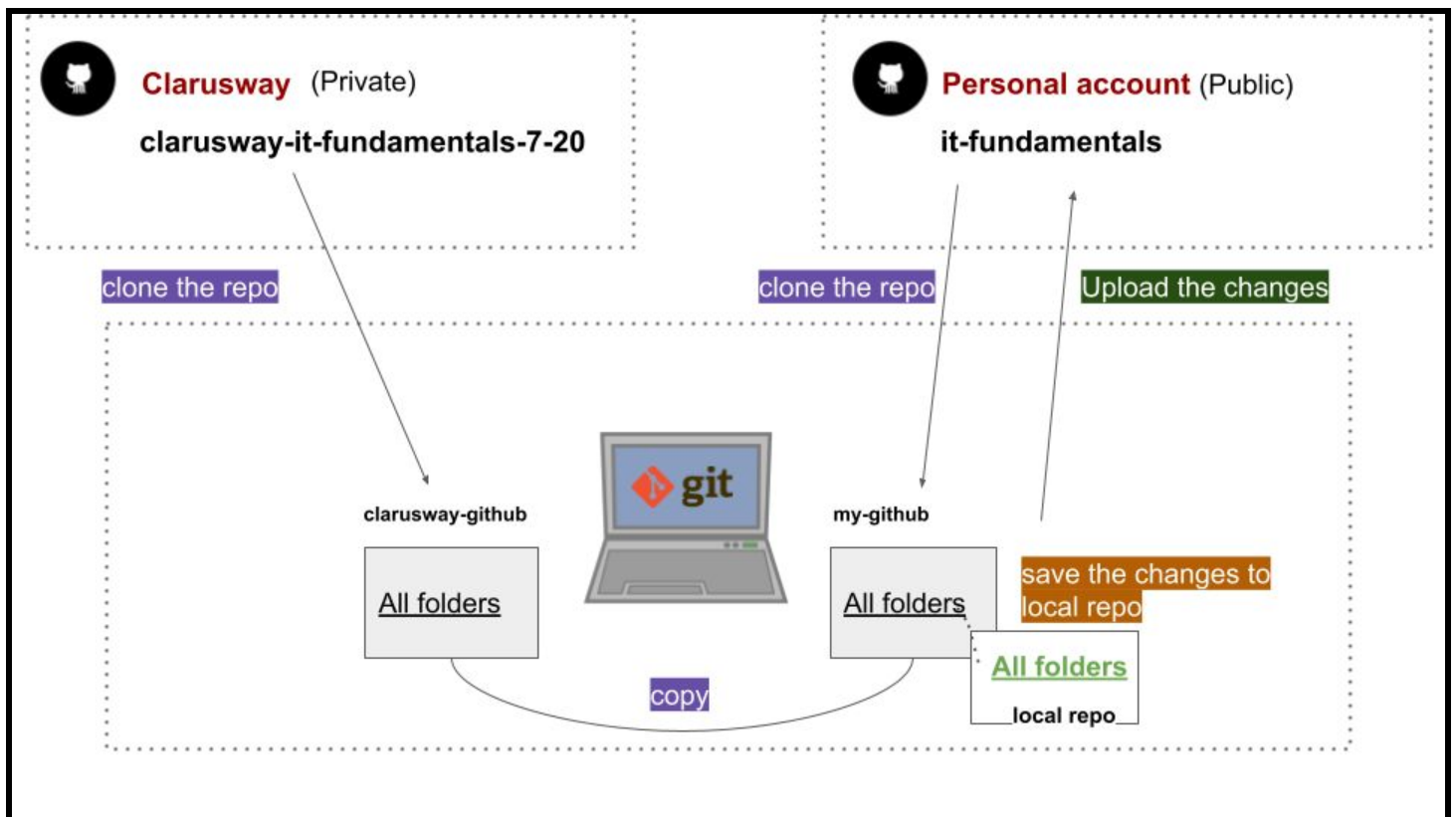


Lab-3

Git



CLARUSWAY©
WAY TO REINVENT YOURSELF



Part-1

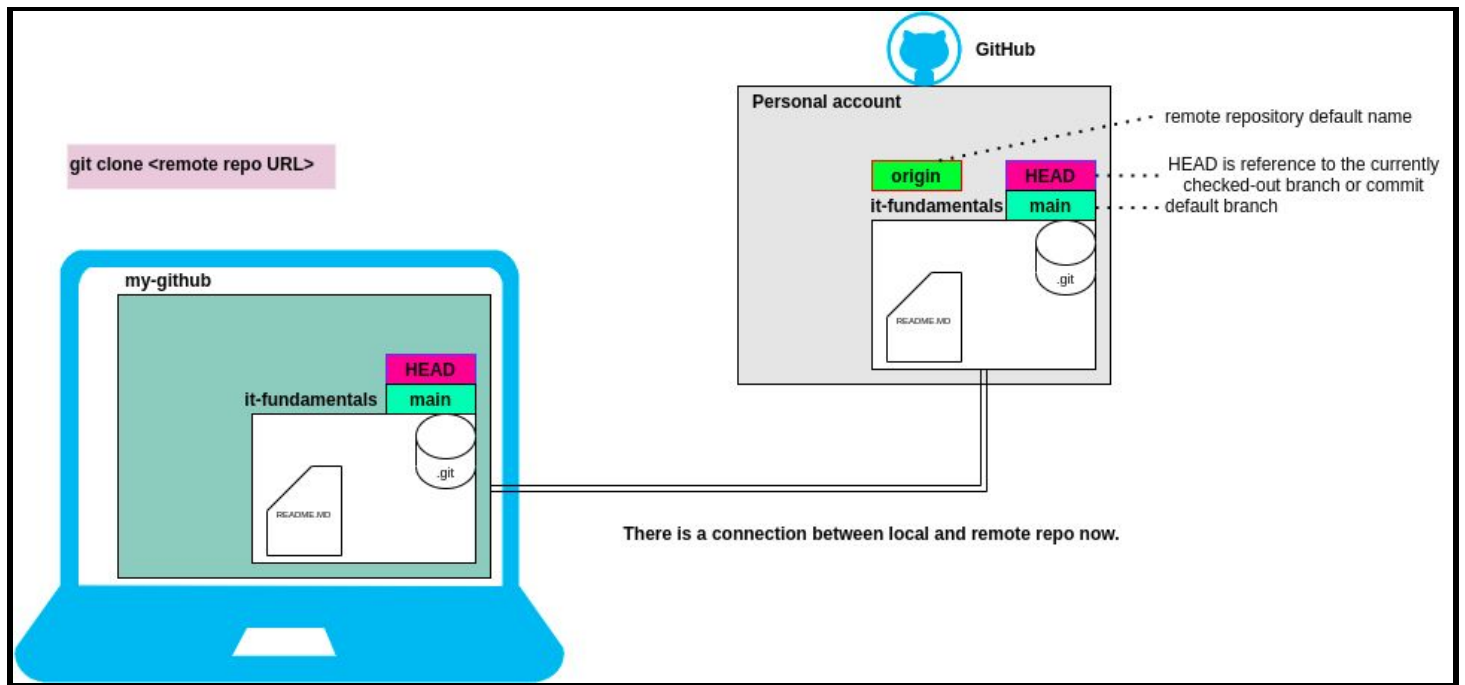
1. Create a repository on your Github account named **it-fundamentals**.
2. Create a folder on your computer named **my-github**.

```
mkdir my-github
```

3. Clone your **it-fundamentals** remote repo in **my-github** folder.

```
cd my-github
```

```
git clone https://github.com/xxxxxxxxx/it-fundamentals
```

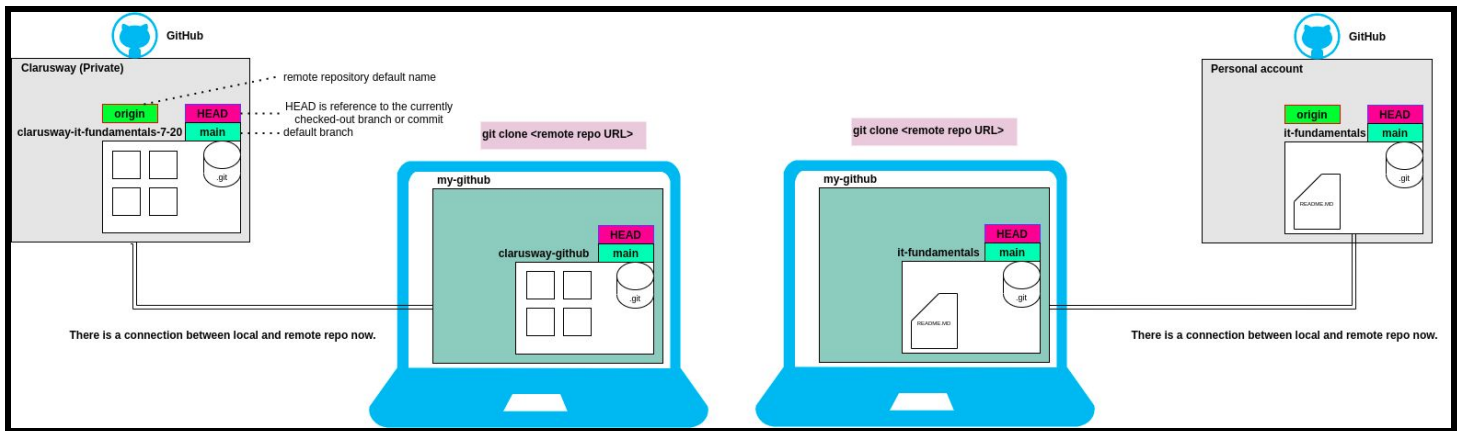


4. Create another folder on your computer named **clarusway-github**.
5. Clone the **clarusway-it-fundamentals-7-20** repo in **clarusway-github** folder.

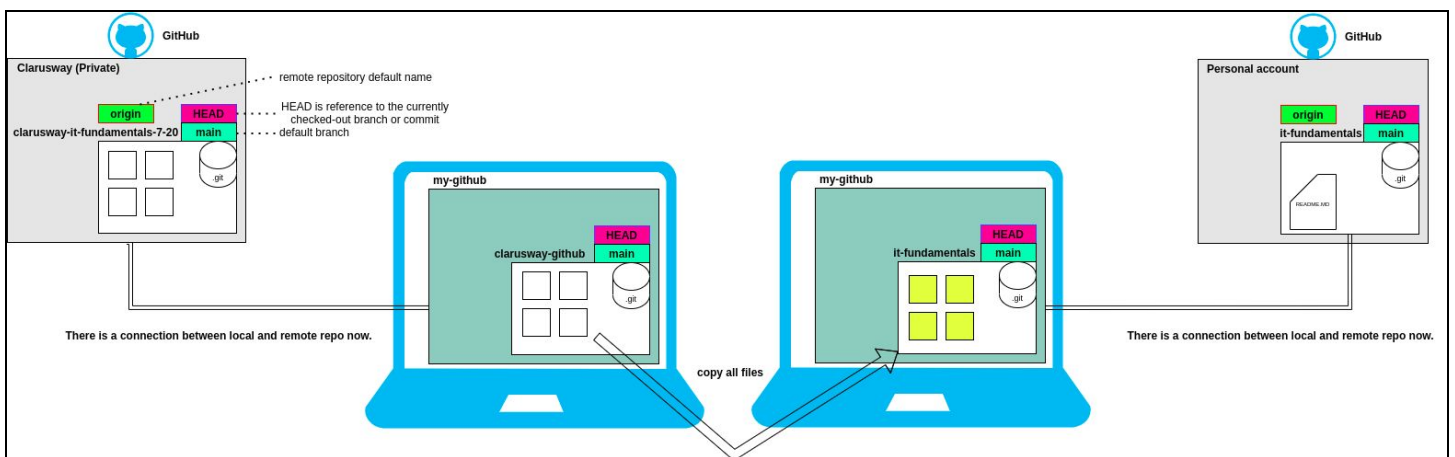
```
mkdir clarusway-github
```

```
cd clarusway-github
```

```
git clone https://github.com/clarusway/clarusway-it-fundamentals-7-20
```

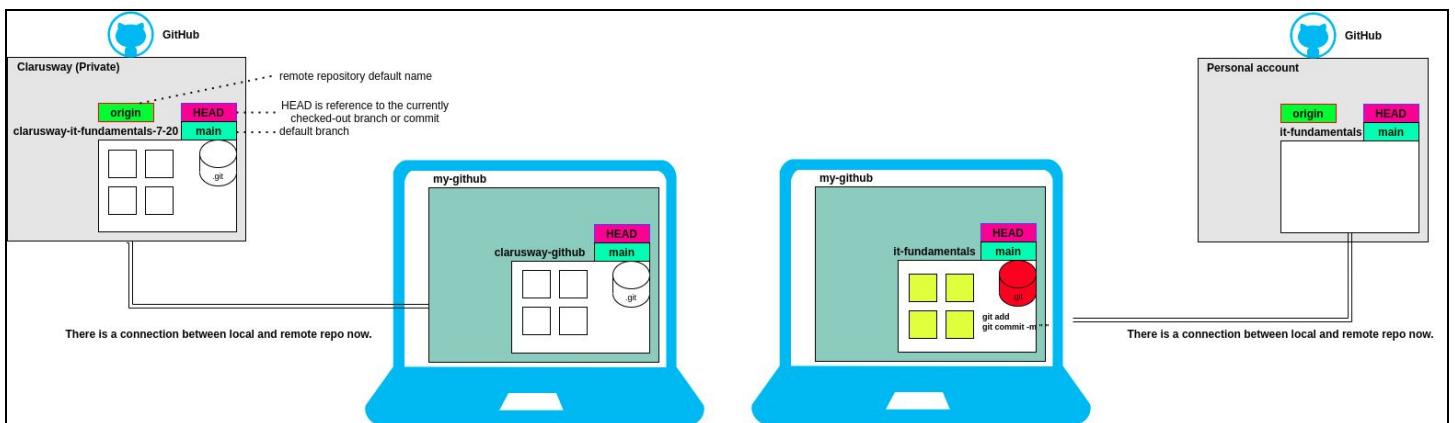


6. Copy all the files and folders in the **clarusway-github** folder to **my-github** folder.



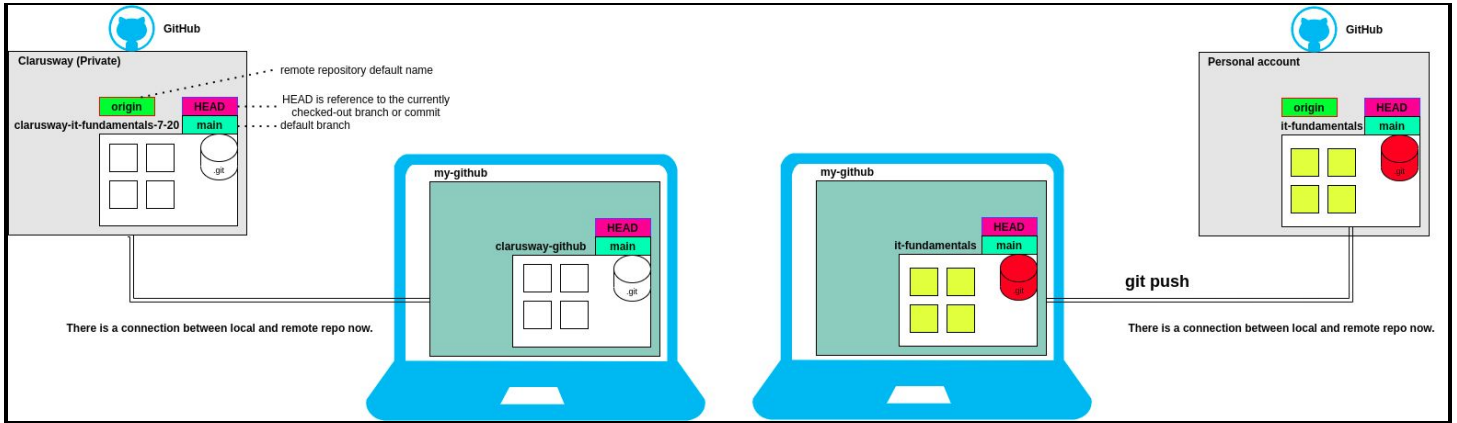
7. Add the changes to your local repo.

```
(cd my-github)
git add .
git commit -m "copied all the files"
```



8. Then send them to your remote repo.

```
git push
```



Part-2

1. Download the changes from **clarusway-it-fundamentals-7-20** (see them in the working directory)

```
git pull
```

2. Copy the **lab1.txt** file from **clarusway-github** folder to **my-github/it-fundamentals/git/lab** folder.
3. Check the status of the project. Add **lab1.txt** file to **index area**. And see the status again.

```
git status  
add lab1.txt  
git status
```

4. Unstaged lab1.txt file. and see the status.

```
git rm --cached lab1.txt  
git status
```

5. Open lab1.txt file with a text editor and write your name in it.

```
vim lab1.txt
```

6. See the status then save the changes to local repo.

```
git status  
add .  
git commit -m "for lab "
```

7. Create a new file named "**lab2.txt**" in the lab folder. And check the status.

8. Save the changes to the local repo. Then check the status of the folder.

```
git commit -am "added new file"  
git status
```

9. See commit history.

```
git log  
git log --oneline
```

10. Switch the first commit. And see the changes at the working directory

```
git checkout <commitID>
```

11. Switch the **main** branch again.

```
git checkout main
```

12. Change the last commit message.

```
git commit --amend -m "message"
```

13. Upload the changes to your remote repository.

```
git push
```

