Subject Name: **Source Code Management**

Subject Code: **22CS003**

Session: **2023-2024**

Department: **DCSE**


**Submitted To:**

**Dr. Himanshi Babbar**
**Professor**
**Chitkara University, Punjab**

**Submitted By:**

**Komal Pahuja**
**2310992057**
**Sem-II (2023-27)**

## **Task 2 Submission** (Week 10)

1. Create a distributed Repository and add members in project team
2. Open and close a pull request.
3. Each project member shall create a pull request on a team members repo and close pull requests generated by team members on own Repo as a maintainer.
4. Publish and print network graphs

Team Member 1 Name: Komal Pahuja Roll No. 2310992042
Team Member 2 Name: Mehak Roll No. 2310992056
Team Member 3 Name:Diksha Vadehra Roll No. 2310992592

# INDEX

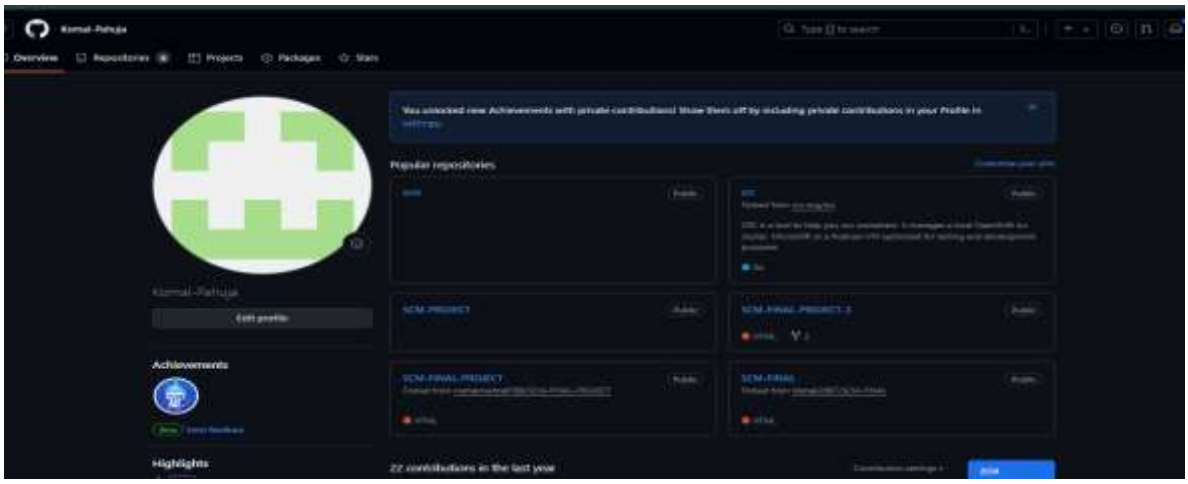
| Sr. No. | Practical Name | Teacher Sign |
|---|---|---|
| 1 | Create a distributed Repository and add members in project team | |
| 2 | Open and close a pull request | |
| 3 | Create a pull request on a team members repo and close pull requests generated by team members on own Repo as a maintainer | |
| 4 | Publish and print network graphs | |

# EXPERIMENT 1

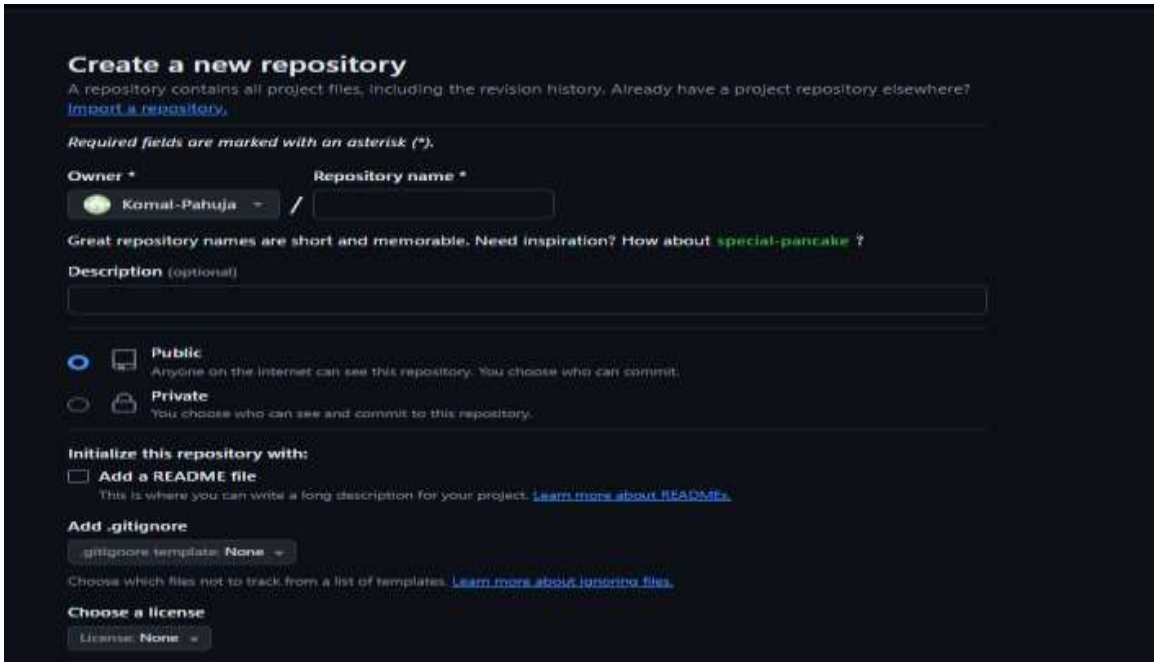
**Aim:** Create a distributed Repository and add members in project team

**Procedure:**

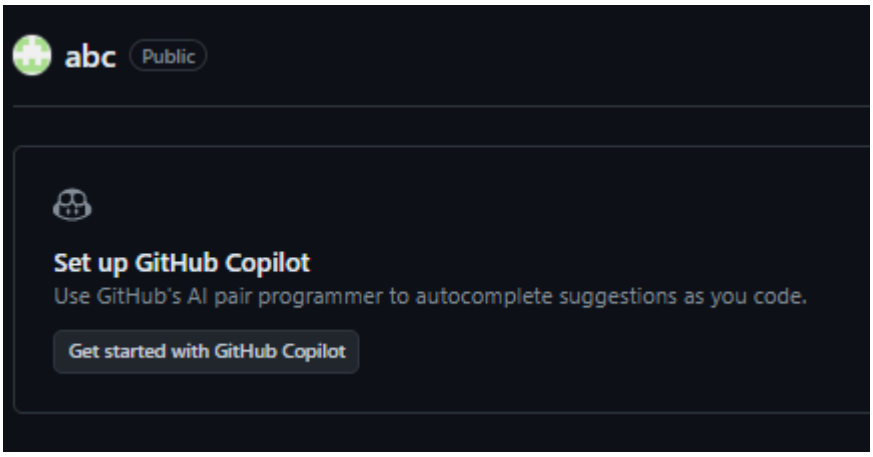
1. After logging in, you will see the homepage as displayed below in your GitHub account. Select Repositories from the menu bar.
2. Click on the 'New' button in the top right corner.



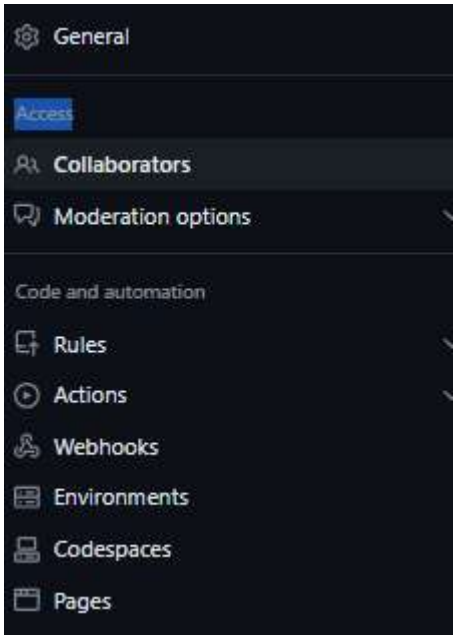
3. Add the repository's description and enter the repository name.
4. Choose whether to make the repository public or private.




5. Choose the import code option if you wish to import code from an existing repository.

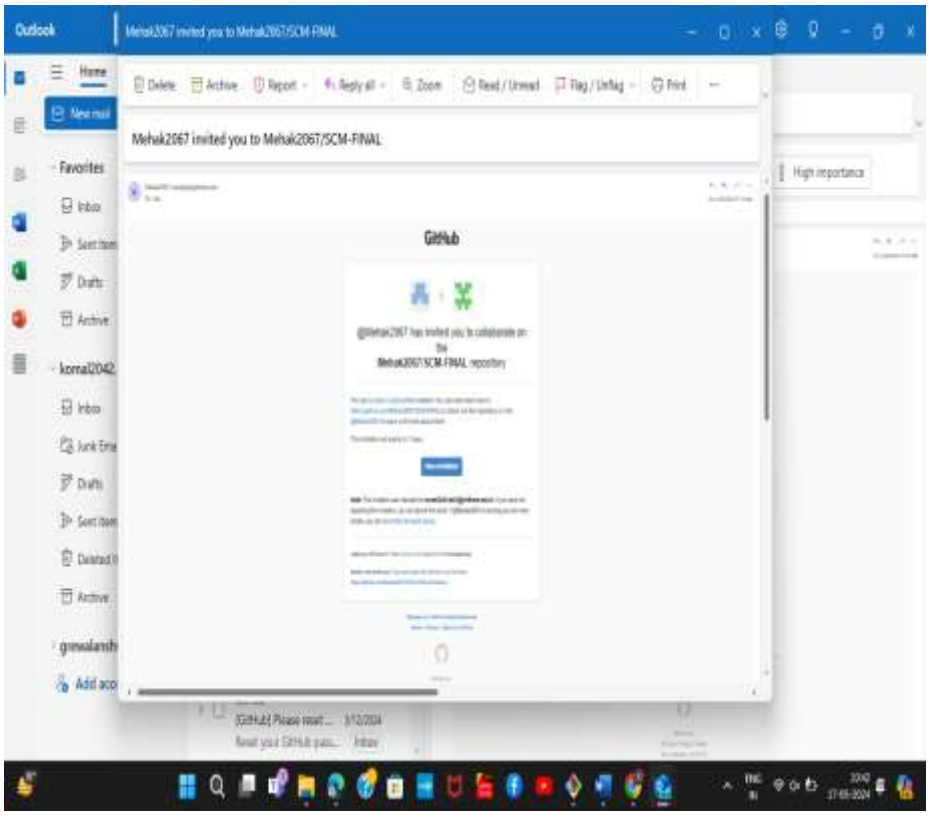



6. Your repository has now been successfully created.
7. Open your repository and use the settings option from the navigation bar to add members. And select the Collaborators menu item from the access tab.

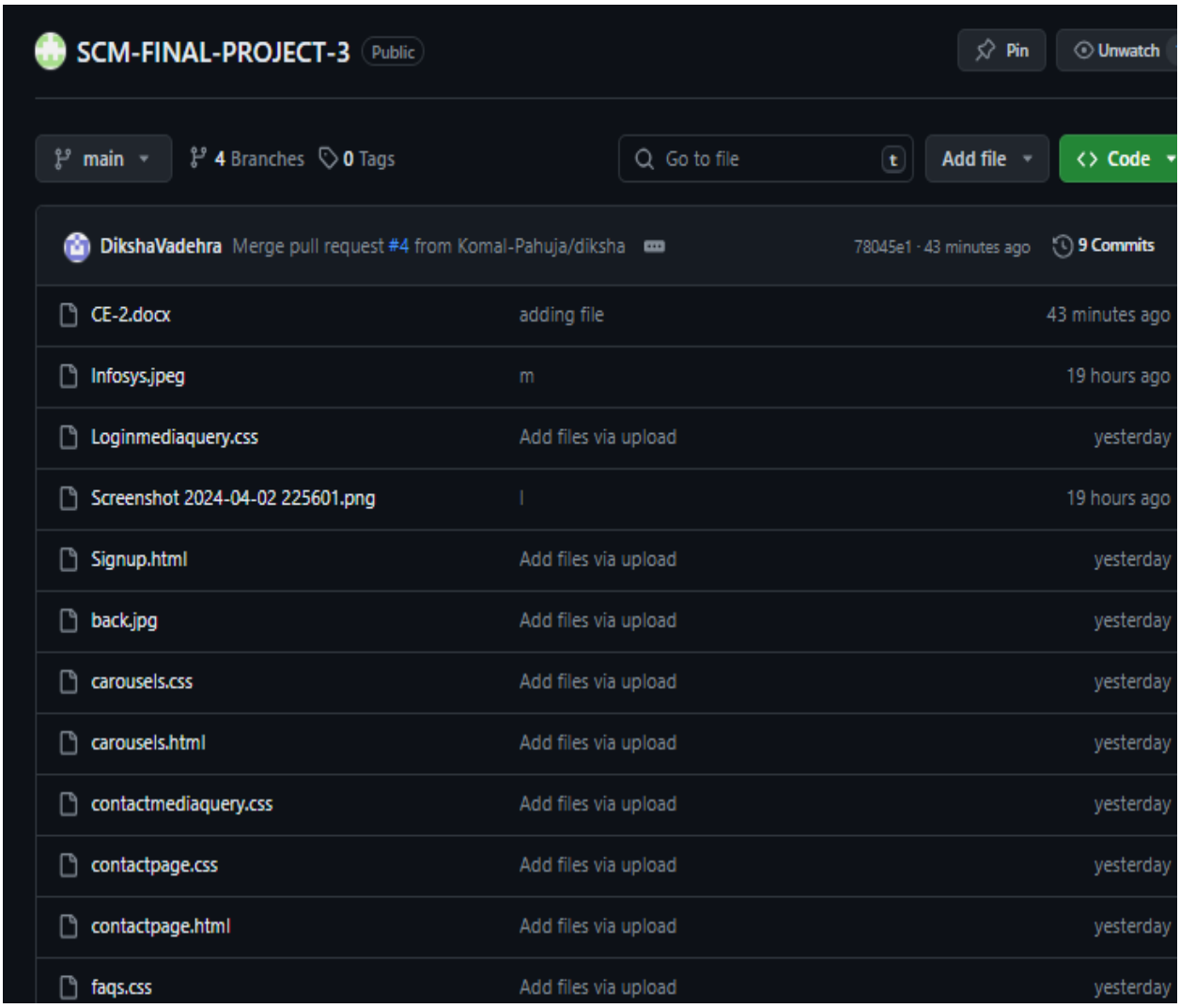



8. Open your repository and select the settings option from the navigation bar to add members. And select the Collaborators option from the access tab.


9. Open your GitHub email account in order to accept your team member's invitation.




10. The option you are now permitted to push will be displayed to you. Each person is now prepared to make a contribution to the project.


SCM-FINAL-PROJECT-3
Public
Pin
Unwatch
main
4 Branches
0 Tags
Go to file
t
Add file
Code
DikshaVadehra Merge pull request #4 from Komal-Pahuja/diksha
78045e1 · 43 minutes ago
9 Commits
CE-2.docx adding file 43 minutes ago
Infosys.jpeg m 19 hours ago
Loginmediaquery.css Add files via upload yesterday
Screenshot 2024-04-02 225601.png I 19 hours ago
Signup.html Add files via upload yesterday
back.jpg Add files via upload yesterday
carousels.css Add files via upload yesterday
carousels.html Add files via upload yesterday
contactmediaquery.css Add files via upload yesterday
contactpage.css Add files via upload yesterday
contactpage.html Add files via upload yesterday
faqs.css Add files via upload yesterday

# EXPERIMENT 2

**Aim:** Open and Close a Pull Request

**Procedure:**

1. To initiate a pull request, first create a new branch using the git branch *branchname* option.

```
Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop
$ git clone https://github.com/Mahi-Awasthi/SCM-Project.git
Cloning into 'SCM-Project'...
remote: Enumerating objects: 34, done.
remote: Counting objects: 100% (34/34), done.
remote: Compressing objects: 100% (34/34), done.
remote: Total 34 (delta 1), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (34/34), 2.76 MiB | 1.73 MiB/s, done.
Resolving deltas: 100% (1/1), done.
```

2. Once a new branch has been created, add a file to it or modify an existing file.

```
Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop
$ cd SCM-Project

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/SCM-Project (main)
$ ls
'3D illustrations.jpeg'       WhyteInktrap-Black.ttf   mm.png
 3dc.png                      dino.png                 phone-call.png
 BuildM3.css                  email.png                pp.png
'CONTACT INFO.css'           'facebook (1).png'        project2.css
'CONTACT INFO.html'           hm.png                   rough.html
 I-got-an-idea-unscreen.gif   hmm.png                 'twitter (1).png'
'MAIN WEBPAGE.html'           hmmm.png                 whatsapp.png
 Manually.css                 hmmmm.png                youtube.png
 Manually.html                instagram.png           'zip project'/
'Screenshot (17).png'         location.png
'Thomas Morgan.jpeg'          mang.png

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/SCM-Project (main)
$ git branch mahi

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/SCM-Project (main)
$ git checkout mahi
Switched to branch 'mahi'

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/SCM-Project (mahi)
$ git status
On branch mahi
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        zip project/

nothing added to commit but untracked files present (use "git add" to track)
```

3. Commit the changes to the local repository.

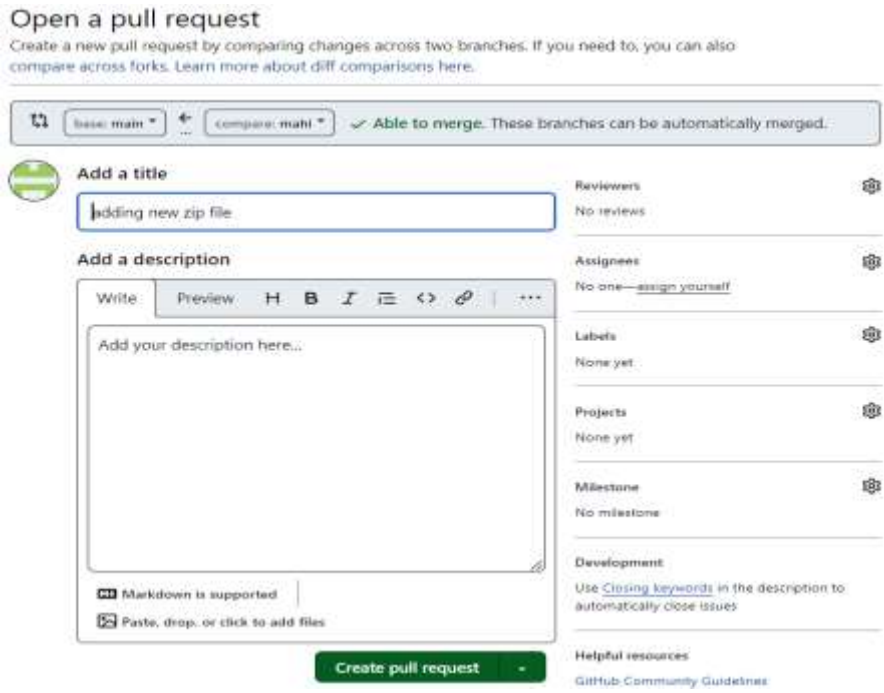
```
Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/SCM-Project (mahi)
$ git add .
warning: in the working copy of 'zip project/icon_updates_small.svg', LF will be
 replaced by CRLF the next time Git touches it

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/SCM-Project (mahi)
$ git commit -m"adding new zip file"
[mahi d74188a] adding new zip file
 40 files changed, 1877 insertions(+)
 create mode 100644 zip project/1608678_envelope_icon.png
 create mode 100644 zip project/5856.jpg
 create mode 100644 zip project/AddressList.css
 create mode 100644 zip project/AddressList.html
 create mode 100644 zip project/Chat-unscreen.gif
 create mode 100644 zip project/Dynamic Calendar JavaScript/script.js
 create mode 100644 zip project/Dynamic Calendar JavaScript/style.css
 create mode 100644 zip project/I-got-an-idea-unscreen.gif
 create mode 100644 zip project/Intercom-Live-Chat-unscreen.gif
 create mode 100644 zip project/Mass Mailing.css
 create mode 100644 zip project/Mass Mailing.html
 create mode 100644 zip project/My Account.css
```

4. After pushing the new branch to the main repository, use the git push origin branch name option.

```
Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/SCM-Project (mahi)
$ git push origin mahi
Enumerating objects: 44, done.
Counting objects: 100% (44/44), done.
Delta compression using up to 4 threads
Compressing objects: 100% (43/43), done.
Writing objects:  86% (37/43), 11.93 MiB | 416.00 KiB/s
Writing objects: 100% (43/43), 20.15 MiB | 568.00 KiB/s, done.
Total 43 (delta 5), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (5/5), completed with 1 local object.
remote:
remote: Create a pull request for 'mahi' on GitHub by visiting:
remote:      https://github.com/Mahi-Awasthi/SCM-Project/pull/new/mahi
remote:
To https://github.com/Mahi-Awasthi/SCM-Project.git
 * [new branch]      mahi -> mahi
```

5. After pushing the new branch, GitHub will either prompt you to create a pull request or you can create one on your own.
6. GitHub will detect any conflicts and ask you to enter a description of your pull request.




7. After opening a pull request all the team members will be sent the request if they want to merge or close the request.
8. If the team member chooses not to merge your pull request they will close your pull request.
9. To close the pull request simply click on close pull request and add comment/ reason why you closed the pull request.
10. You can see all the pull request generated and how they were dealt with by clicking on pull request option.

# EXPERIMENT 3

**Aim:** Create a pull request on a team member's repo and close pull requests generated by team members on own Repo as a maintainer

**Procedure:**

1. Make the necessary repository changes, then add and commit them to a new branch in the local repository.

```
Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop
$ mkdir 2310992057-scm

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop
$ cd 2310992057-scm/

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm
$ git clone https://github.com/Mahi-Awasthi/SCM-Project-3.git
Cloning into 'SCM-Project-3'...
remote: Enumerating objects: 3, done.
remote: Counting objects: 100% (3/3), done.
remote: Compressing objects: 100% (2/2), done.
remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (3/3), 9.22 MiB | 446.00 KiB/s, done.

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm
$ ls
SCM-Project-3/
```

```
Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm
$ cd SCM-Project-3

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm/SCM-Project-3 (ma
in)
$ git branch mahi

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm/SCM-Project-3 (main)
$ git branch
  mahi
* main

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm/SCM-Project-3 (main)
$ git checkout mahi
Switched to branch 'mahi'

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm/SCM-Project-3 (mahi)
$ git branch
* mahi
  main

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm/SCM-Project-3 (mahi)
$ pwd
/c/Users/Dell/OneDrive/Desktop/2310992057-scm/SCM-Project-3
```

```
Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm/SCM-Project-3 (mahi)
$ ls
'ZIP (1).zip'  'zip project'/

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm/SCM-Project-3 (mahi)
$ git add .
warning: in the working copy of 'zip project/icon_updates_small.svg', LF will be re
placed by CRLF the next time Git touches it

Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm/SCM-Project-3 (mahi)
$ git status
On branch mahi
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        new file:   zip project/1608678_envelope_icon.png
        new file:   zip project/5856.jpg
        new file:   zip project/AddressList.css
        new file:   zip project/AddressList.html
```

```
Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm/SCM-Project-3 (mahi)
$ git commit -m"Adding new zip file"
[mahi 1bd19c4] Adding new zip file
 40 files changed, 1877 insertions(+)
 create mode 100644 zip project/1608678_envelope_icon.png
 create mode 100644 zip project/5856.jpg
 create mode 100644 zip project/AddressList.css
 create mode 100644 zip project/AddressList.html
 create mode 100644 zip project/Chat-unscreen.gif
 create mode 100644 zip project/Dynamic Calendar JavaScript/script.js
 create mode 100644 zip project/Dynamic Calendar JavaScript/style.css
 create mode 100644 zip project/I-got-an-idea-unscreen.gif
 create mode 100644 zip project/Intercom-Live-Chat-unscreen.gif
 create mode 100644 zip project/Mass Mailing.css
 create mode 100644 zip project/Mass Mailing.html
 create mode 100644 zip project/My Account.css
 create mode 100644 zip project/My Account.html
 create mode 100644 zip project/Privacy Policy.css
```

2. Push the modified branch using git push origin *branchname*.

```
Dell@DESKTOP-HLP1MEL MINGW64 ~/OneDrive/Desktop/2310992057-scm/SCM-Project-3 (mahi)
$ git push origin mahi
Enumerating objects: 44, done.
Counting objects: 100% (44/44), done.
Delta compression using up to 4 threads
Compressing objects: 100% (43/43), done.
Writing objects: 100% (43/43), 20.15 MiB | 1.07 MiB/s, done.
Total 43 (delta 4), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (4/4), done.
remote:
remote: Create a pull request for 'mahi' on GitHub by visiting:
remote:      https://github.com/Mahi-Awasthi/SCM-Project-3/pull/new/mahi
remote:
To https://github.com/Mahi-Awasthi/SCM-Project-3.git
 * [new branch]      mahi -> mahi
```

3. Open a pull request by following the procedure from the above experiment.
4. Each team member will be able to see the pull request after it has been created.
5. Ask your team member to login to his/her Github account.
6. In the pull request menu, they'll see a new notification.
7. Click on it. The pull request generated by you will be visible to them.
8. Select "Pull Request." There will be a choice between merging the pull request with the main branch or closing it.
9. All team members' access to the main branch will be updated upon choosing the merge branch option.
10. Closing the pull request results in it not being accepted and not being merged into the main branch.
11. The procedure is comparable to when you close and merge the pull request. All it takes is for someone else to carry it out.
12. With this, we have finished creating and closing pull requests. Additionally, we wrap up the pull request's merging into the main branch.

# EXPERIMENT 4

**Aim:** Publish and print network graphs

**Theory:**

The network graph is a valuable feature for developers on GitHub. This tool displays the repository network's branch history, including root and fork branches with network-specific commits.

Graphs for a repository include data on traffic, dependencies, contributors, commits, forks, and network activity. Maintaining a repository can provide valuable insights on user behavior and motivations.

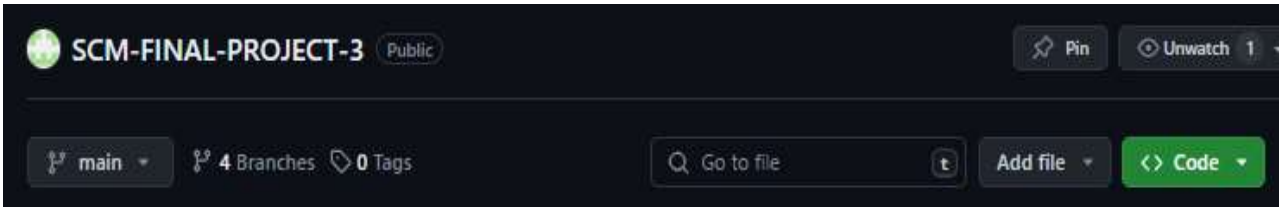
Some repository graphs are available only in public repositories with GitHub Free:

• Pulse • Contributors • Traffic • Commits • Code frequency • Network

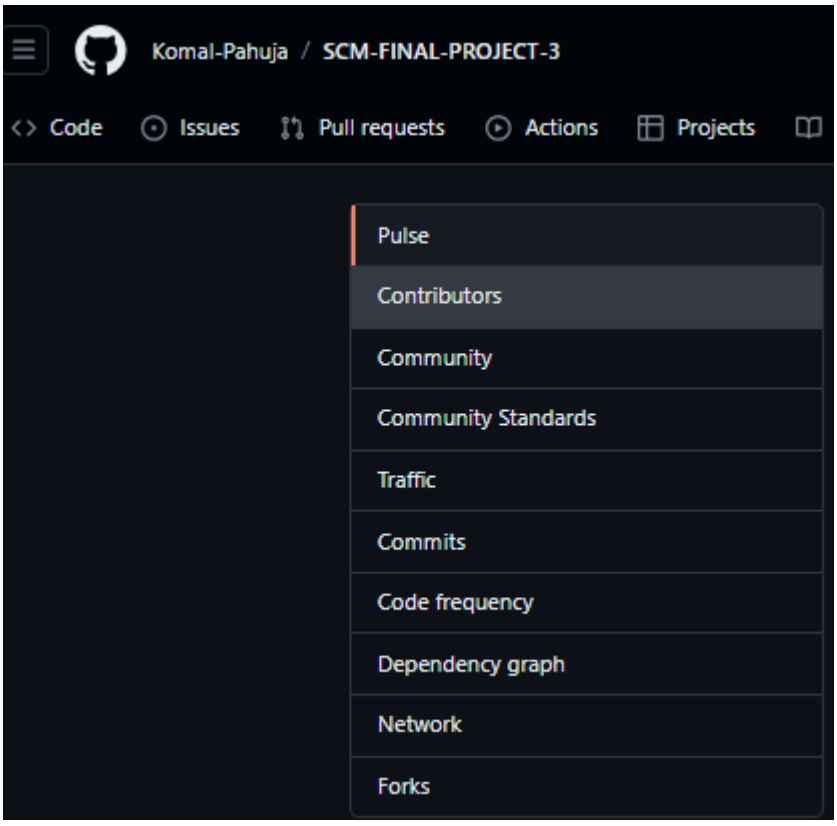
**Procedure:**

**Accessing the network graphs of respective repository:**

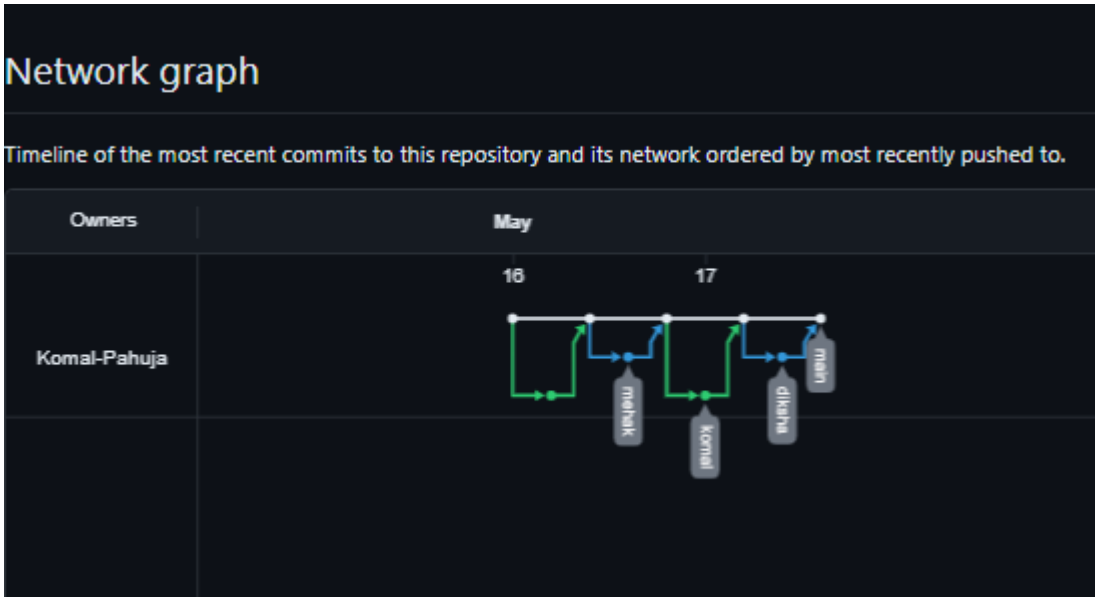
1. On GitHub.com, navigate to the main page of the repository.
2. Under your repository name, click Insights.




3. At the left sidebar, click on Network.




4. The network graph for your repository shows the history of all branches, including the original repository and forks with unique commits.

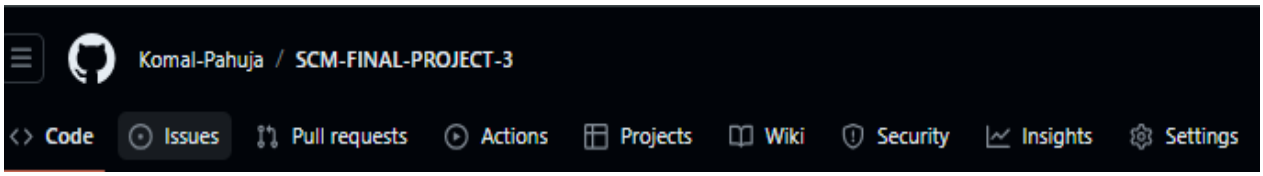



**Listing the forks of a repository**

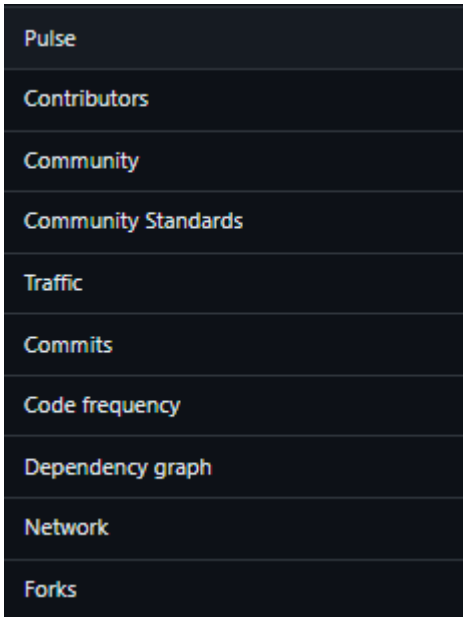
Forks are listed alphabetically by the username of the person who forked the repository

Clicking the number of forks shows you the full network. From there you can click "members" to see who forked the repo.

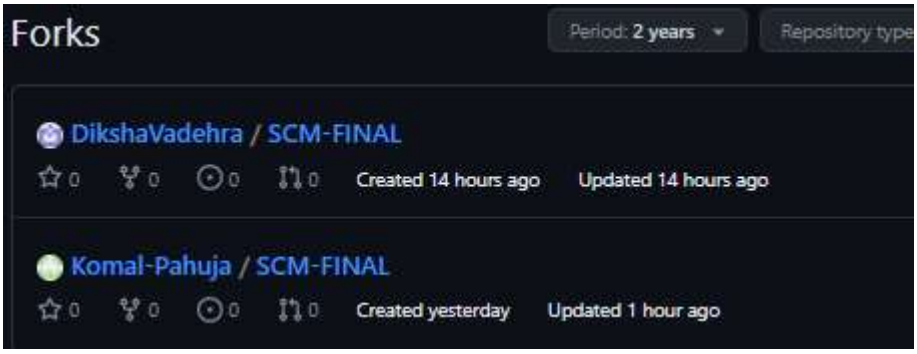
1. On GitHub.com, navigate to the main page of the repository.




2. Under your repository name, click Insights.
3. In the left sidebar, click Forks.




4. Here you can see all the forks.




**Viewing the dependencies of a repository**

The dependency tree provides insight into the code that your repository relies on.

Most software relies on code created and maintained by other developers, sometimes known as a supply chain. Consider utilities, libraries, and frameworks.

Your code relies on these dependencies, so any problems or vulnerabilities can impact it.

It is critical to assess and maintain these dependencies.