A Project report on

The coffee shop

with

Source Code Management

(CS181)

Submitted by

Team Member 1	Khushi	2110992029
Team Member 2	Isha	2110992064
Team Member 3	Vanshika	2110992051
Team Member 4	Prakriti	2110992053

Department of Computer Science & Engineering

Chitkara University Institute of Engineering and Technology, Punjab

Jan- June (2021-22)

Institute/School Name	**Chitkara University Institute of Engineering and Technology**		
Department Name	**Department of Computer Science & Engineering**		
Programme Name	**Bachelor of Engineering (B.E.), Computer Science & Engineering**		
Course Name	**Source Code Management**	Session	**2021-22**
Course Code	**CS181**	Semester/Batch	**2nd/2021**
Vertical Name	**zeta**	Group No	G-27
Course Coordinator	**Dr. Sachendra Singh**		
Faculty Name	**Dr. Sachendra Singh**		

Submitted by:

Name: khushi

Table of Content

1. Version control with Git

What is GIT and why is it used?

Git is a version control system that is widely used in the programming world. It is used for tracking changes in the source code during software development. It was developed in 2005 by Linus Torvalds, the creator of the Linux operating system kernel.

Git is a speedy and efficient distributed VCS tool that can handle projects of any size, from small to very large ones. Git provides cheap local branching, convenient staging areas, and multiple workflows. It is free, open-source software that lowers the cost because developers can use Git without paying money. It provides support for non-linear development. Git enables multiple developers or teams to work separately without having an impact on the work of others.

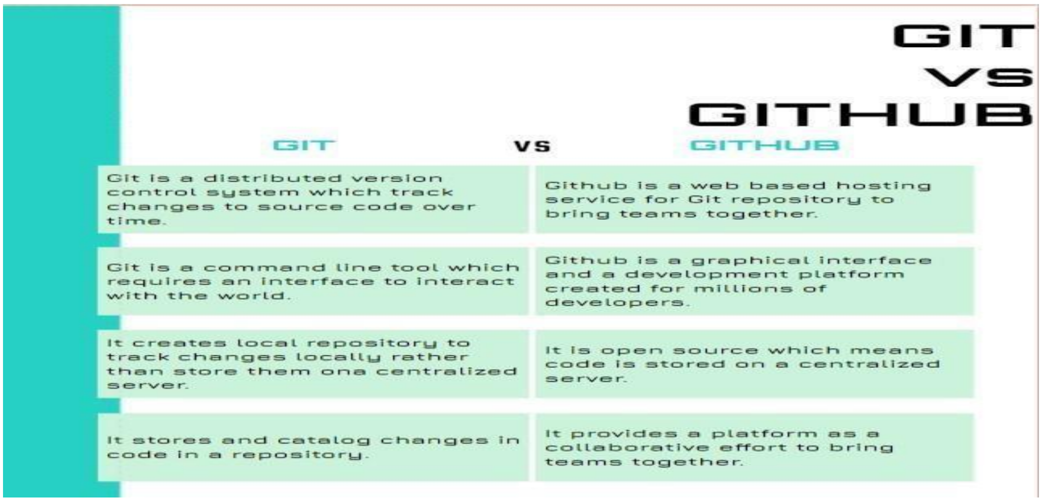
Git is an example of a distributed version control system (DVCS) (hence Distributed Version Control System).

What is GITHUB?

It is the world’s largest open-source software developer community platform where the users upload their projects using the software Git.

GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere. This tutorial teaches you GitHub essentials like repositories, branches, commits, and pull requests.

What is the difference between GIT and GITHUB?



GIT
VS
GITHUB

GIT	GITHUB
Git is a distributed version control system which track changes to source code over time.	Github is a web based hosting service for Git repository to bring teams together.
Git is a command line tool which requires an interface to interact with the world.	Github is a graphical interface and a development platform created for millions of developers.
It creates local repository to track changes locally rather than store them ona centralized server.	It is open source which means code is stored on a centralized server.
It stores and catalog changes in code in a repository.	It provides a platform as a collaborative effort to bring teams together.

What is Repository?

A repository is a directory or storage space where your projects can live. Sometimes GitHub users shorten this to “repo.” It can be local to a folder on your computer, or it can be a storage space on GitHub or another online host. You can keep code files, text files, image files, you name it, inside a repository.

What is Version Control System (VCS)?

A version control system is a tool that helps you manage “versions” of your code or changes to your code while working with a team over remote distances. Version control keeps track of every modification in a special kind of database that is accessible to the version control software. Version control software (VCS) helps you revert back to an older version just in case a bug or issue is introduced to the system or fixing a mistake without disrupting the work of other team members.

Types of VCS

1. Local Version Control System
2. Centralized Version Control System
3. Distributed Version Control System

I. **Local Version Control System:** Local Version Control System is located in your local machine. If the local machine crashes, it would not be possible to retrieve the files, and all the information will be lost. If anything happens to a single version, all the versions made after that will be lost.

II. **Centralized Version Control System:** In the Centralized Version Control Systems, there will be a single central server that contains all the files related to the project, and many collaborators checkout files from this single server (you will only have a working copy). The problem with the Centralized Version Control Systems is if the central server crashes, almost everything related to the project will be lost.

III. **Distributed Version Control System:** In a distributed version control system, there will be one or more servers and many collaborators similar to the centralized system. But the difference is, not only do they check out the latest version, but each collaborator will have an exact copy of the main repository on their local machines. Each user has their own repository and a working copy. This is very useful because even if the server crashes we would not lose everything as several copies are residing in several other computers.

2. Problem Statement

One common management-related problem faced by coffee shop owners is slow service, especially in establishments that offer many customized coffee preparations. The best POS systems also make management hassle-free by tracking customer information, using traffic patterns to forecast staffing needs, adjusting pricing, generating reports that contain actionable information, facilitating tip reconciliation, and integrating with other solutions, such as accounting software.

3. Objective

PROJECT NAME: COFFEE SHOP

The objective of this project is to provide a coffee shop that will not only be known for serving great coffee but also an environment to relax, socialize, and network. This goals of this project will be:

- To lease, renovate and open an environmentally friendly coffeeshop that will attract customers. The shop will have a low economic footprint and consider the environment as the first priority of the business

- To open with-in six months at a cost not to exceed 3% of the budgeted $400,000.
- Design a seating capacity for 30 people all with available outlets for electronic devices and shop wide Wi Fi service

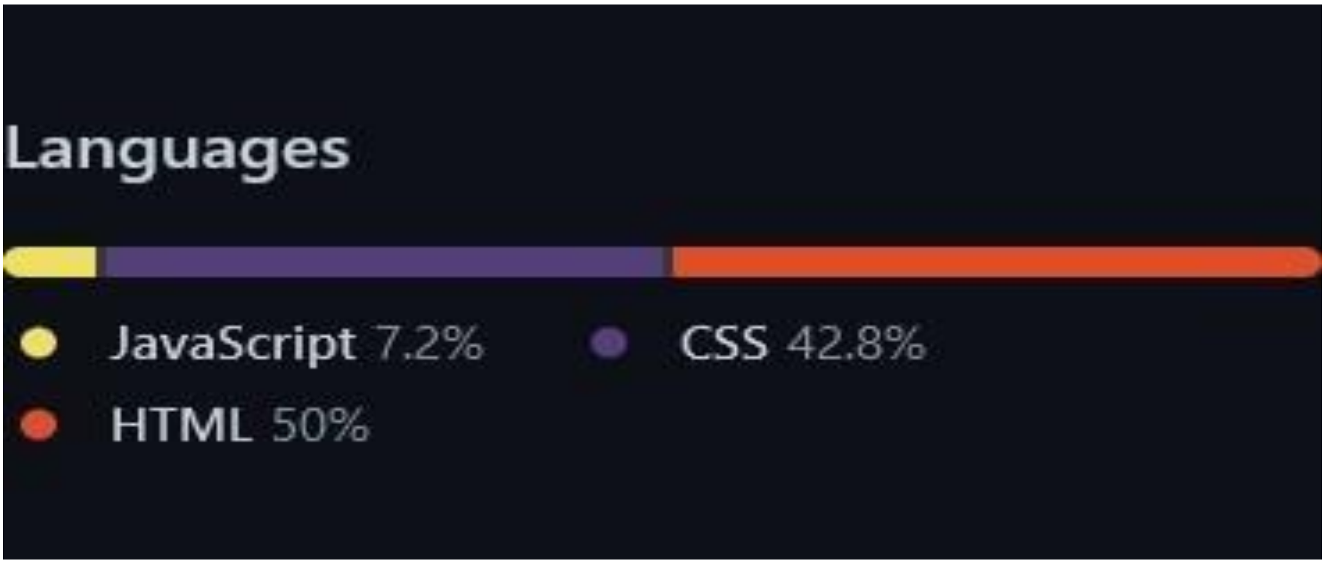
- We will be offering many different types of coffee from all around the world

- Employ a volunteer program in which customers can pay for their order with lab or. This will ensure that each and every person is accommodated and has an opportunity to enjoy coffee and food in our shop regardless of the amount they have.

4. Resources Required.

Frontend – HTML5, CSS3, Java script

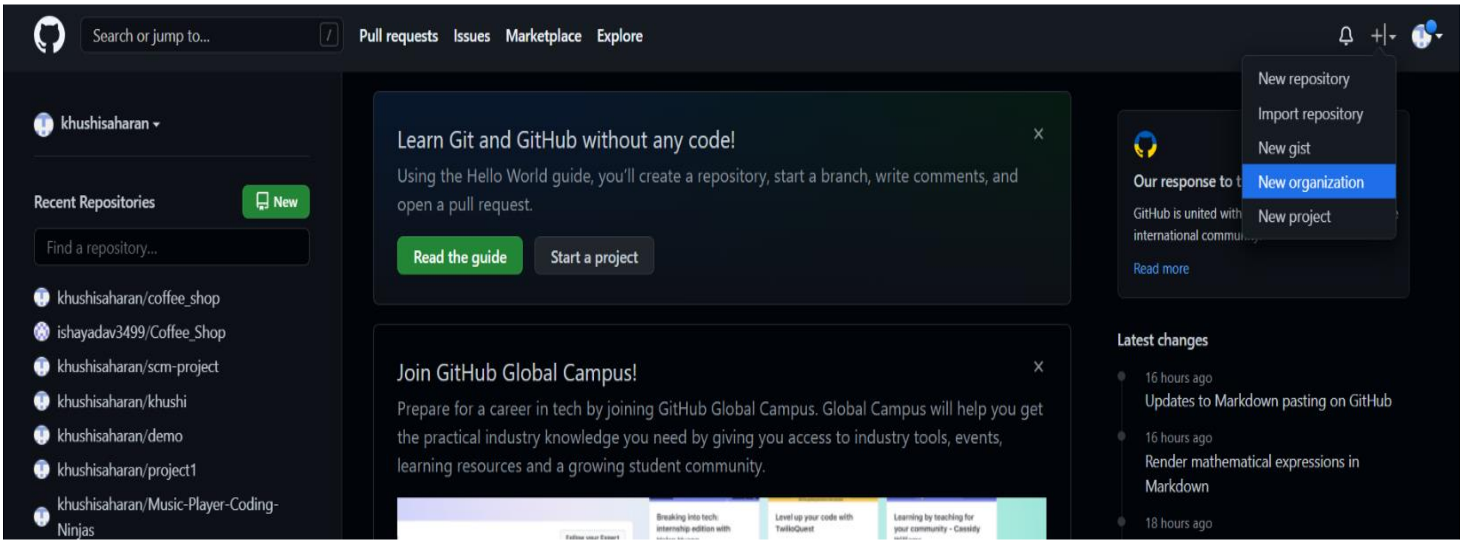
Backend – NodeJS



5. Concepts , Commands, Workflow and Discussions.

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

- Login to your GitHub account and you will land on the homepage as shown below. Click on the button shown in the menu bar and then click on New Organization.



Learn Git and GitHub without any code!

Using the Hello World guide, you'll create a repository, start a branch, write comments, and open a pull request.

Join GitHub Global Campus!

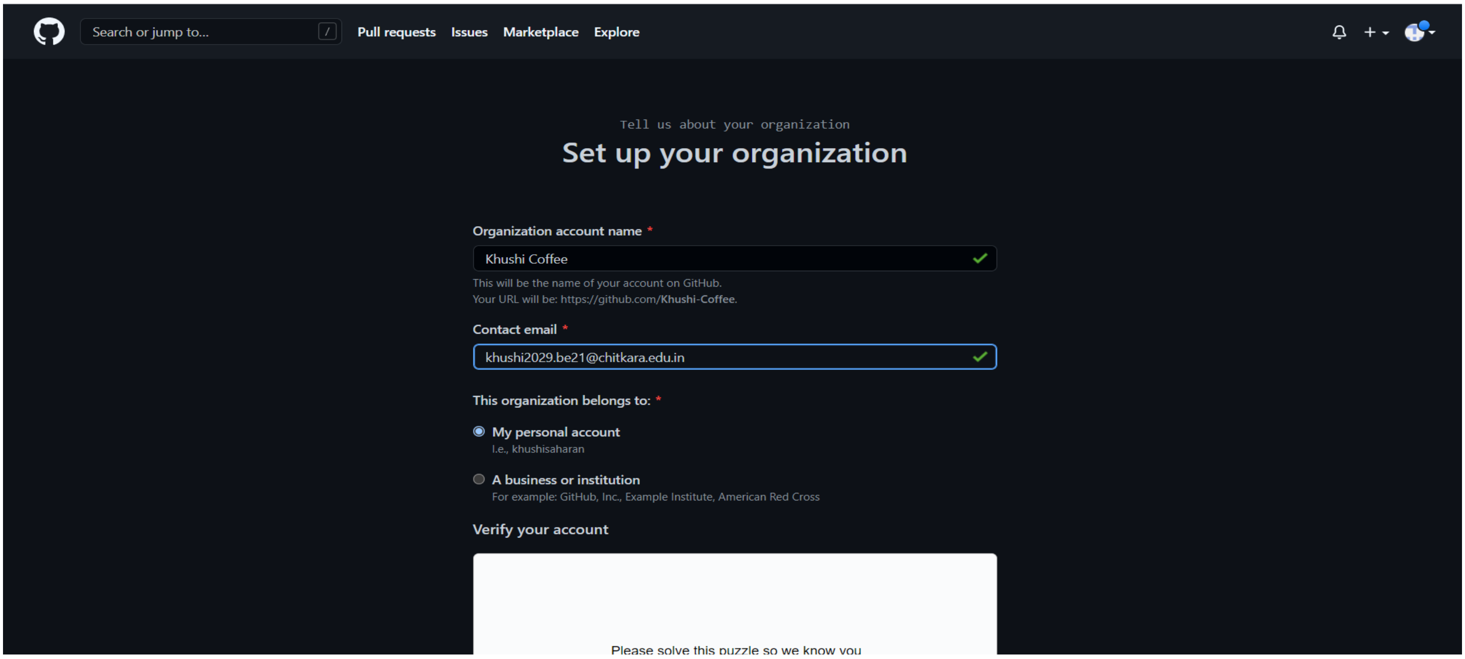
Prepare for a career in tech by joining GitHub Global Campus. Global Campus will help you get the practical industry knowledge you need by giving you access to industry tools, events, learning resources and a growing student community.

GitHub is united with international commu...

Updates to Markdown pasting on GitHub

Render mathematical expressions in Markdown

- Set Up Your Organization. Fill Your Organization’s Name and Other Details.



Tell us about your organization

Set up your organization

Organization account name *

Khushi Coffee ✓

This will be the name of your account on GitHub.
Your URL will be: https://github.com/Khushi-Coffee.

Contact email *

khushi2029.be21@chitkara.edu.in ✓

This organization belongs to: *

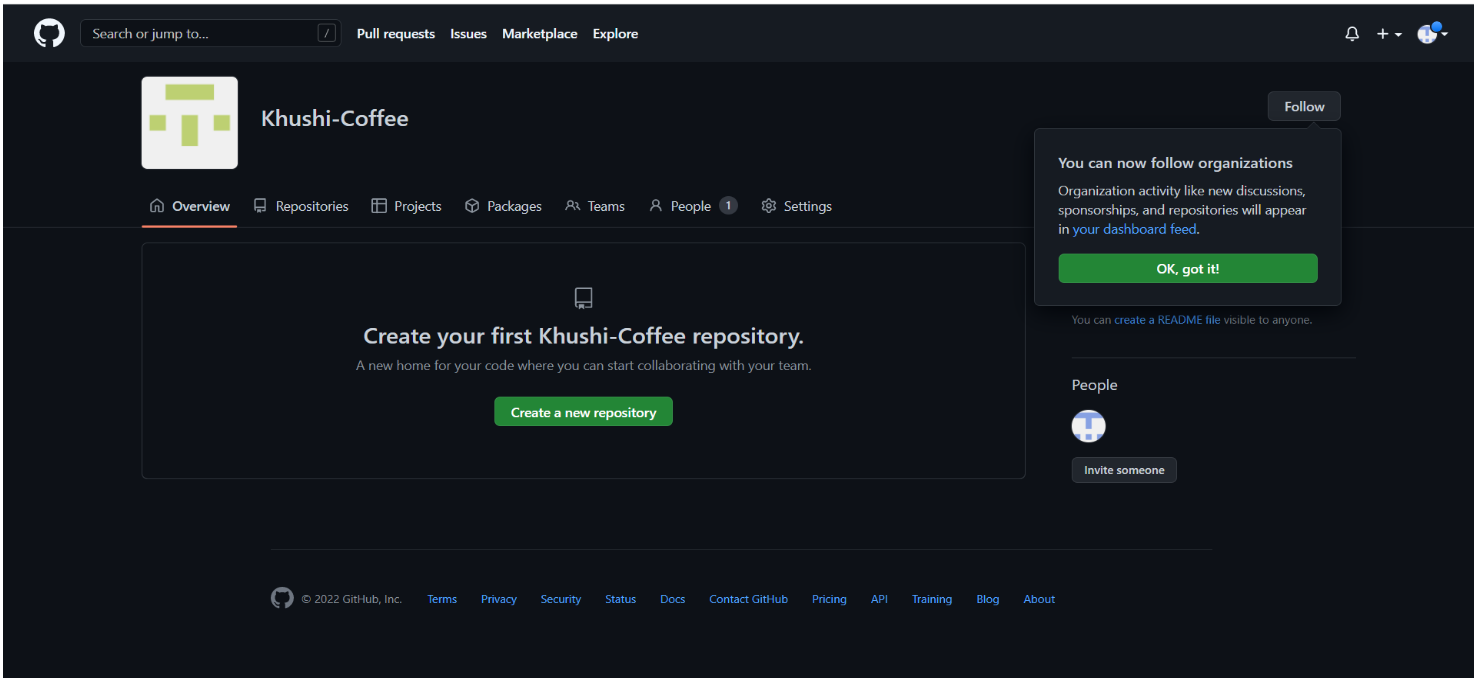
● My personal account
I.e., khushisaharan

○ A business or institution
For example: GitHub, Inc., Example Institute, American Red Cross

Verify your account

Please solve this puzzle so we know you

- After creating the repository, we have to create a Repository.



Khushi-Coffee

Create your first Khushi-Coffee repository.

A new home for your code where you can start collaborating with your team.

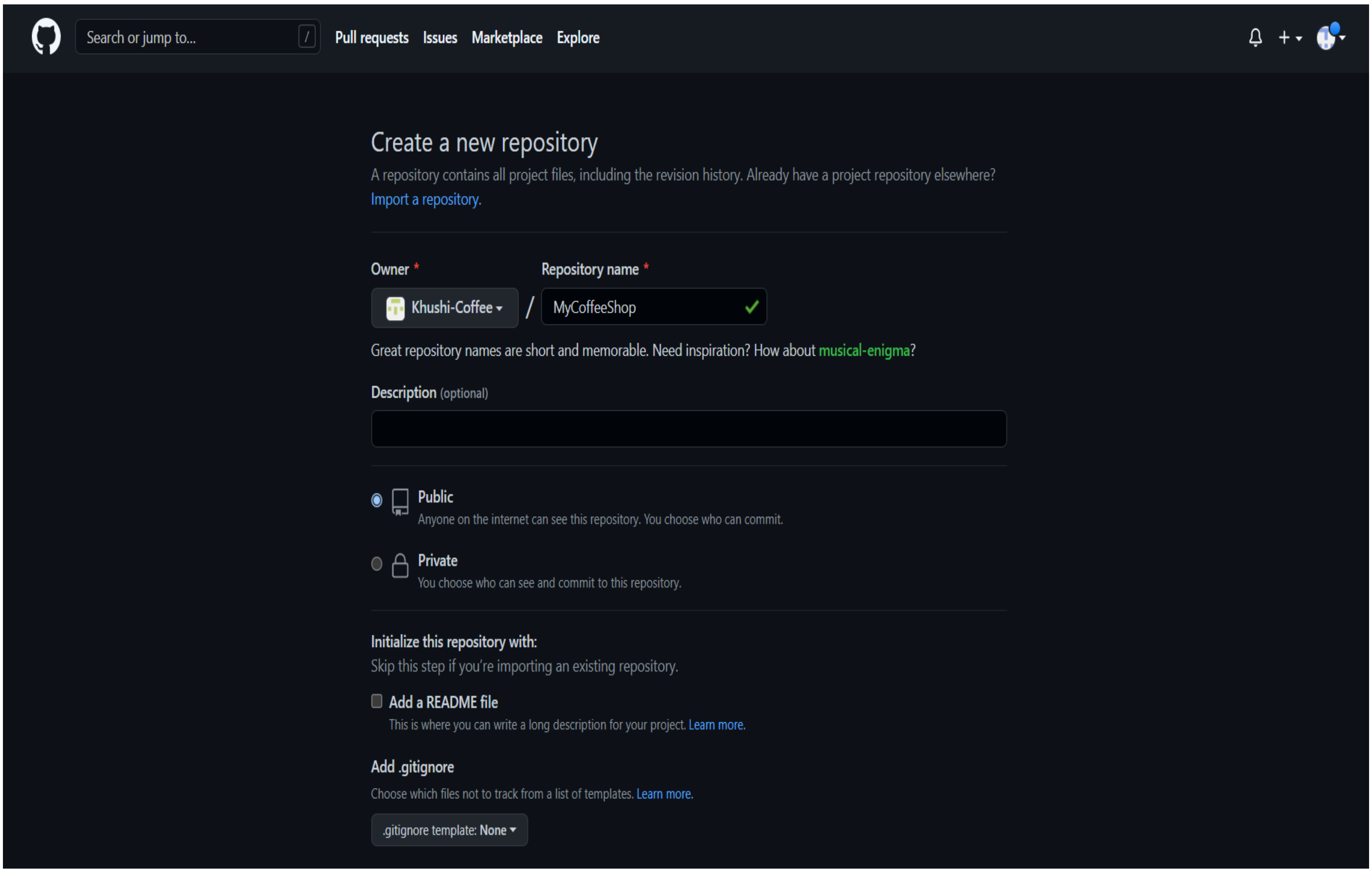
Create a new repository

You can now follow organizations

Organization activity like new discussions, sponsorships, and repositories will appear in your dashboard feed.

OK, got it!

You can create a README file visible to anyone.



Create a new repository

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

Owner * Khushi-Coffee / Repository name * MyCoffeeShop ✓

Great repository names are short and memorable. Need inspiration? How about musical-enigma?

Description (optional)

● 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.

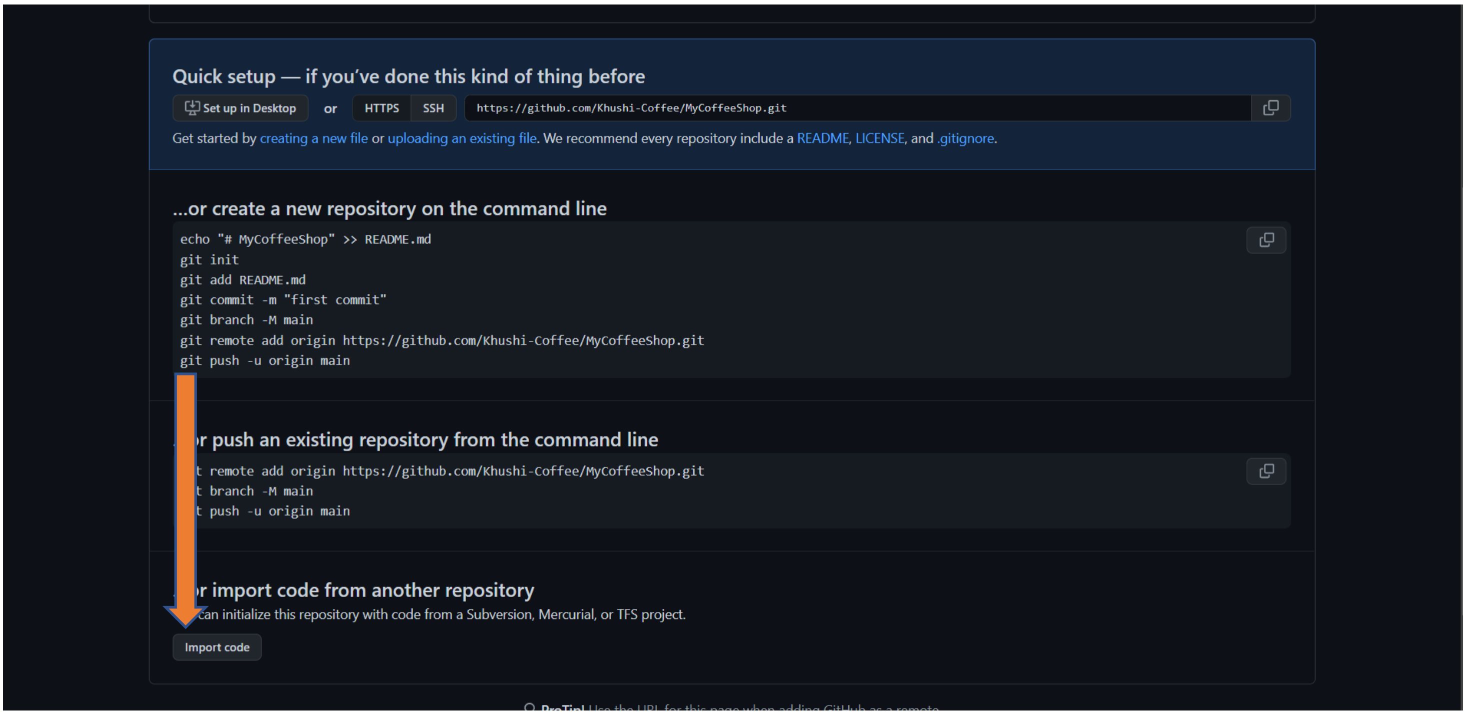
Initialize this repository with:
Skip this step if you're importing an existing repository.

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

Add .gitignore
Choose which files not to track from a list of templates. Learn more.

.gitignore template: None

- If you want to import code from an existing repository select the import code option.



Quick setup — if you've done this kind of thing before

https://github.com/Khushi-Coffee/MyCoffeeShop.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# MyCoffeeShop" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Khushi-Coffee/MyCoffeeShop.git
git push -u origin main
```

r push an existing repository from the command line

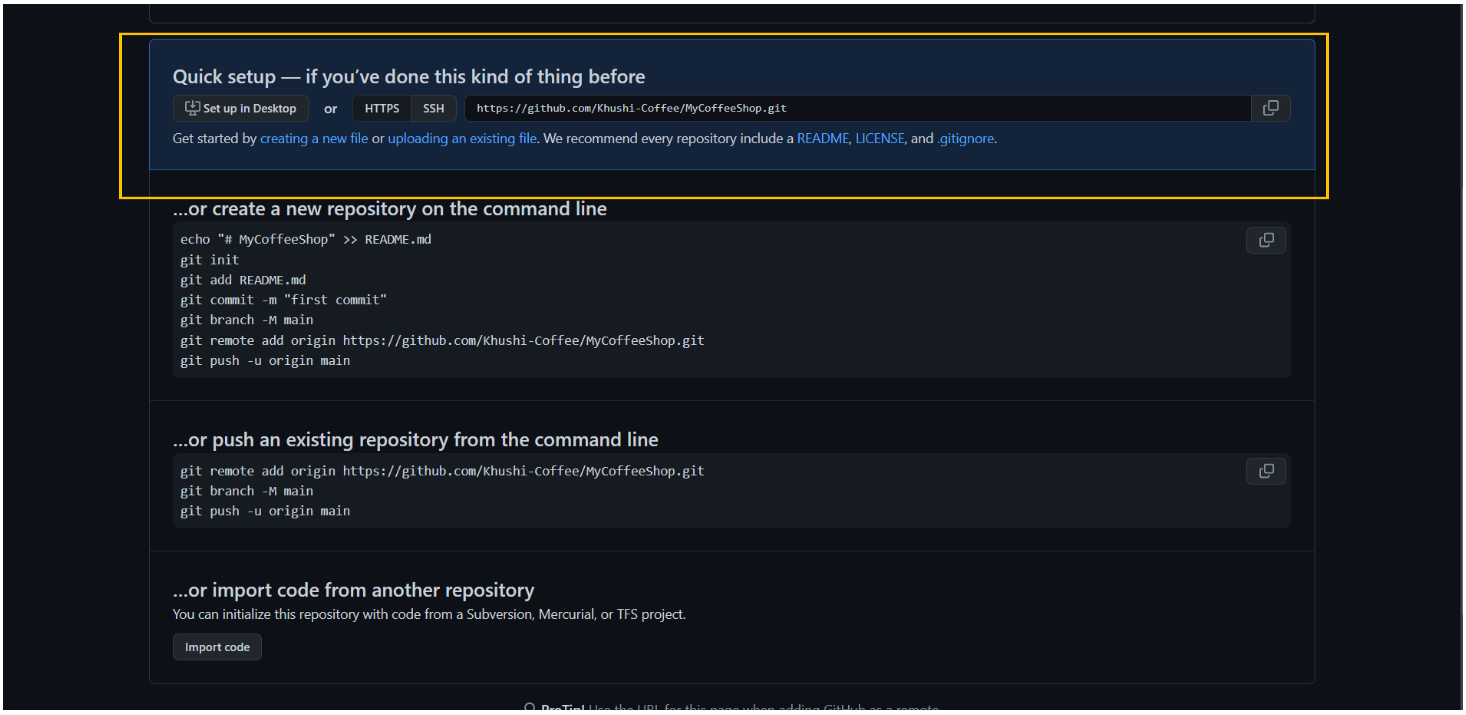
```
t remote add origin https://github.com/Khushi-Coffee/MyCoffeeShop.git
t branch -M main
t push -u origin main
```

r import code from another repository

can initialize this repository with code from a Subversion, Mercurial, or TFS project.

ProTip! Use the URL for this page when adding GitHub as a remote.

- To create a new file or upload an existing file into your repository select the option in the following box.



Quick setup — if you've done this kind of thing before

https://github.com/Khushi-Coffee/MyCoffeeShop.git

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

...or create a new repository on the command line

```
echo "# MyCoffeeShop" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/Khushi-Coffee/MyCoffeeShop.git
git push -u origin main
```

...or push an existing repository from the command line

```
git remote add origin https://github.com/Khushi-Coffee/MyCoffeeShop.git
git branch -M main
git push -u origin main
```

...or import code from another repository

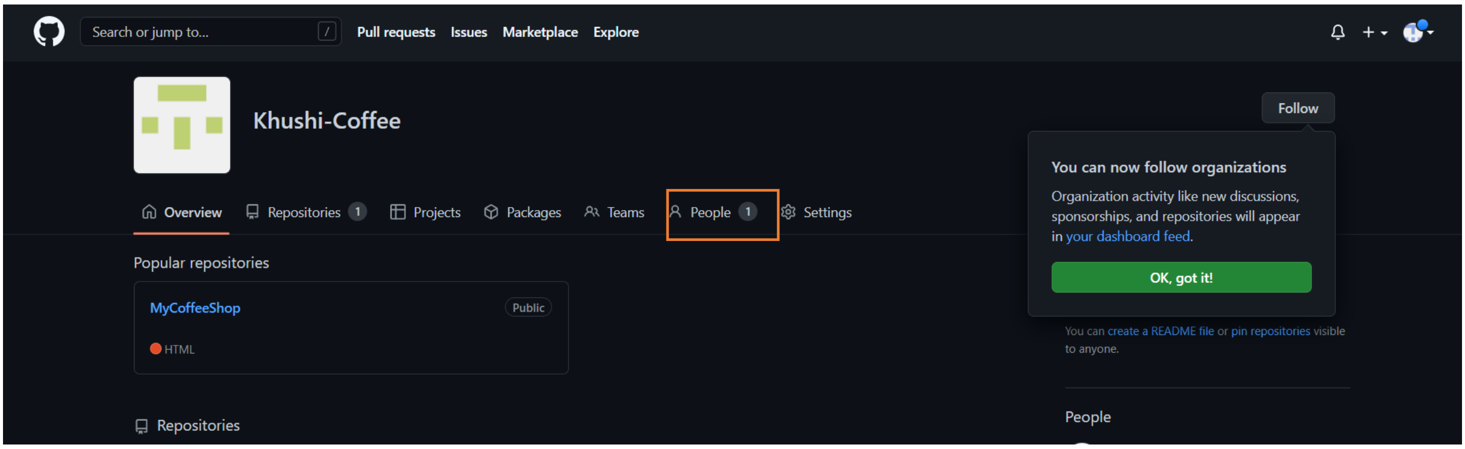
You can initialize this repository with code from a Subversion, Mercurial, or TFS project.

ProTip! Use the URL for this page when adding GitHub as a remote.

- Now, you have created your repository successfully.

- To add members to your repository, open your Organization and select People option in the navigation bar.

- Click on Collaborators option under the access tab.

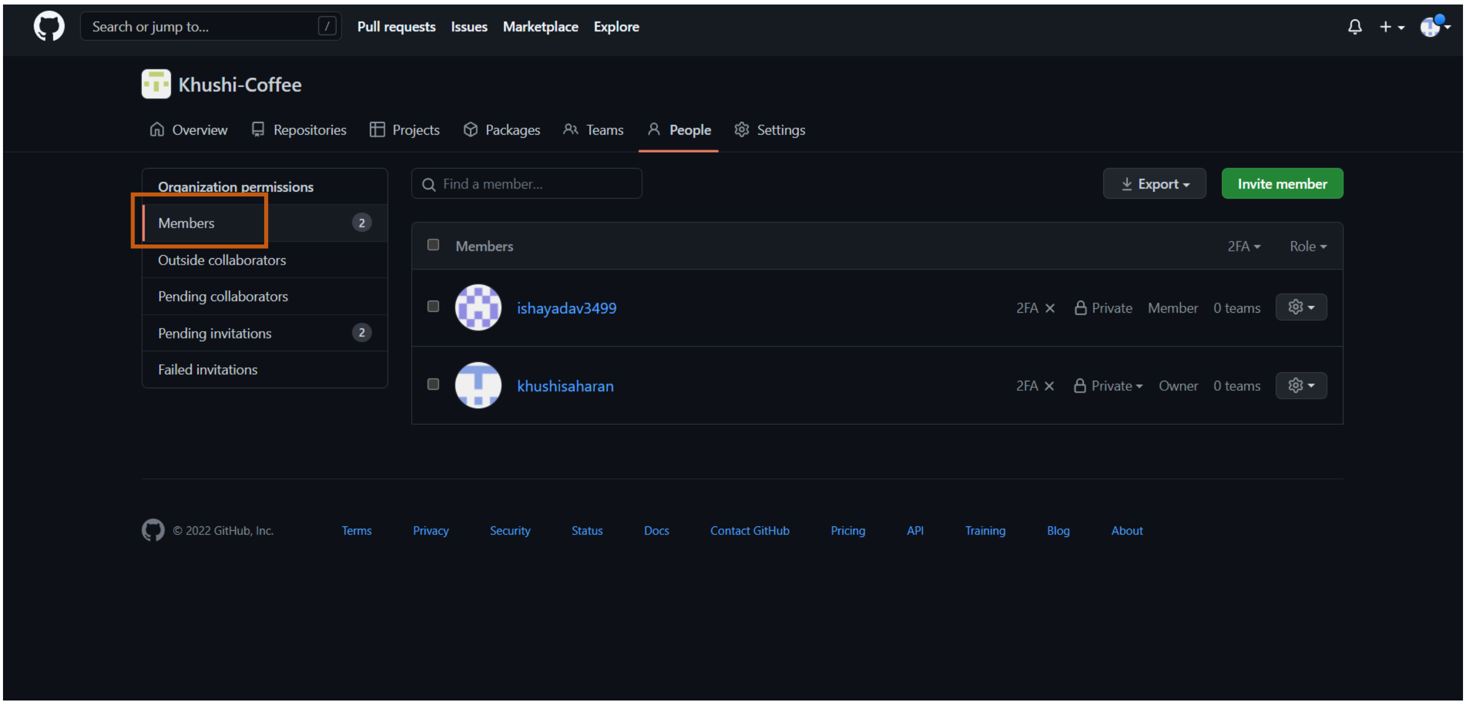


You can now follow organizations

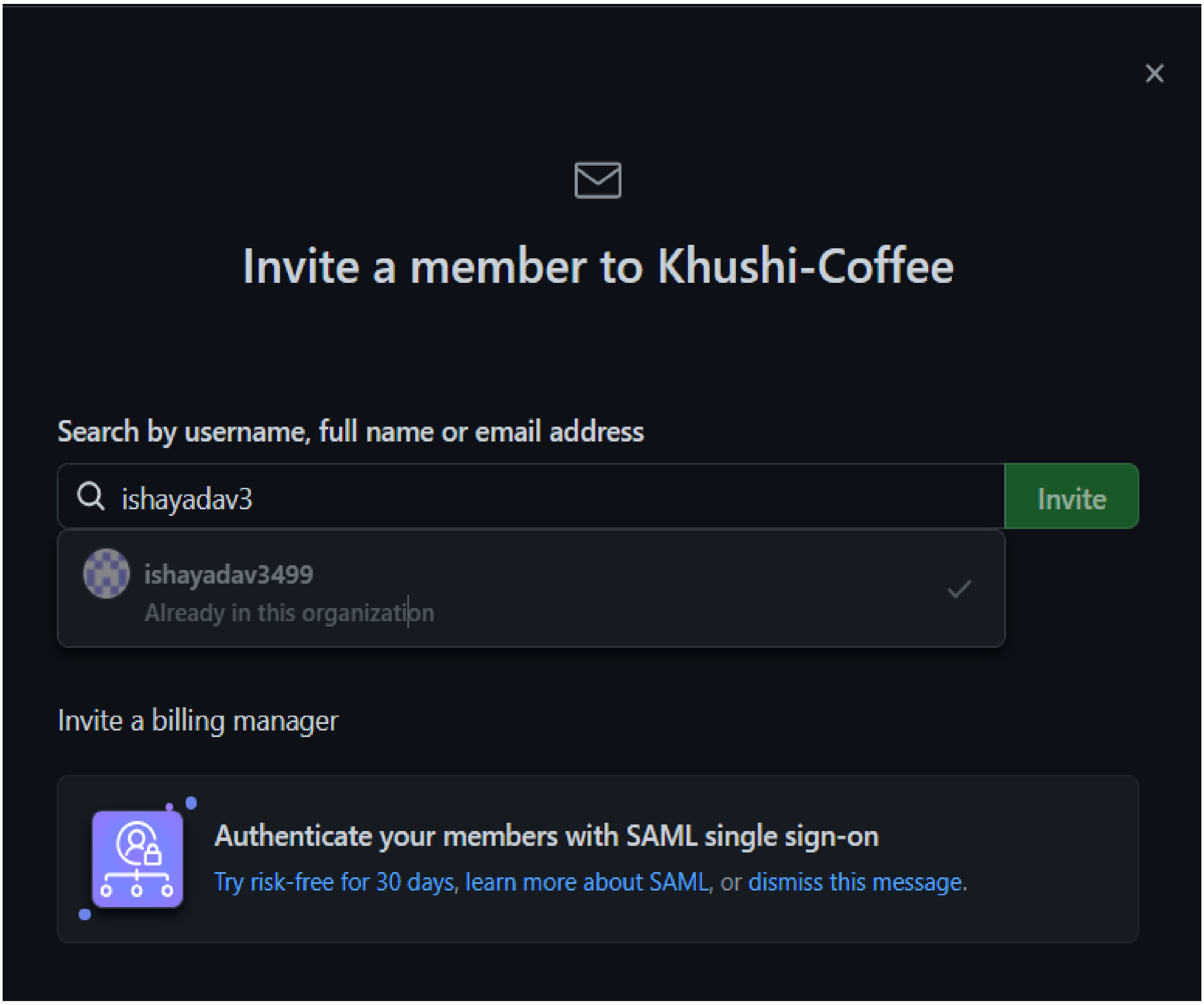
Organization activity like new discussions, sponsorships, and repositories will appear in your dashboard feed.

You can create a README file or pin repositories visible to anyone.

- To add members click on the add people option and search the id of your respective team member.



Members	2FA			Role	
ishayadav3499	2FA ✕	Private	Member	0 teams	
khushisaharan	2FA ✕	Private	Owner	0 teams	



Invite a member to Khushi-Coffee

Search by username, full name or email address

ishayadav3

ishayadav3499

Already in this organization

Invite a billing manager

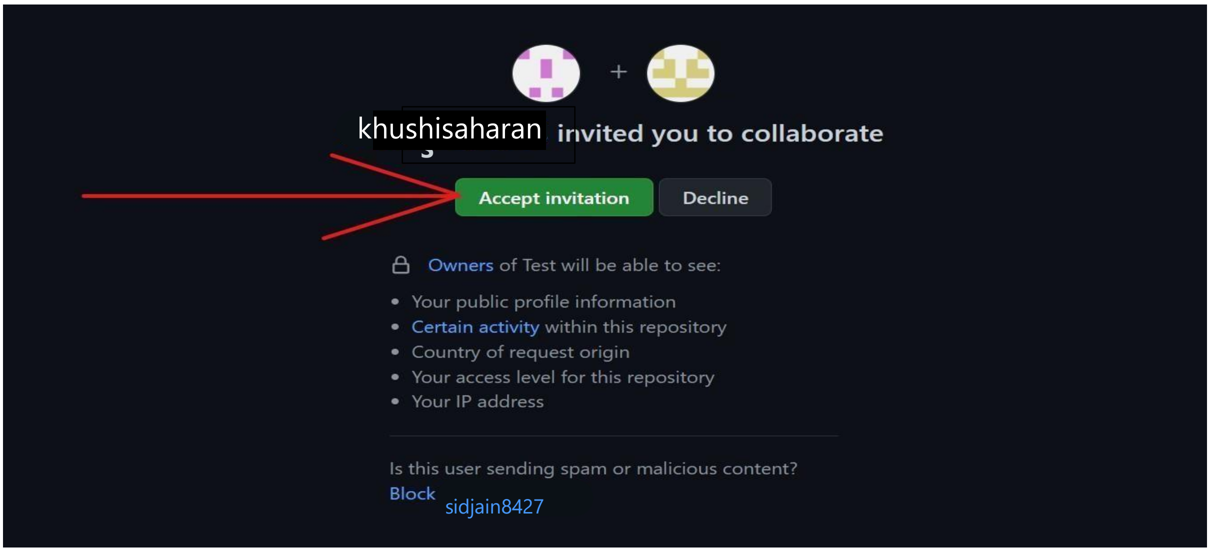
Authenticate your members with SAML single sign-on

Try risk-free for 30 days, learn more about SAML, or dismiss this message.

- To accept the invitation from your team member, open your email registered with GitHub.

- You will receive an invitation mail from the repository owner. Open the email and click on accept invitation.

- You will be redirected to GitHub where you can either select to accept or decline the invitation.



khushisaharan invited you to collaborate

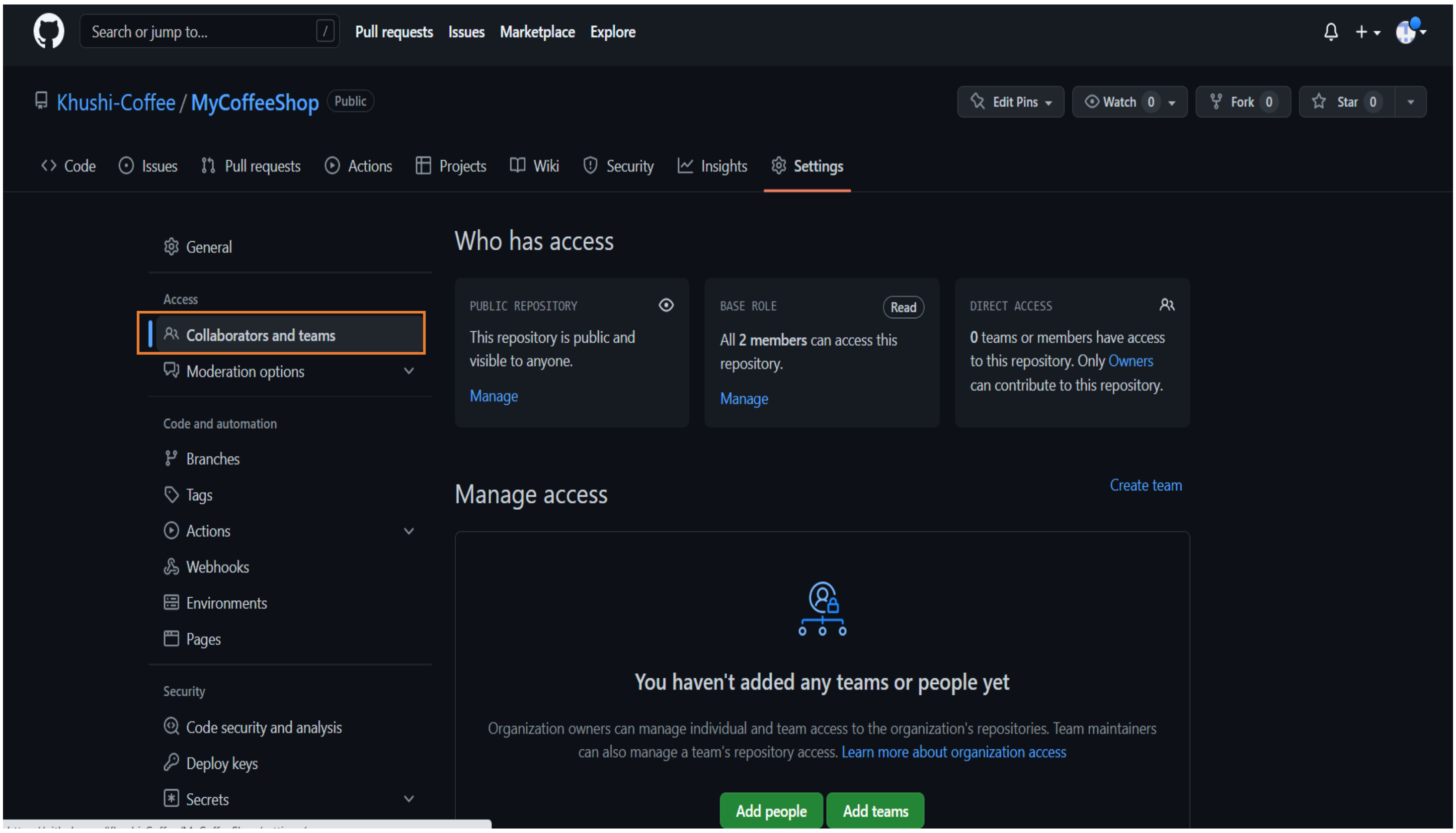
Owners of Test will be able to see:

- Your public profile information
- Certain activity within this repository
- Country of request origin
- Your access level for this repository
- Your IP address

Is this user sending spam or malicious content?

Block sidjain8427

- Next, Open the desired Repository in the Organisation. Look and click on Settings -> Collaborators and Teams. Here you can Manage the role of each collaborator.



Who has access

PUBLIC REPOSITORY: This repository is public and visible to anyone. Manage

BASE ROLE (Read): All 2 members can access this repository. Manage

DIRECT ACCESS: 0 teams or members have access to this repository. Only Owners can contribute to this repository.

Manage access

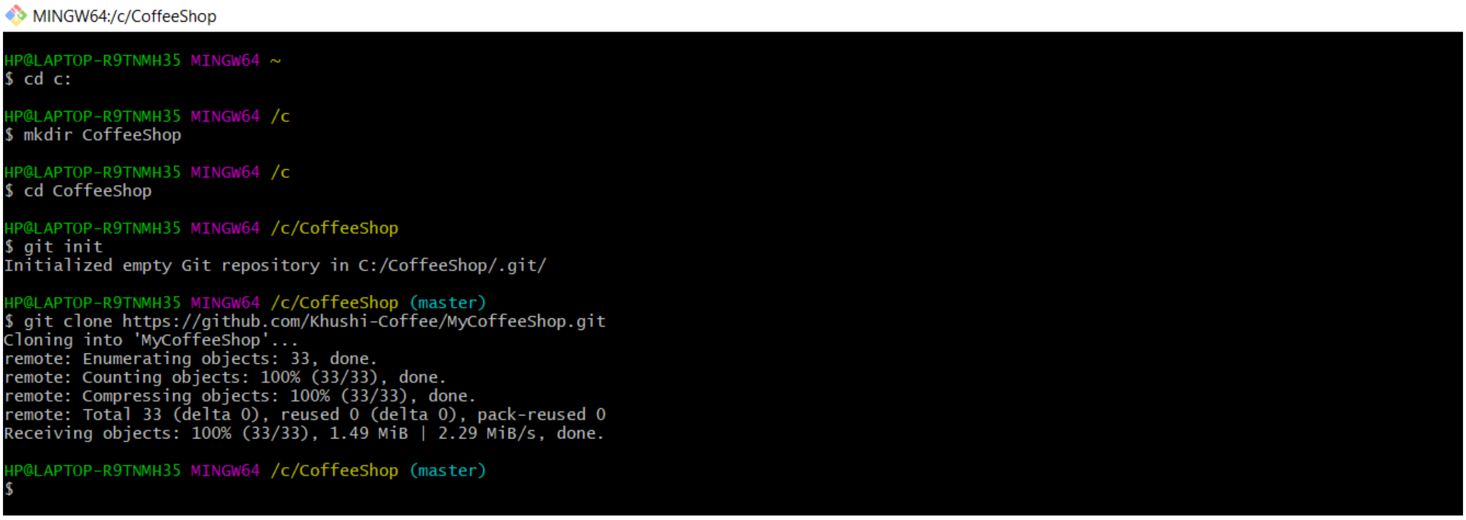
You haven't added any teams or people yet

Organization owners can manage individual and team access to the organization's repositories. Team maintainers can also manage a team's repository access. Learn more about organization access

Experiment No. 02

Aim: Open and Close a Pull Request

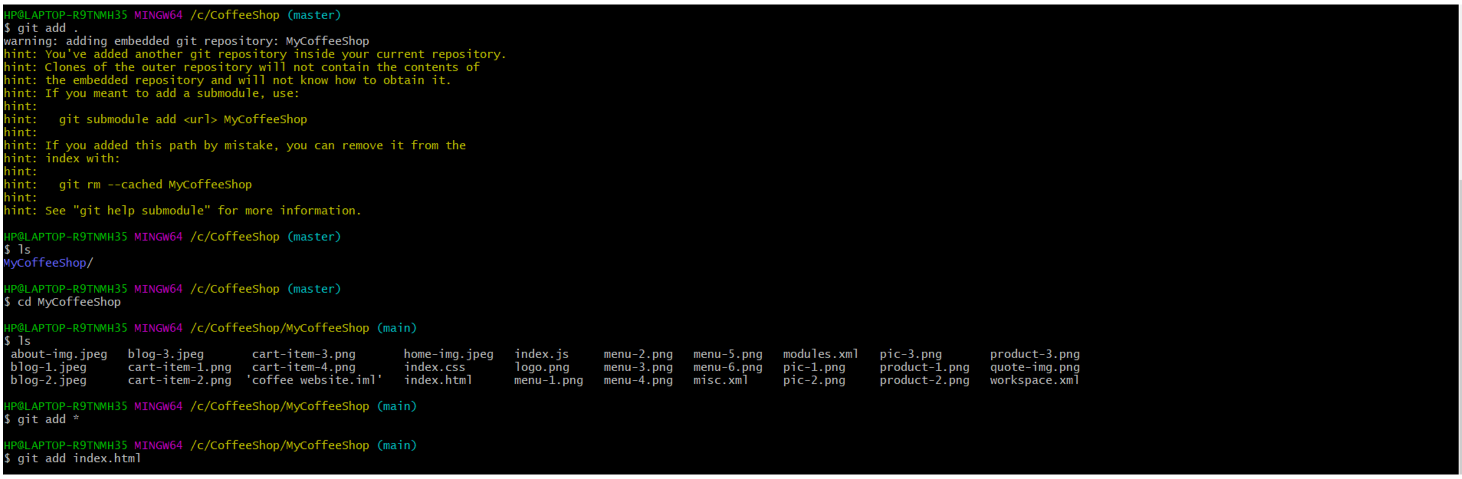
- To Open a Pull Request, First of All, it will be required to fork the repository and commit changes into your own.



MINGW64:/c/CoffeeShop

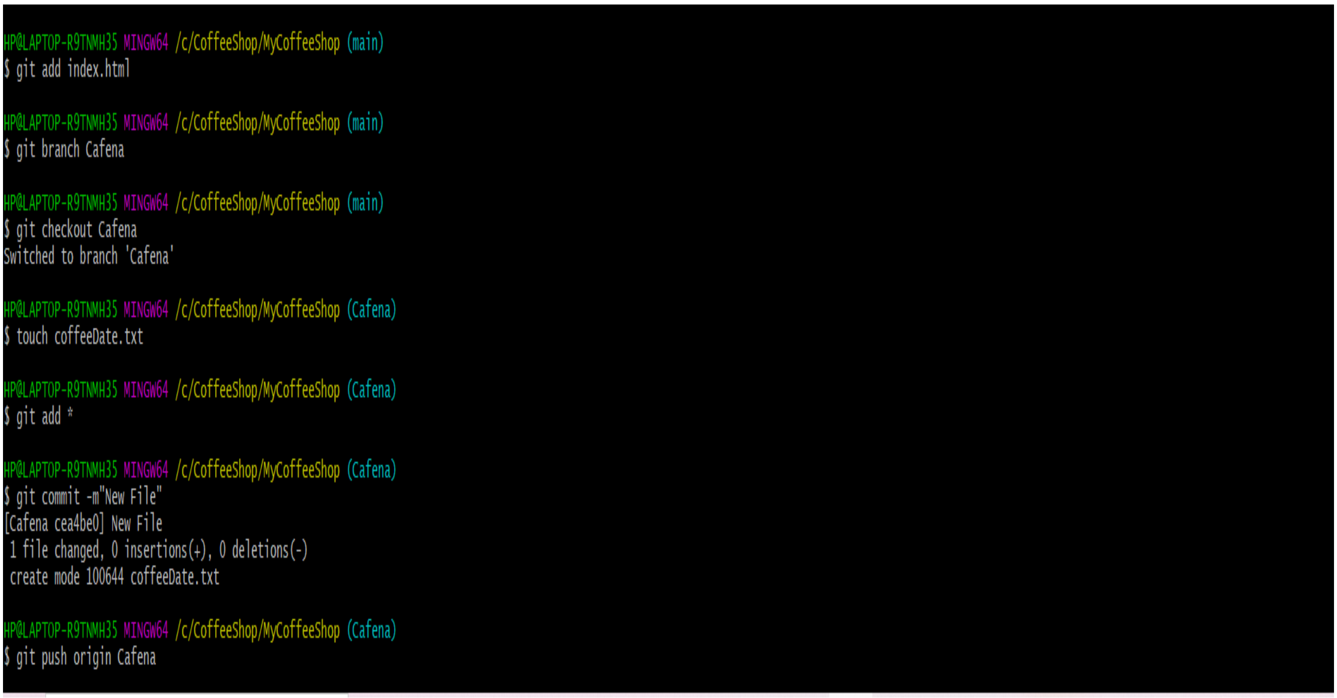
```
HP@LAPTOP-R9TNMH35 MINGW64 ~
$ cd c:

HP@LAPTOP-R9TNMH35 MINGW64 /c
$ mkdir CoffeeShop

HP@LAPTOP-R9TNMH35 MINGW64 /c
$ cd CoffeeShop

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop
$ git init
Initialized empty Git repository in C:/CoffeeShop/.git/

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop (master)
$ git clone https://github.com/Khushi-Coffee/MyCoffeeShop.git
Cloning into 'MyCoffeeShop'...
remote: Enumerating objects: 33, done.
remote: Counting objects: 100% (33/33), done.
remote: Compressing objects: 100% (33/33), done.
remote: Total 33 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (33/33), 1.49 MiB | 2.29 MiB/s, done.

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop (master)
$
```

- Add and commit the changes to the local repository.

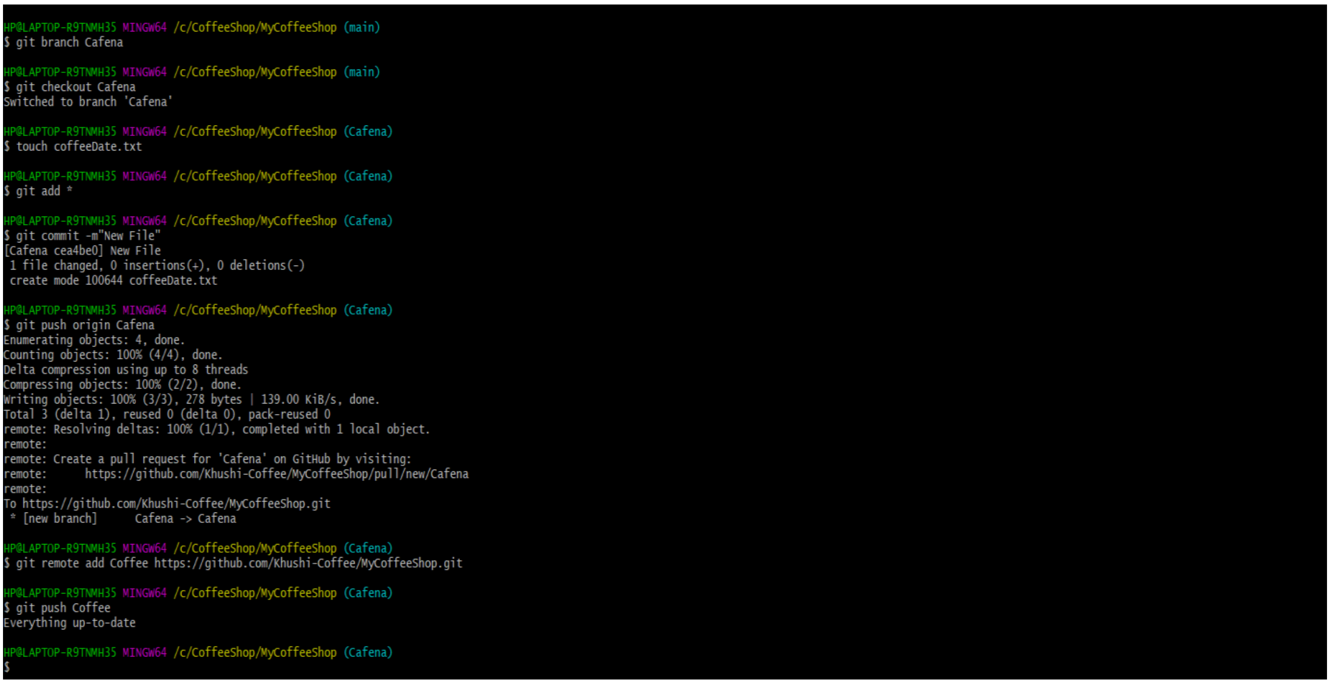


```
HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop (master)
$ git add .
warning: adding embedded git repository: MyCoffeeShop
hint: You've added another git repository inside your current repository.
hint: Clones of the outer repository will not contain the contents of
hint: the embedded repository and will not know how to obtain it.
hint: If you meant to add a submodule, use:
hint:
hint:   git submodule add <url> MyCoffeeShop
hint:
hint: If you added this path by mistake, you can remove it from the
hint: index with:
hint:
hint:   git rm --cached MyCoffeeShop
hint:
hint: See "git help submodule" for more information.

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop (master)
$ ls
MyCoffeeShop/

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop (master)
$ cd MyCoffeeShop

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (main)
$ ls
 about-img.jpeg   blog-3.jpeg       cart-item-3.png        home-img.jpeg   index.js     menu-2.png   menu-5.png   modules.xml   pic-3.png       product-3.png
 blog-1.jpeg      cart-item-1.png   cart-item-4.png        index.css       logo.png     menu-3.png   menu-6.png   pic-1.png     product-1.png   quote-img.png
 blog-2.jpeg      cart-item-2.png  'coffee website.iml'    index.html      menu-1.png   menu-4.png   misc.xml     pic-2.png     product-2.png   workspace.xml

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (main)
$ git add *

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (main)
$ git add index.html
```

- Use git push origin branch name option to push the new branch to the main repository.

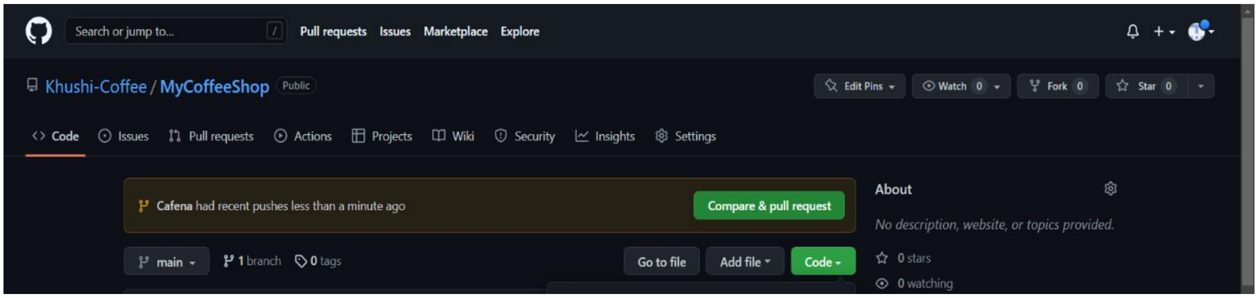


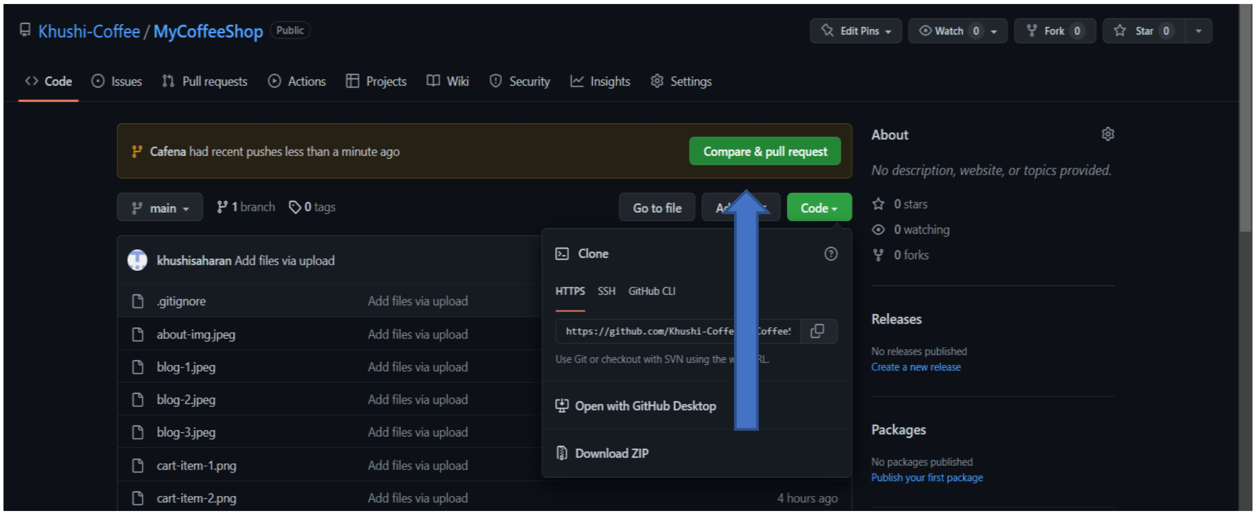
```
HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (main)
$ git add index.html

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (main)
$ git branch Cafena

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (main)
$ git checkout Cafena
Switched to branch 'Cafena'

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (Cafena)
$ touch coffeeDate.txt

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (Cafena)
$ git add *

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (Cafena)
$ git commit -m"New File"
[Cafena cea4be0] New File
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 coffeeDate.txt

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (Cafena)
$ git push origin Cafena
```



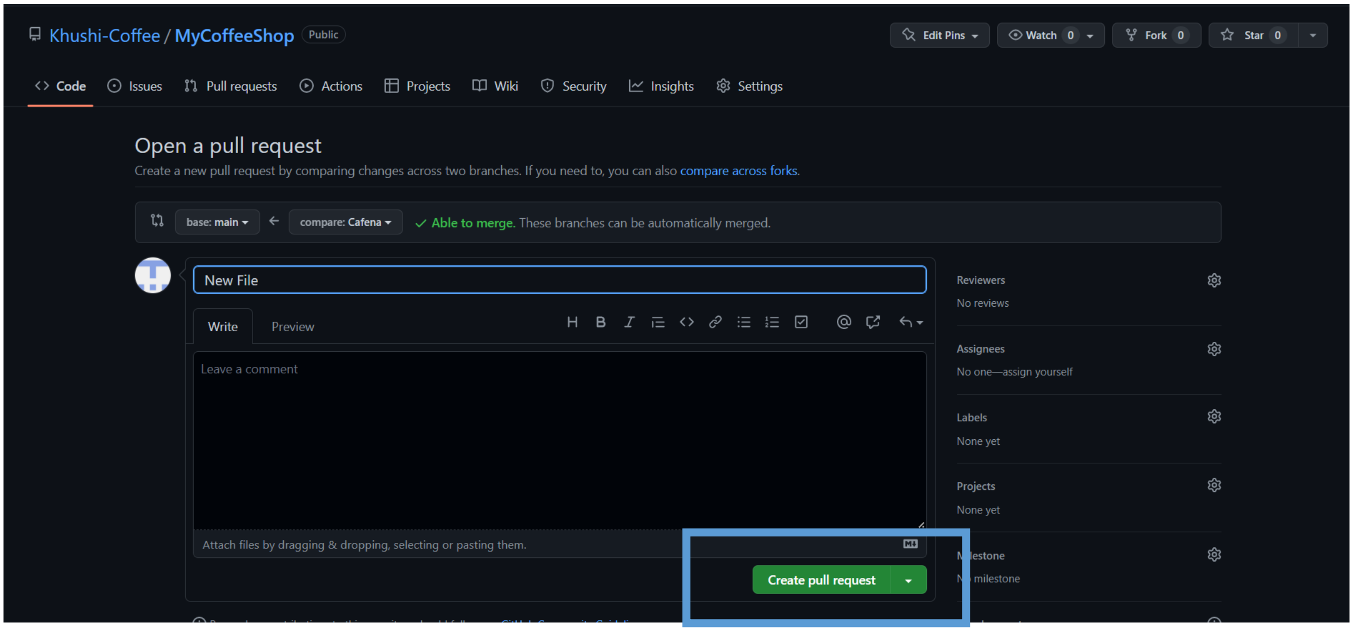
```
HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (main)
$ git branch Cafena

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (main)
$ git checkout Cafena
Switched to branch 'Cafena'

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (Cafena)
$ touch coffeeDate.txt

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (Cafena)
$ git add *

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (Cafena)
$ git commit -m"New File"
[Cafena cea4be0] New File
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 coffeeDate.txt

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (Cafena)
$ git push origin Cafena
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 278 bytes | 139.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
remote:
remote: Create a pull request for 'Cafena' on GitHub by visiting:
remote:      https://github.com/Khushi-Coffee/MyCoffeeShop/pull/new/Cafena
remote:
To https://github.com/Khushi-Coffee/MyCoffeeShop.git
 * [new branch]      Cafena -> Cafena

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (Cafena)
$ git remote add Coffee https://github.com/Khushi-Coffee/MyCoffeeShop.git

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (Cafena)
$ git push Coffee
Everything up-to-date

HP@LAPTOP-R9TNMH35 MINGW64 /c/CoffeeShop/MyCoffeeShop (Cafena)
$
```

- After pushing new branch GitHub will either automatically ask you to create a pull request or you can create your own pull request.

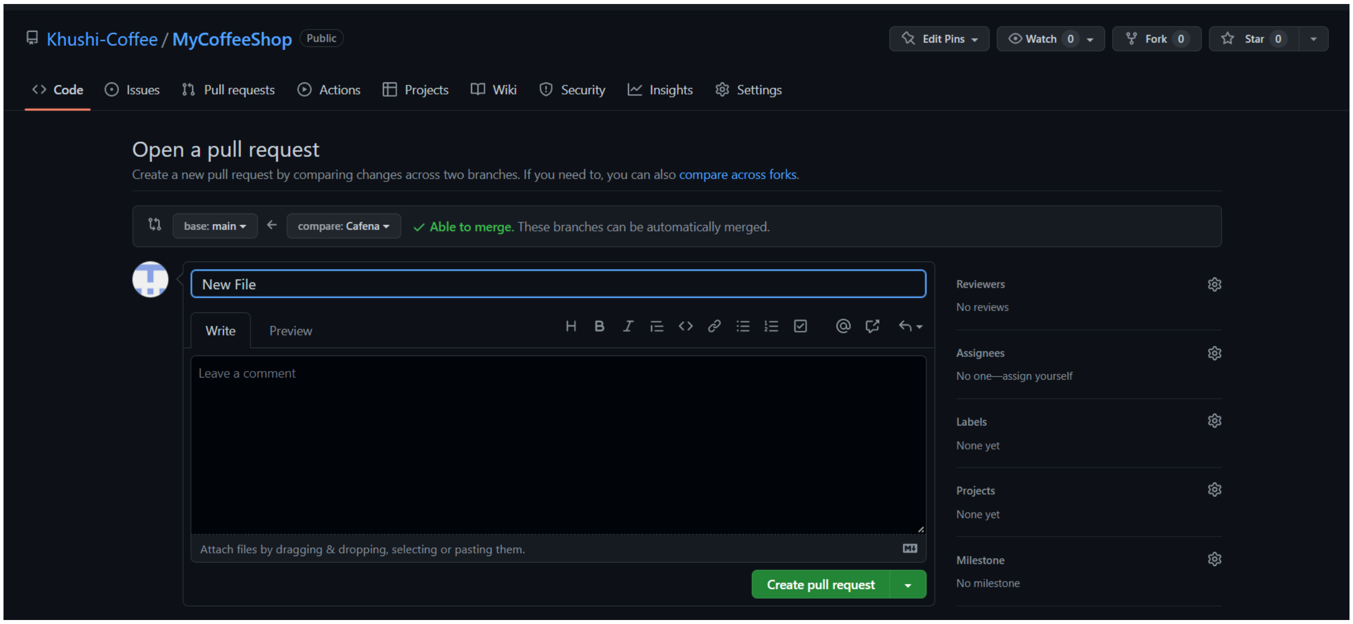




- To create your own pull request click on pull request option.



- GitHub will detect any conflicts and ask you to enter a description of your pull request.



Open a pull request

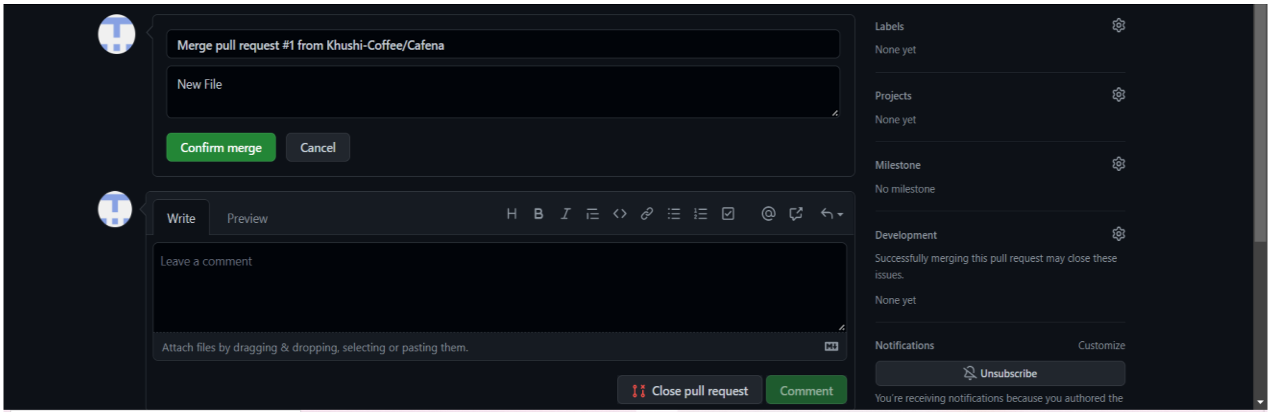
Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks.

✓ Able to merge. These branches can be automatically merged.

New File

Attach files by dragging & dropping, selecting or pasting them.

- After opening a pull request all the team members will be sent the request if they want to merge or close the request.



Merge pull request #1 from Khushi-Coffee/Cafena

New File

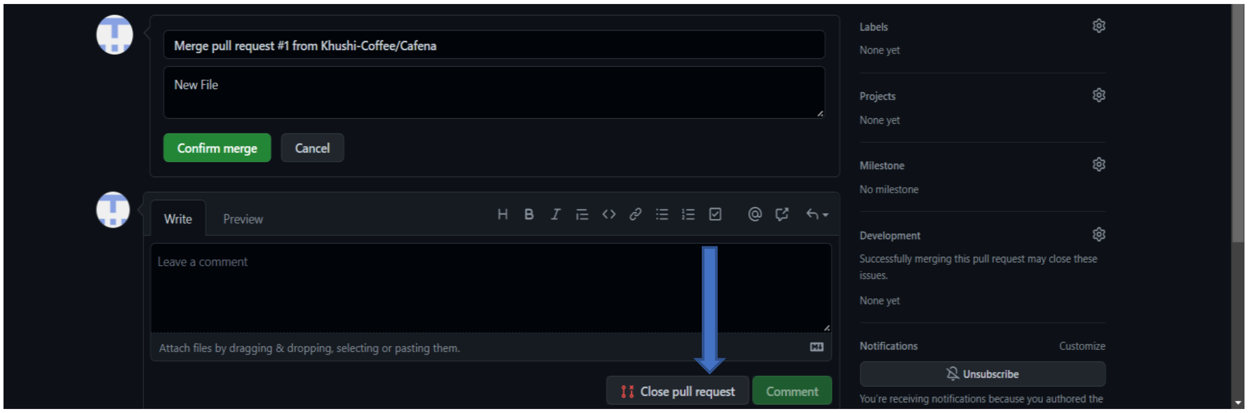
Attach files by dragging & dropping, selecting or pasting them.

Successfully merging this pull request may close these issues.

You're receiving notifications because you authored the

- If the team member chooses not to merge your pull request they will close you're the pull request.

- To close the pull request simply click on close pull request and add comment/ reason why you closed the pull request.



Merge pull request #1 from Khushi-Coffee/Cafena

New File

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Labels
None yet

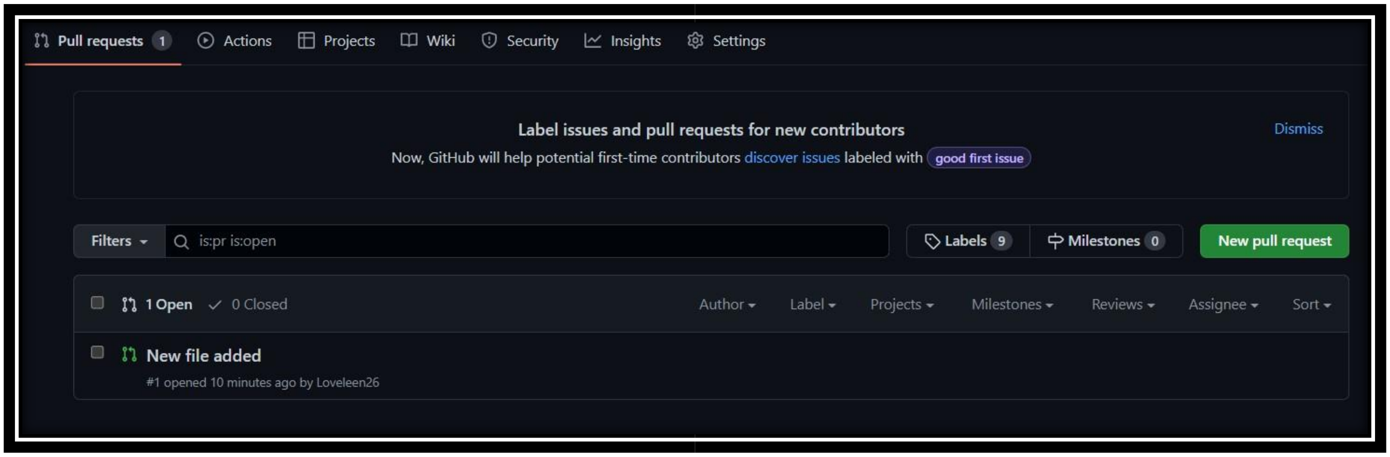
Projects
None yet

Milestone
No milestone

Development
Successfully merging this pull request may close these issues.
None yet

Notifications
You're receiving notifications because you authored the

- You can see all the pull request generated and how they were dealt with by clicking on pull request option.



Label issues and pull requests for new contributors

Now, GitHub will help potential first-time contributors discover issues labeled with good first issue

is:pr is:open

1 Open ✓ 0 Closed

New file added
#1 opened 10 minutes ago by Loveleen26

Experiment No. 03

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

To create a pull request on a team member's repository and close requests by any other team members as a maintainer follow the procedure given below:-

- Do the required changes in the repository, add and commit these changes in the local repository in a new branch.
- Push the modified branch using git push origin branch name.
- Open a pull request by following the procedure from the above experiment.
- The pull request will be created and will be visible to all the team members.
- Ask your team member to login to his/her Git hub account.
- They will notice a new notification in the pull request menu.
- Click on it. The pull request generated by you will be visible to them.

- Click on the pull request. Two option will be available, either to close the pull request or Merge the request with the main branch.

- By selecting the merge branch option the main branch will get updated for all the team members.

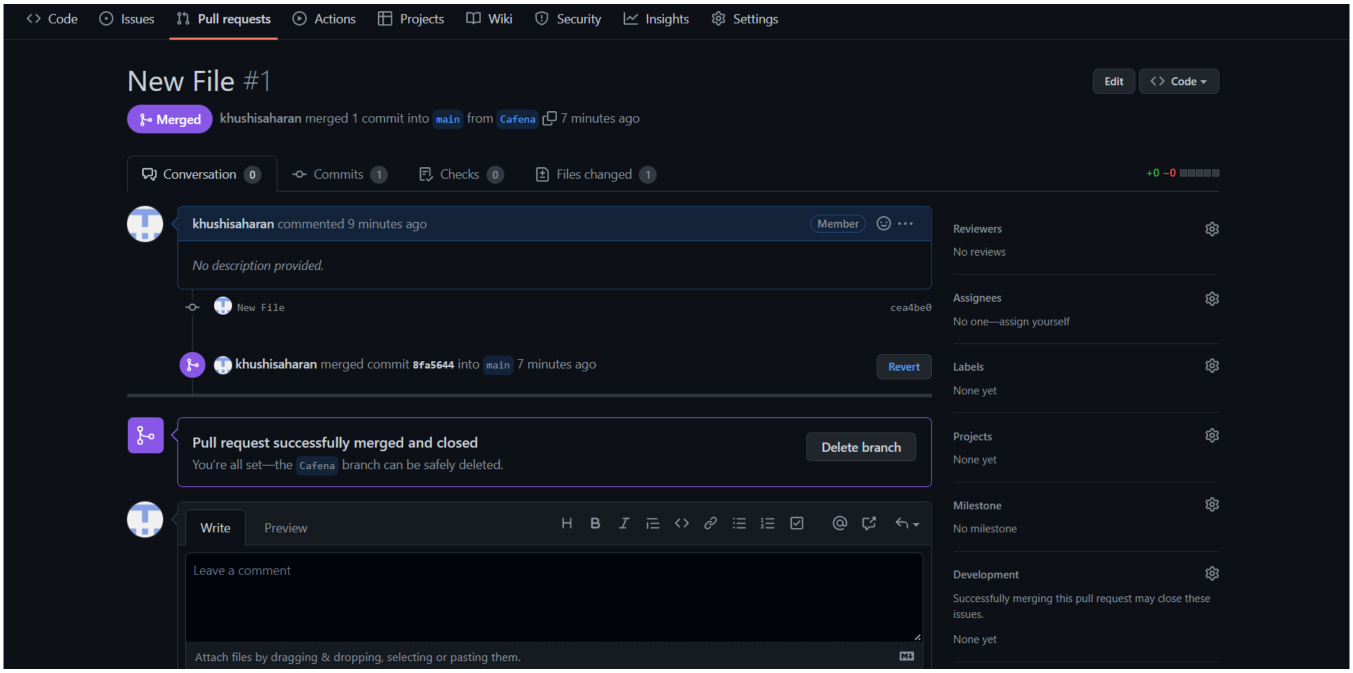
-

- By selecting close the pull request the pull request is not accepted and not merged with main branch.

- The process is similar to closing and merging the pull request by you. It simply includes an external party to execute.

- The result of merging the pull request .

- The result of closing the request is shown below.



New File #1

Merged khushisaharan merged 1 commit into main from Cafena 7 minutes ago

khushisaharan commented 9 minutes ago

No description provided.

New File cea4be0

khushisaharan merged commit 8fa5644 into main 7 minutes ago

Pull request successfully merged and closed

You're all set—the Cafena branch can be safely deleted.

Leave a comment

Attach files by dragging & dropping, selecting or pasting them.

Reviewers: No reviews

Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

Milestone: No milestone

Development: Successfully merging this pull request may close these issues. None yet

Experiment No. 04

Aim: Publish and print network graphs

The network graph is one of the useful features for developers on GitHub. It is used to display the branch history of the entire repository network, including branches of the root repository and branches of forks that contain commits unique to the network.

A repository's graphs give you information on traffic, projects that depend on the repository, contributors and commits to the repository, and a repository's forks and network. If you maintain a repository, you can use this data to get a better understanding of who's using your repository and why they're using it.

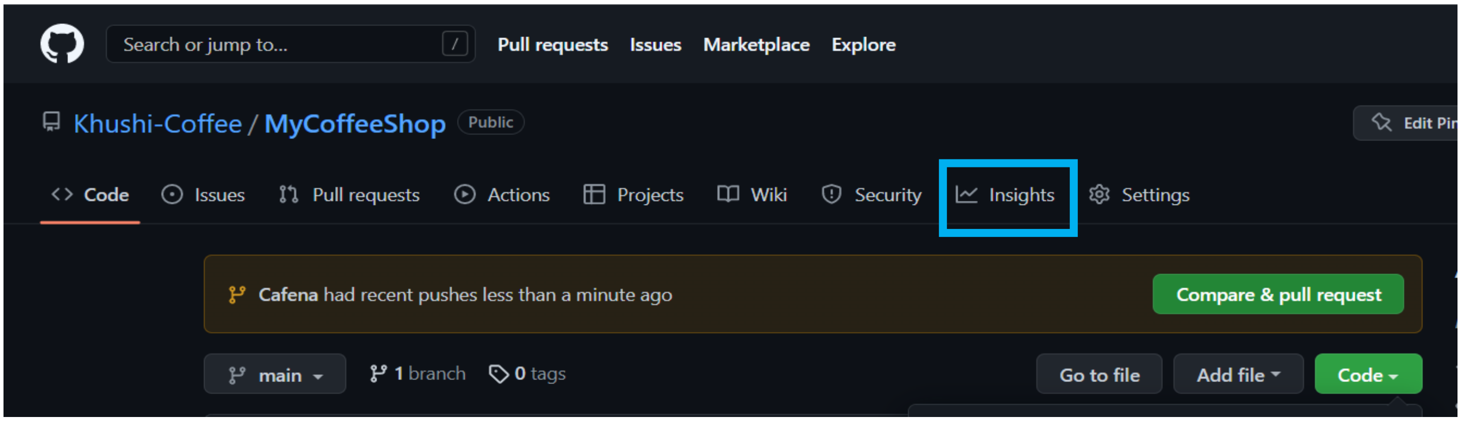
Some repository graphs are available only in public repositories with GitHub Free:

- Pulse
- Contributors
- Traffic
- Commits
- Code frequency
- Network

Steps to access network graphs of respective repository

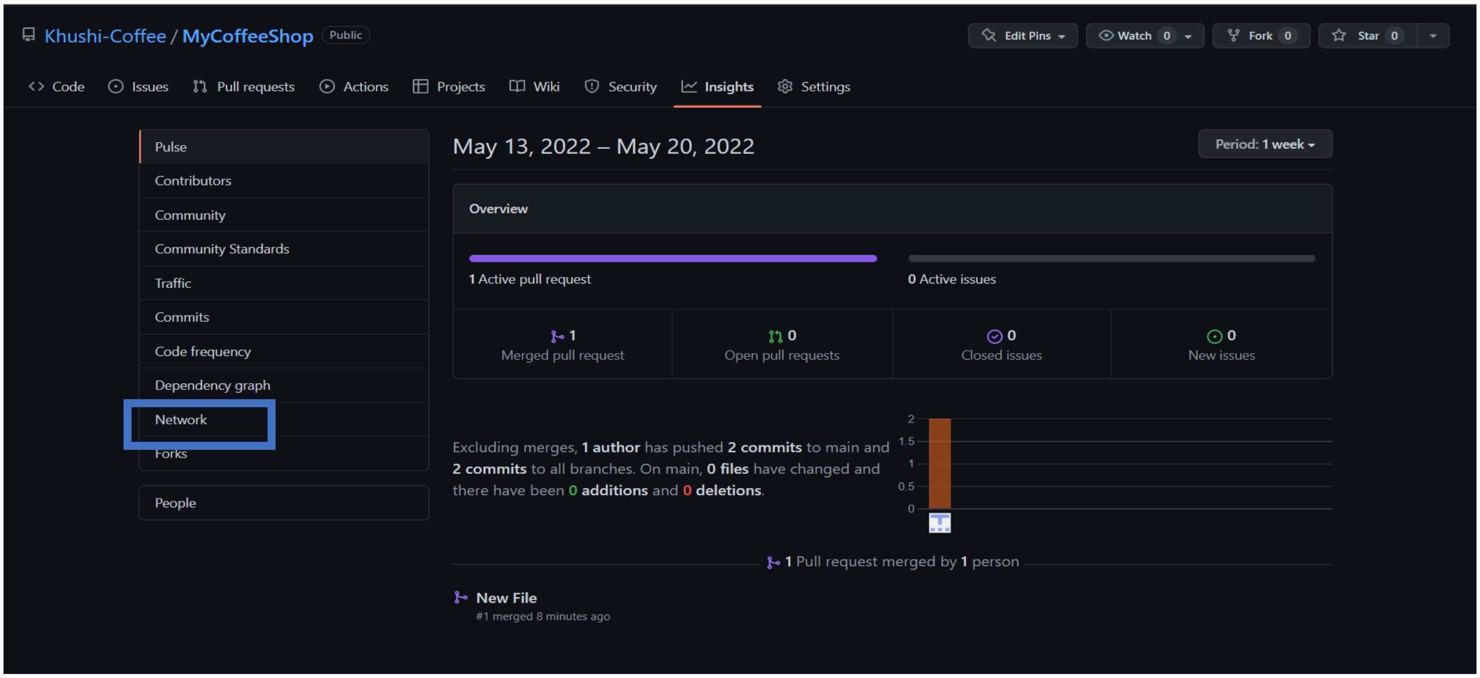
1. On GitHub.com, navigate to the main page of the repository.

2. Under your repository name,click Insights.



Cafena had recent pushes less than a minute ago

3.At the left sidebar, click on Network.



May 13, 2022 – May 20, 2022

Overview

1 Active pull request

0 Active issues

1 Merged pull request

0 Open pull requests

0 Closed issues

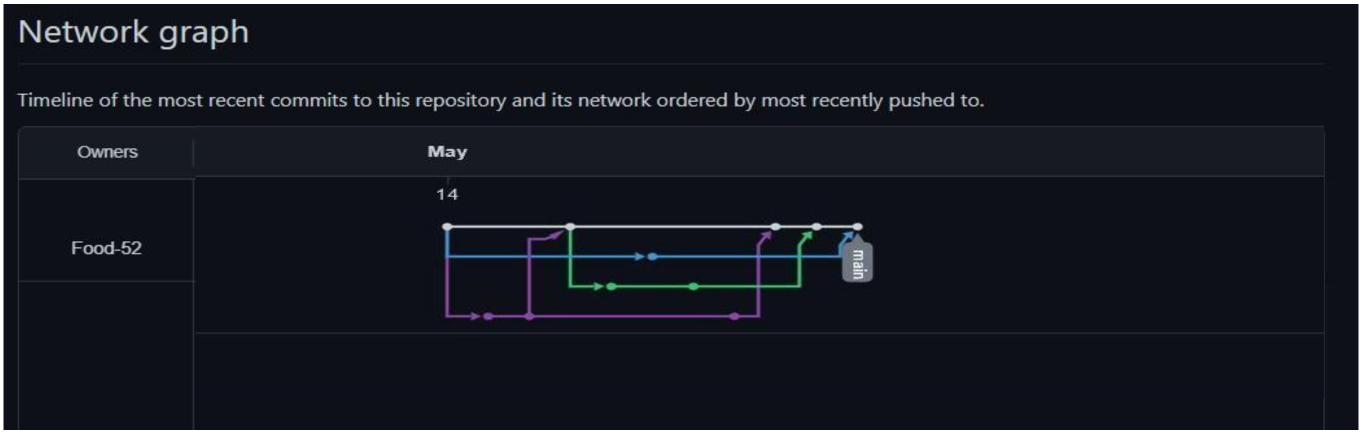
0 New issues

Excluding merges, 1 author has pushed 2 commits to main and 2 commits to all branches. On main, 0 files have changed and there have been 0 additions and 0 deletions.

1 Pull request merged by 1 person

New File
#1 merged 8 minutes ago

You will get the network graph of your repository which displays the branch history of the entire repository network, including branches of the root repository and branches of forks that contain commits unique to the network.



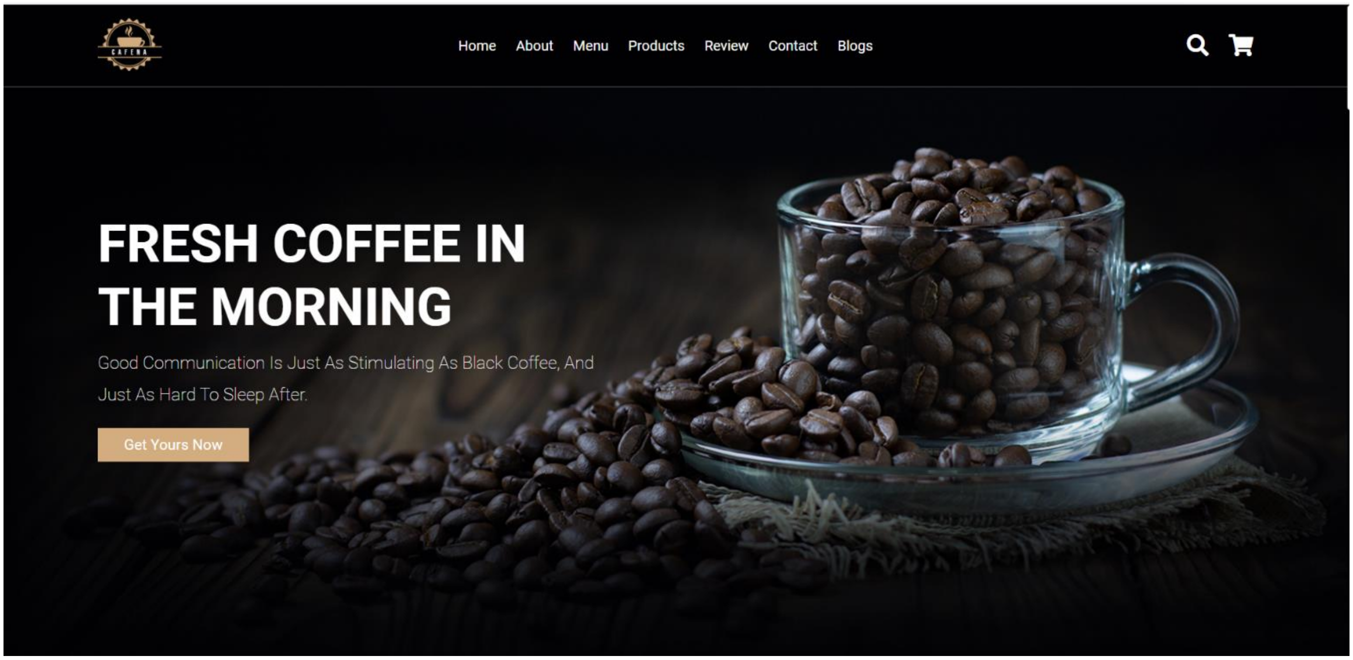
Network graph

Timeline of the most recent commits to this repository and its network ordered by most recently pushed to.

5. Workflow and Discussion

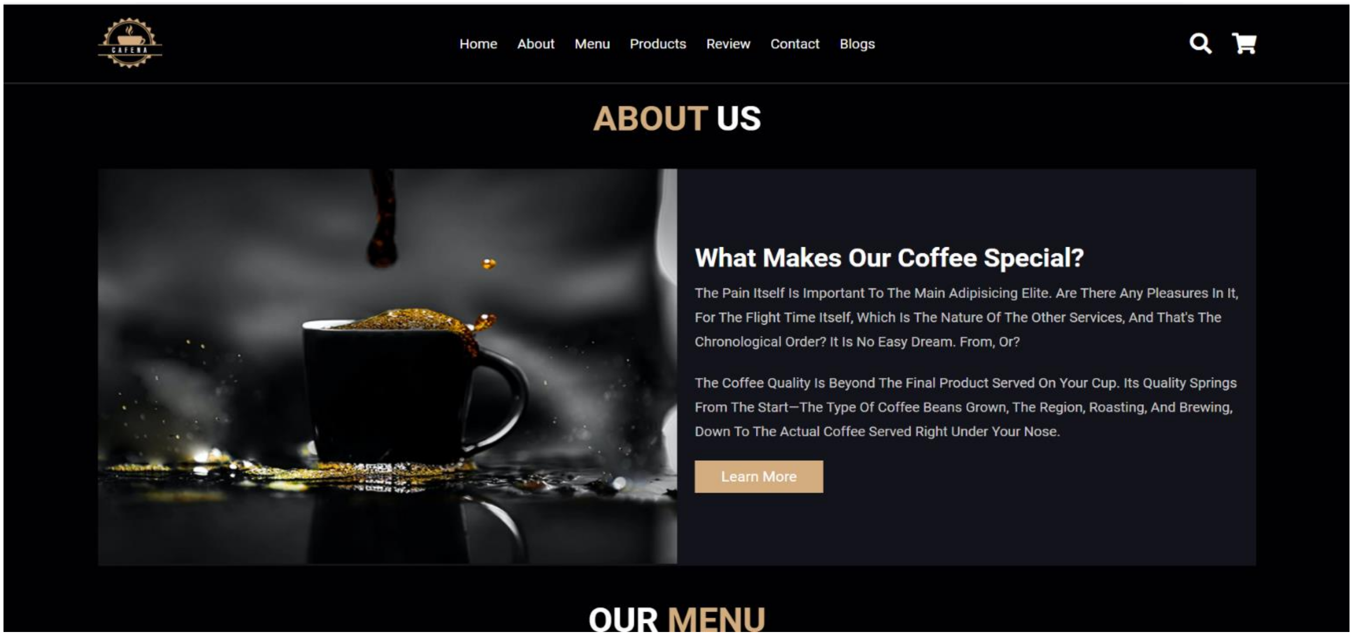
During the development of the application we kept in mind the digital framework and financial network online. And we took an approach on how to create a infrastructure that enables cashless economy. The development of this project started with a collaboration of the team while implementing the basic framework and overview and later the project development started which was at last uploaded on Git-Hub via collaboration.

6. Snapshots of Project



FRESH COFFEE IN THE MORNING

Good Communication Is Just As Stimulating As Black Coffee, And Just As Hard To Sleep After.



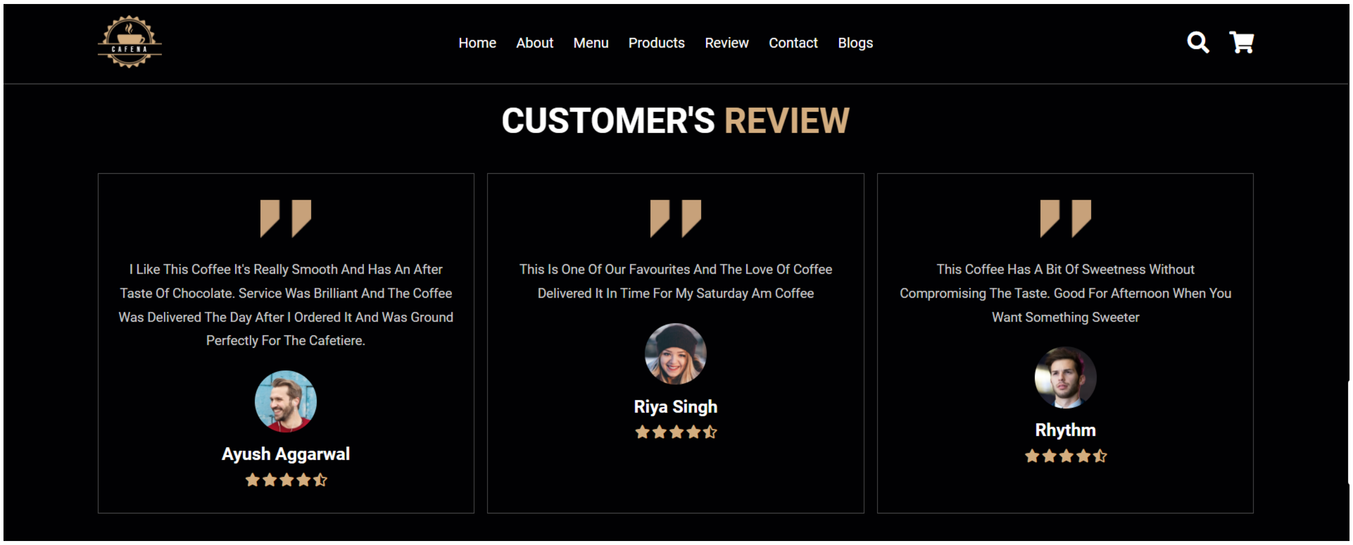
ABOUT US

What Makes Our Coffee Special?

The Pain Itself Is Important To The Main Adipisicing Elite. Are There Any Pleasures In It, For The Flight Time Itself, Which Is The Nature Of The Other Services, And That's The Chronological Order? It Is No Easy Dream. From, Or?

The Coffee Quality Is Beyond The Final Product Served On Your Cup. Its Quality Springs From The Start—The Type Of Coffee Beans Grown, The Region, Roasting, And Brewing, Down To The Actual Coffee Served Right Under Your Nose.

OUR MENU



CUSTOMER'S REVIEW

I Like This Coffee It's Really Smooth And Has An After Taste Of Chocolate. Service Was Brilliant And The Coffee Was Delivered The Day After I Ordered It And Was Ground Perfectly For The Cafetiere.

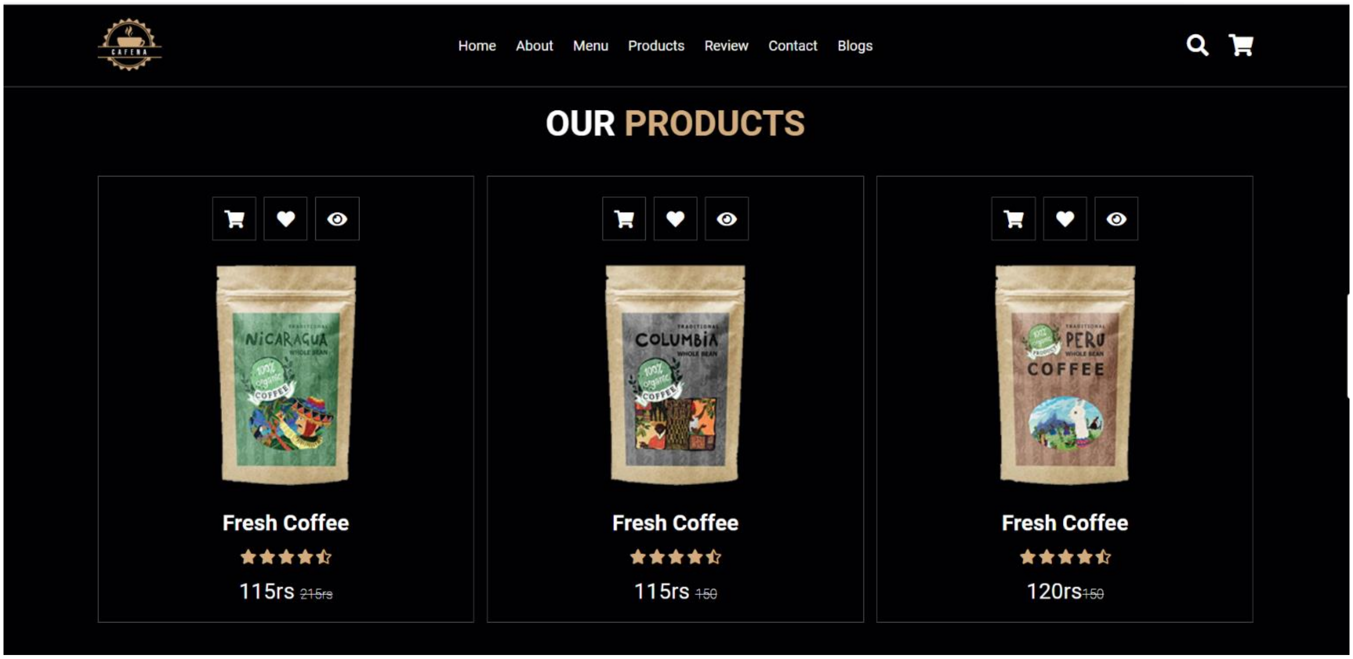
Ayush Aggarwal ★★★★☆

This Is One Of Our Favourites And The Love Of Coffee Delivered It In Time For My Saturday Am Coffee

Riya Singh ★★★★☆

This Coffee Has A Bit Of Sweetness Without Compromising The Taste. Good For Afternoon When You Want Something Sweeter

Rhythm ★★★★☆

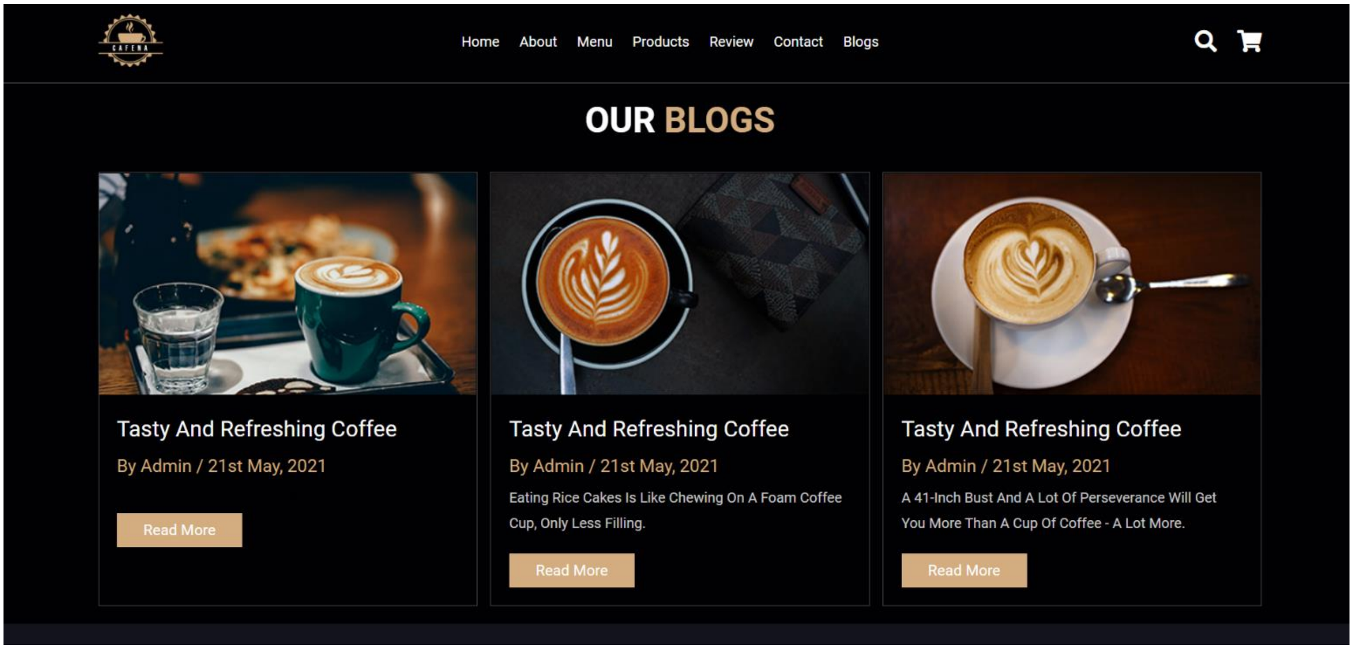


OUR PRODUCTS

Fresh Coffee ★★★★☆ 115rs ~~215rs~~

Fresh Coffee ★★★★☆ 115rs ~~150~~

Fresh Coffee ★★★★☆ 120rs~~150~~



OUR BLOGS

Tasty And Refreshing Coffee

By Admin / 21st May, 2021

Tasty And Refreshing Coffee

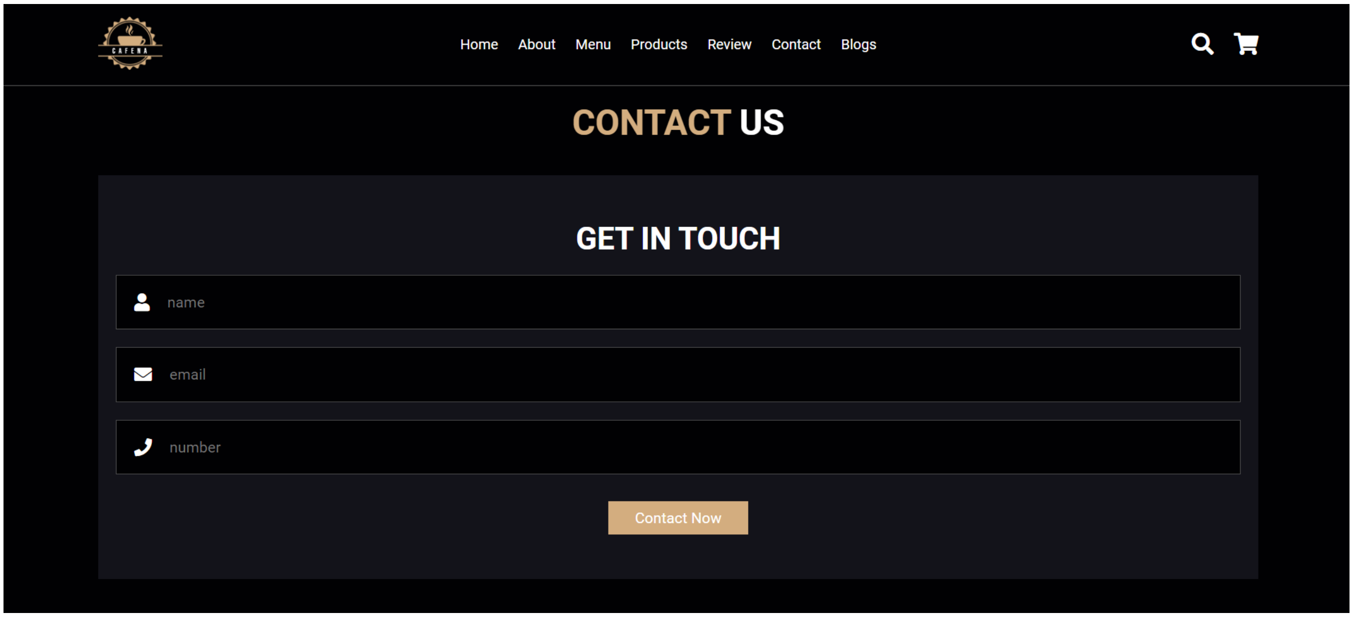
By Admin / 21st May, 2021

Eating Rice Cakes Is Like Chewing On A Foam Coffee Cup, Only Less Filling.

Tasty And Refreshing Coffee

By Admin / 21st May, 2021

A 41-Inch Bust And A Lot Of Perseverance Will Get You More Than A Cup Of Coffee - A Lot More.



CONTACT US

GET IN TOUCH