

Lab XX – Title

Author:

Course Section:

Date:

INTRODUCTION

In this lab, I will be able to successfully set up GitHub on my virtual machine. GitHub is a virtual control system that tracks changes and allows programmers to coordinate their work during software development. I will also be able to perform some basic operations on GitHub to familiarize myself with the system.

PROCESS

Step 0: Signing Up for a GitHub account

Breakpoint 0: This step was very straightforward, but I was asked to verify my student account, so I had to upload an image of my student ID. Other than this, I was able to create an account and locate the settings page for the screenshot.

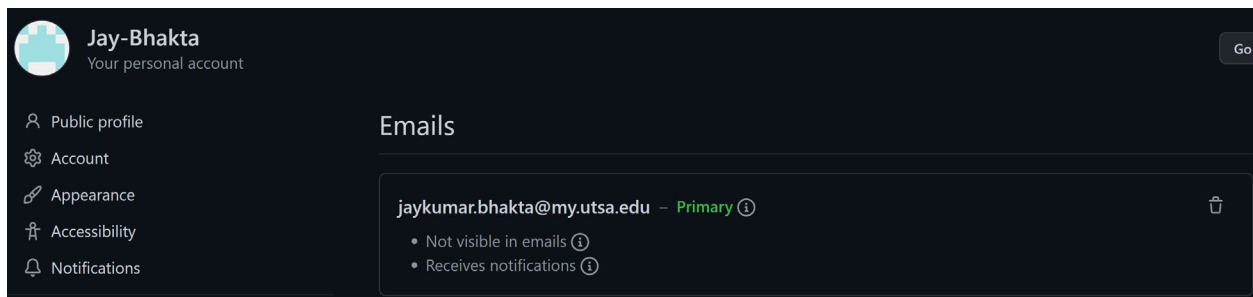


Figure 1: This screenshot depicts my GitHub username and the primary email associated with my account.

Breakpoint 1: For this step, I was able to configure my git username and email on my Linux machine terminal.

```
jay@jay-VirtualBox:~$ git config --global user.name "Jay-Bhakta"
jay@jay-VirtualBox:~$ git config --global user.email "jaykumar.bhakta@my.utsa.edu"
jay@jay-VirtualBox:~$ git config --global --list
user.name=Jay-Bhakta
user.email=jaykumar.bhakta@my.utsa.edu
jay@jay-VirtualBox:~$
```

Figure 2: Configuring my git username and email.

Breakpoint 2: In this step, I created a new repository on the GitHub website. I named the repo “IS1003” and made it private according to the instructions.

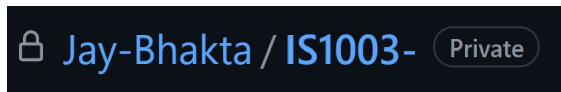
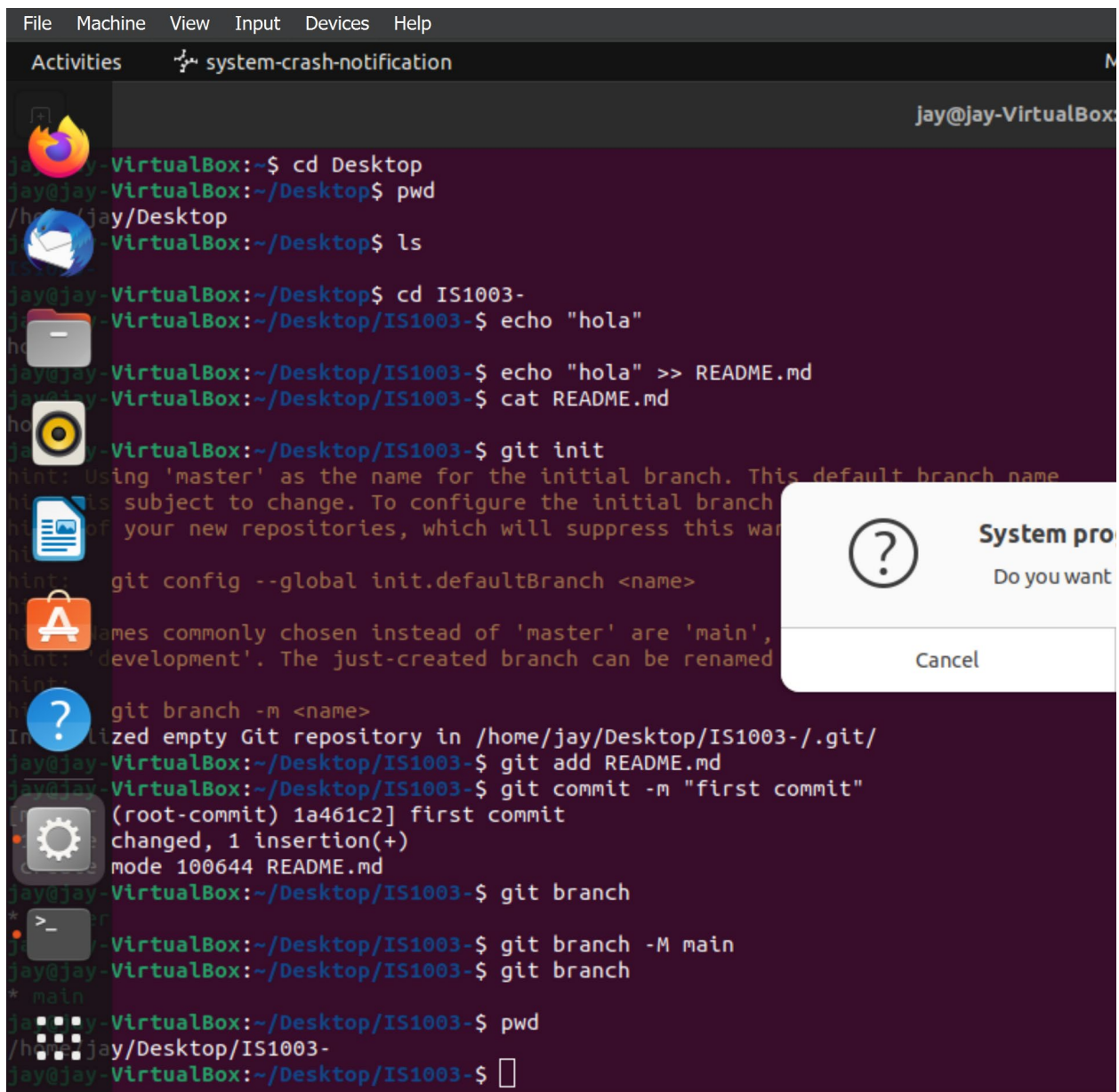


Figure 3: My repository on GitHub.

Breakpoint 3: In this step, I started by changing the directory to desktop, then made a subfolder named IS1003-. Next, to create the local repo, I had to create the README.md file. I created the README.md file by redirection. I used `echo [Hola]` and output it to the README.md file. Next, to initialize the local repo, I used the `[git init]` command. Next, I added the README.md file to the staging area (a temporary holding place for files) by using the `[git add README.md]` command. Next, to create an actual version of the repo from all the files in the staging area, I used the `[git commit -M "first commit"]` command. Finally, to verify the default branch name, I used the `[git branch]` command then to change the branch name, I used the `[git branch -M main]` command.



```
File Machine View Input Devices Help
Activities system-crash-notification
jay@jay-VirtualBox:~$ cd Desktop
jay@jay-VirtualBox:~/Desktop$ pwd
/home/jay/Desktop
jay@jay-VirtualBox:~/Desktop$ ls
IS1003-
jay@jay-VirtualBox:~/Desktop$ cd IS1003-
jay@jay-VirtualBox:~/Desktop/IS1003-$ echo "hola"
hola
jay@jay-VirtualBox:~/Desktop/IS1003-$ echo "hola" >> README.md
jay@jay-VirtualBox:~/Desktop/IS1003-$ cat README.md
hola
jay@jay-VirtualBox:~/Desktop/IS1003-$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name for your new repositories, which will suppress this warning, use
hint: the --global init.defaultBranch option. To learn more about branch naming, see
hint: https://git-scm.com/docs/git-init#_branch_naming
jay@jay-VirtualBox:~/Desktop/IS1003-$ git config --global init.defaultBranch <name>
jay@jay-VirtualBox:~/Desktop/IS1003-$ git branch -m <name>
hint: Using 'main' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name for your new repositories, which will suppress this warning, use
hint: the --global init.defaultBranch option. To learn more about branch naming, see
hint: https://git-scm.com/docs/git-init#_branch_naming
jay@jay-VirtualBox:~/Desktop/IS1003-$ git add README.md
jay@jay-VirtualBox:~/Desktop/IS1003-$ git commit -m "first commit"
[main (root-commit) 1a461c2] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
jay@jay-VirtualBox:~/Desktop/IS1003-$ git branch
* main
jay@jay-VirtualBox:~/Desktop/IS1003-$ git branch -M main
jay@jay-VirtualBox:~/Desktop/IS1003-$ git branch
* main
jay@jay-VirtualBox:~/Desktop/IS1003-$ pwd
/home/jay/Desktop/IS1003-
jay@jay-VirtualBox:~/Desktop/IS1003-$
```

Figure 4: All the different commands to create a local repo.

Breakpoint 4: To sync the local repo with my remote repo on GitHub, I used the SSH (secure shell protocol) method. With this method, you can use the SSH protocol and SSH keys. With the SSH protocol, you can connect and authenticate to remote servers and services. With the SSH keys, you can connect to GitHub without supplying your username and personal access token every time. Therefore, I was able to generate an SSH key with the [ssh-keygen -t ed25519 -C "jaykumar.bhakta@my.utsa.edu"] command. Next, I verified that the key existed in the .ssh file. Next, I copied the SSH key to add to my GitHub account.

```
jay@jay-VirtualBox:~/Desktop/IS1003-$ ssh-keygen -t ed25519 -C "jaykumar.bhakta@my.utsa.edu"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/jay/.ssh/id_ed25519):
Created directory '/home/jay/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/jay/.ssh/id_ed25519
Your public key has been saved in /home/jay/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:x31NVDJ9BjKIQ81ijuSEXdERxo0l3r77Yn6BMMYHglo jaykumar.bhakta@my.utsa.edu
The key's randomart image is:
+--[ED25519 256]--+
|    + o+X=o +o+ |
|    E = BoB  o ++|
|    o + *.+ . .o |
|    . o.*o.. o  |
|    .S++.. . .  |
|    o. ..       |
|    . .         |
|    o..         |
|    o+=.        |
+-----[SHA256]-----+
```

Figure 5: Generating an SSH key.

```
jay@jay-VirtualBox:~/.ssh$ cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIKHpjVA07GPPjYnGL7wH15R/5BBXKgB/S/hISnIwMLe8 jaykumar.bhakta@my.utsa.edu
jay@jay-VirtualBox:~/.ssh$ ^C
```


Figure 6: Verifying that the key exists and copying the key for my GitHub account.

SSH keys

New SSH key

This is a list of SSH keys associated with your account. Remove any keys that you do not recognize.

Authentication Keys



IS1003-

SHA256:x31NVDJ9BjKIQ81ijuSEXdERxo0l3r77Yn6BMMYHglo

Added on Mar 19, 2023

Never used — Read/write

SSH

Delete

Figure 7: Adding the SSH key to my GitHub account.

Breakpoint 5: To sync the repos, I used the link I copied from my GitHub account to use with the `[git remote add origin git@github.com<your repo link>]` command on my Linux terminal. Then to check my SSH connection, I used the `[cd -v]` command.

```

jay@jay-VirtualBox:~/.ssh$ cd IS1003-
bash: cd: IS1003-: No such file or directory
jay@jay-VirtualBox:~/.ssh$ cd Desktop
bash: cd: Desktop: No such file or directory
jay@jay-VirtualBox:~/.ssh$ pwd
/home/jay/.ssh
jay@jay-VirtualBox:~/.ssh$ cd ~
jay@jay-VirtualBox:~$ pwd
/home/jay
jay@jay-VirtualBox:~$ ls
Desktop  Documents  Downloads  IS1003-  Music  Pictures  Public  snap  Templates  Videos
jay@jay-VirtualBox:~$ cd IS1003-
jay@jay-VirtualBox:~/IS1003-$ cd home
bash: cd: home: No such file or directory
jay@jay-VirtualBox:~/IS1003-$ cd ~
jay@jay-VirtualBox:~$ cd Desktop
jay@jay-VirtualBox:~/Desktop$ cd IS1003-
jay@jay-VirtualBox:~/Desktop/IS1003-$ ls
README.md
jay@jay-VirtualBox:~/Desktop/IS1003-$ git remote add origin ^[[200~git@github.com:Jay-Bhakta/IS1003-.git~
jay@jay-VirtualBox:~/Desktop/IS1003-$ git remote -v
origin  git@github.com:Jay-Bhakta/IS1003-.git~ (fetch)
origin  git@github.com:Jay-Bhakta/IS1003-.git~ (push)
jay@jay-VirtualBox:~/Desktop/IS1003-$

```

Figure 8: Syncing the SSH connection and checking on the links.

Breakpoint 6: This step was a challenge for me because I thought that I did everything correctly, and when I went to type the {git push -u origin main} command, I encountered an error that stated that permission was denied and I had to check if I had access rights. I panicked when that happened and was confused about what I did wrong. I posted on slack to receive help and used what I was informed to fix the error, but the error still showed. The next day I tried again and looked at the tips section on Lab 2, but it didn't fix anything, so I contacted you (the professor). While I was waiting for a reply, I started over and created a new repo on GitHub and a new file with a different name on my local machine. Then, I followed the instructions and was able to successfully complete step 6. The push command is a command that allows you to depict changes from your local repo files in Linux VM to your GitHub account.

```
jay@jay-VirtualBox:~/Desktop/IS1003$ echo "hello"
hello
jay@jay-VirtualBox:~/Desktop/IS1003$ echo "hello" >> README.md
jay@jay-VirtualBox:~/Desktop/IS1003$ cat README.md
hello
jay@jay-VirtualBox:~/Desktop/IS1003$ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /home/jay/Desktop/IS1003/.git/
jay@jay-VirtualBox:~/Desktop/IS1003$
jay@jay-VirtualBox:~/Desktop/IS1003$ git add README.md
jay@jay-VirtualBox:~/Desktop/IS1003$ git commit -m "first commit"
[master (root-commit) 6420a85] first commit
 1 file changed, 1 insertion(+)
 create mode 100644 README.md
jay@jay-VirtualBox:~/Desktop/IS1003$ git branch
* master
jay@jay-VirtualBox:~/Desktop/IS1003$ git branch -M main
jay@jay-VirtualBox:~/Desktop/IS1003$ git branch
* main
```

Figure 9: My work from Step 3.

```

jay@jay-VirtualBox:~/Desktop/IS1003$ ssh-keygen -t ed25519 -C "jaykumar.bhakta@my.utsa.edu"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/jay/.ssh/id_ed25519):
/home/jay/.ssh/id_ed25519 already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/jay/.ssh/id_ed25519
Your public key has been saved in /home/jay/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:p1xFVJ3sRXGkYFe83RY6WGAjk4z0inQL3YJo3ZgMFbM jaykumar.bhakta@my.utsa.edu
The key's randomart image is:
+--[ED25519 256]--+
|  ..+o.o.=*.*BB |
|    = 0.oo+o.+ ==|
|  o E = o  + + * |
|  . . + +  o o o+|
|    . oS o  ..   |
|    . +         |
|    o           |
|-----+
+----[SHA256]-----+
jay@jay-VirtualBox:~/Desktop/IS1003$ cd ~
jay@jay-VirtualBox:~$ cd Desktop
jay@jay-VirtualBox:~/Desktop$ cd IS1003
jay@jay-VirtualBox:~/Desktop/IS1003$ ls
README.md

```

Figure 10: My work from Step 5.

```

jay@jay-VirtualBox:~/Desktop/IS1003$ cd Jay-Bhakta
bash: cd: Jay-Bhakta: No such file or directory
jay@jay-VirtualBox:~/Desktop/IS1003$ cd .ssh
bash: cd: .ssh: No such file or directory
jay@jay-VirtualBox:~/Desktop/IS1003$ ls -al
total 16
drwxrwxr-x 3 jay jay 4096 Mar 20 18:30 .
drwxr-xr-x 4 jay jay 4096 Mar 20 17:27 ..
drwxrwxr-x 8 jay jay 4096 Mar 20 18:32 .git
-rw-rw-r-- 1 jay jay  6 Mar 20 18:20 README.md
jay@jay-VirtualBox:~/Desktop/IS1003$ cd /home/Jay-Bhakta/.ssh
bash: cd: /home/Jay-Bhakta/.ssh: No such file or directory
jay@jay-VirtualBox:~/Desktop/IS1003$ cd /home/jay/.ssh
jay@jay-VirtualBox:~/.ssh$ ls -al
total 20
drwx----- 2 jay jay 4096 Mar 19 17:12 .
drwxr-x--- 16 jay jay 4096 Mar 20 18:32 ..
-rw----- 1 jay jay  419 Mar 20 18:34 id_ed25519
-rw-r--r-- 1 jay jay 109 Mar 20 18:34 id_ed25519.pub
-rw-r--r-- 1 jay jay 142 Mar 19 17:12 known_hosts
jay@jay-VirtualBox:~/.ssh$ cat id_ed25519.pub
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIHC9ukUYgnmyYr0055HkyThmH80DcMff0tceF19zor5h jaykumar.bhakta@my.utsa.edu

```

Figure 11: My work for Step 4&5.


```

jay@jay-VirtualBox:~$ cd Desktop
jay@jay-VirtualBox:~/Desktop$ ls
IS1003  IS1003-
jay@jay-VirtualBox:~/Desktop$ cd IS1003
jay@jay-VirtualBox:~/Desktop/IS1003$ ls
README.md
jay@jay-VirtualBox:~/Desktop/IS1003$ git remote add origin git@github.com:Jay-Bhakta/IS1003.git
jay@jay-VirtualBox:~/Desktop/IS1003$ git remote -v
origin  git@github.com:Jay-Bhakta/IS1003.git (fetch)
origin  git@github.com:Jay-Bhakta/IS1003.git (push)
jay@jay-VirtualBox:~/Desktop/IS1003$ git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 224 bytes | 224.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To github.com:Jay-Bhakta/IS1003.git
 * [new branch]      main -> main
Branch 'main' set up to track remote branch 'main' from 'origin'.

```

Figure 12: My work from Step 5&6.

Breakpoint 7: On GitHub, I clicked the edit icon to delete “hello” and replaced it with “howye doin”. I then to confirm I scrolled down and clicked “Commit changes”.

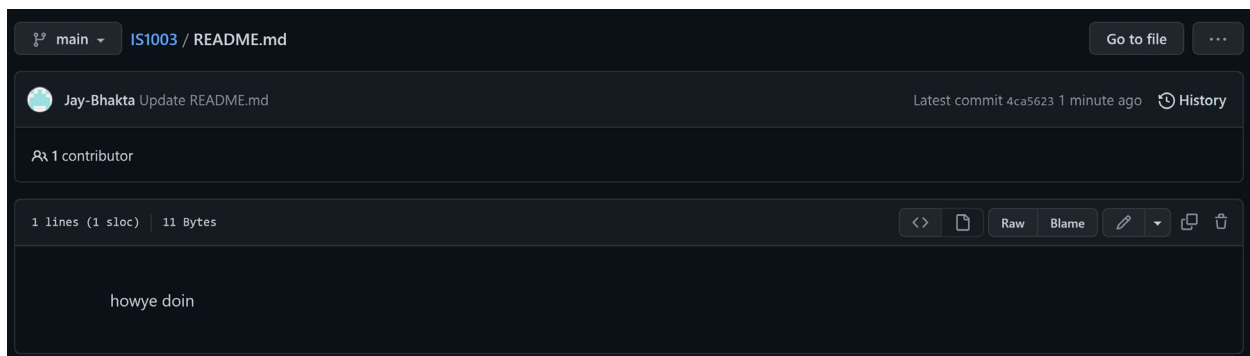


Figure 13: My work from step 7.

Breakpoint 8: I edited the message that was sent from my local repo on my remote repo. To see the change on my local repo, I need to pull the transition from my remote repo. I do this by using the “git pull origin main.” Then, to see the message, I cat the file so [cat README.md]. The pull command is a command which allows you to depict changes made on your remote repo onto your local repo.


```

jay@jay-VirtualBox:~/Desktop/IS1003$ git pull origin main
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (3/3), 633 bytes | 633.00 KiB/s, done.
From github.com:Jay-Bhakta/IS1003
* branch          main      -> FETCH_HEAD
   6420a85..4ca5623  main      -> origin/main
Updating 6420a85..4ca5623
Fast-forward
 README.md | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
jay@jay-VirtualBox:~/Desktop/IS1003$ cat README.md
howye doin

```

Figure 14: My work from step 8.

Breakpoint 9: For this I had to choose a GitHub repo to fork and clone. On GitHub, I clicked on the explore tab then the topics tab. I chose the Algorithm topic because I wanted to see how algorithms are created, what they look like, and what they accomplish. From there, I chose a repo named “interview” because it included a summary of basic characteristics needed for job seeking and technology such as program library, algorithm, system, network, and more. The clone command allows users to work on files locally with whatever software and without impacting their remote repos.

```

jay@jay-VirtualBox:~/Desktop/IS1003$ git clone git@github.com:Jay-Bhakta/interview.git
Cloning into 'interview'...
remote: Enumerating objects: 1512, done.
remote: Counting objects: 100% (447/447), done.
remote: Compressing objects: 100% (92/92), done.
remote: Total 1512 (delta 365), reused 355 (delta 355), pack-reused 1065
Receiving objects: 100% (1512/1512), 5.26 MiB | 2.72 MiB/s, done.
Resolving deltas: 100% (874/874), done.

```

Figure 15: My work for the git tasks/output for step 9.

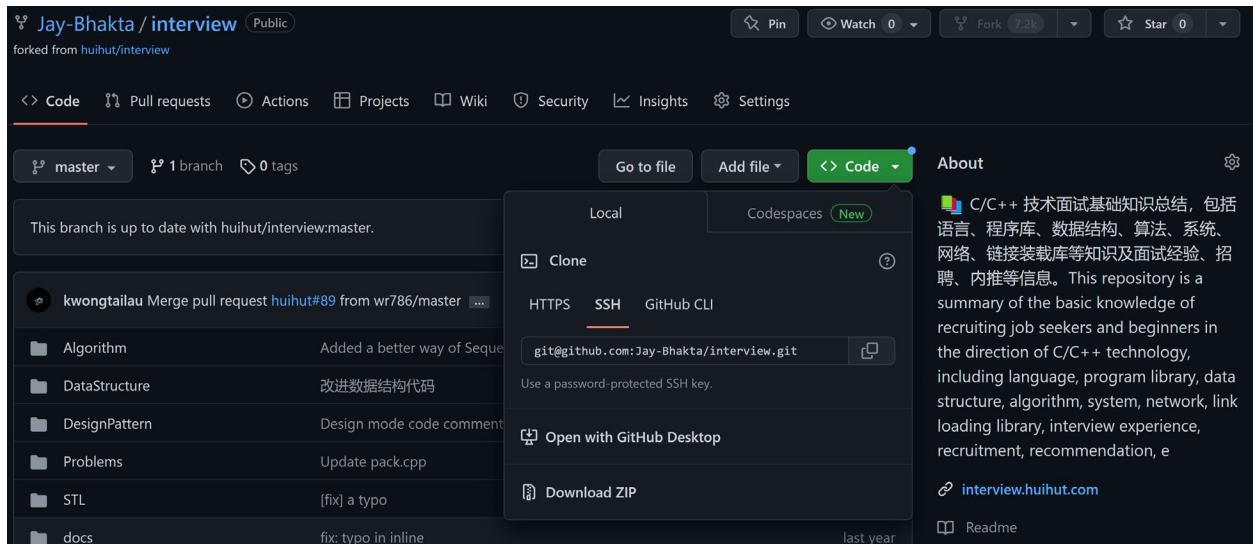


Figure 16: My work for GitHub in step 9.

LIMITATIONS/CONCLUSION

I was able to complete all the steps in this lab. The objective of this lab was to be able to create a GitHub account, setup a git on our local machines, and synchronize our local repo with out GitHub repo. This lab was like a introduction of what programmers do during software development. And this process is very important not only for software developers but also for information technology professionals that work with evolving technology.

REFERENCES

COLLABORATION

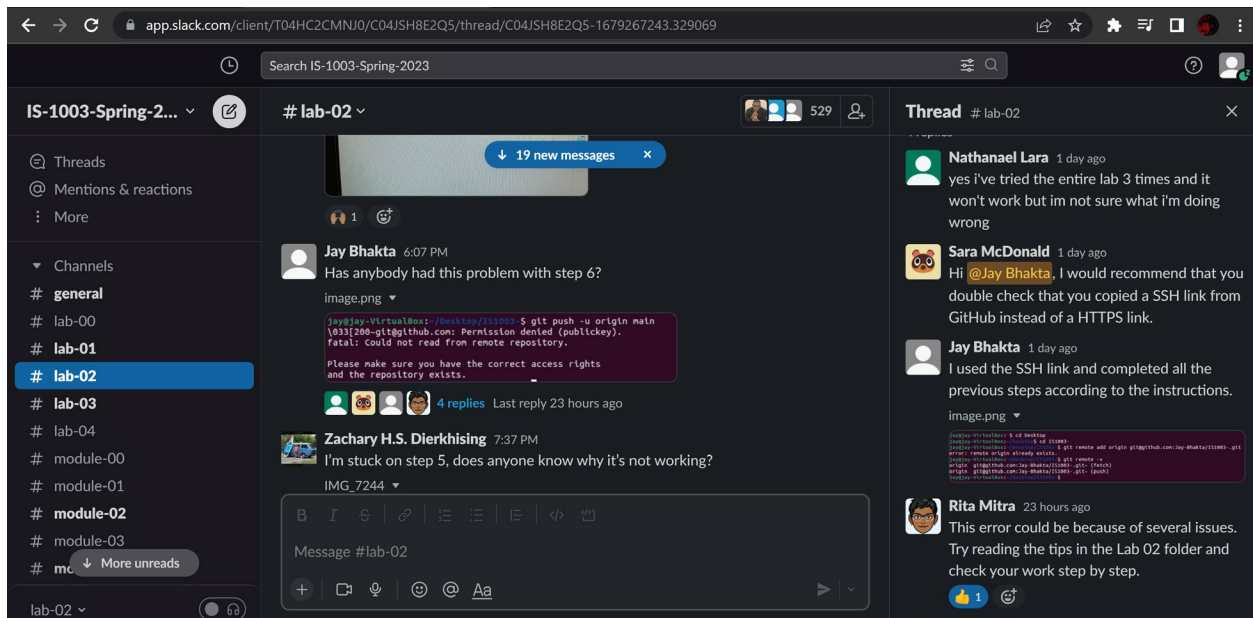


Figure 17: Proof of collaboration using slack.