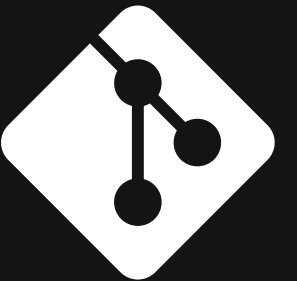
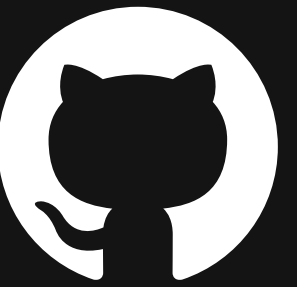


# Version Control with Git and GitHub

*A webinar by:*



DATA SCIENCE  
COMMUNITY SRM



Rotaract Club  
SRM Kattankulatur



# Topics for This Session



## INTRO TO TOPICS

Git, GitHub and GitHub Desktop

## WHAT IS GIT?

Explanation and Important Git commands

## WHAT IS GITHUB?

Explanation and Git Flow

## GITHUB DESKTOP

Working with Repositories

## WORKING WITH GIT, GITHUB & GITHUB DESKTOP

Branching, PR, Commits, Forks



# Speakers

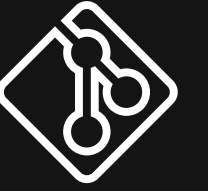


Hritik Bhandari

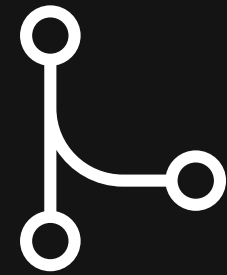


Akshat Anand

# What is Git?



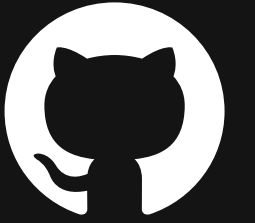
Git is a widely used Version Control System (VCS) that lets you keep track of all the modifications you make to your code. This means that if a new feature is causing any errors, you can easily roll back to a previous version.



But Git isn't just any VCS, it's a Distributed VCS. This means that every collaborator of the project will have a history of the changes made on their local machine. So people can work on different features of the project without having to communicate with the server hosting the remote version of the project. This is super efficient and you can easily merge any changes made to the project with the remote copy.



# What is GitHub?



GitHub is a widely used platform for version control that uses Git at its core. It lets you host the remote version of your project from where all the collaborators can have access to it. Not just your own team members, but any member of GitHub can contribute to your code (that is of course if you choose to accept the changes made).

GitHub is like a social platform where you can find a plethora of open-source projects with their codes. All the new and emerging technologies can be found on this platform. You can collaborate on amazing projects and have discussions on your contributions! This is the best open-source platform you'll find.

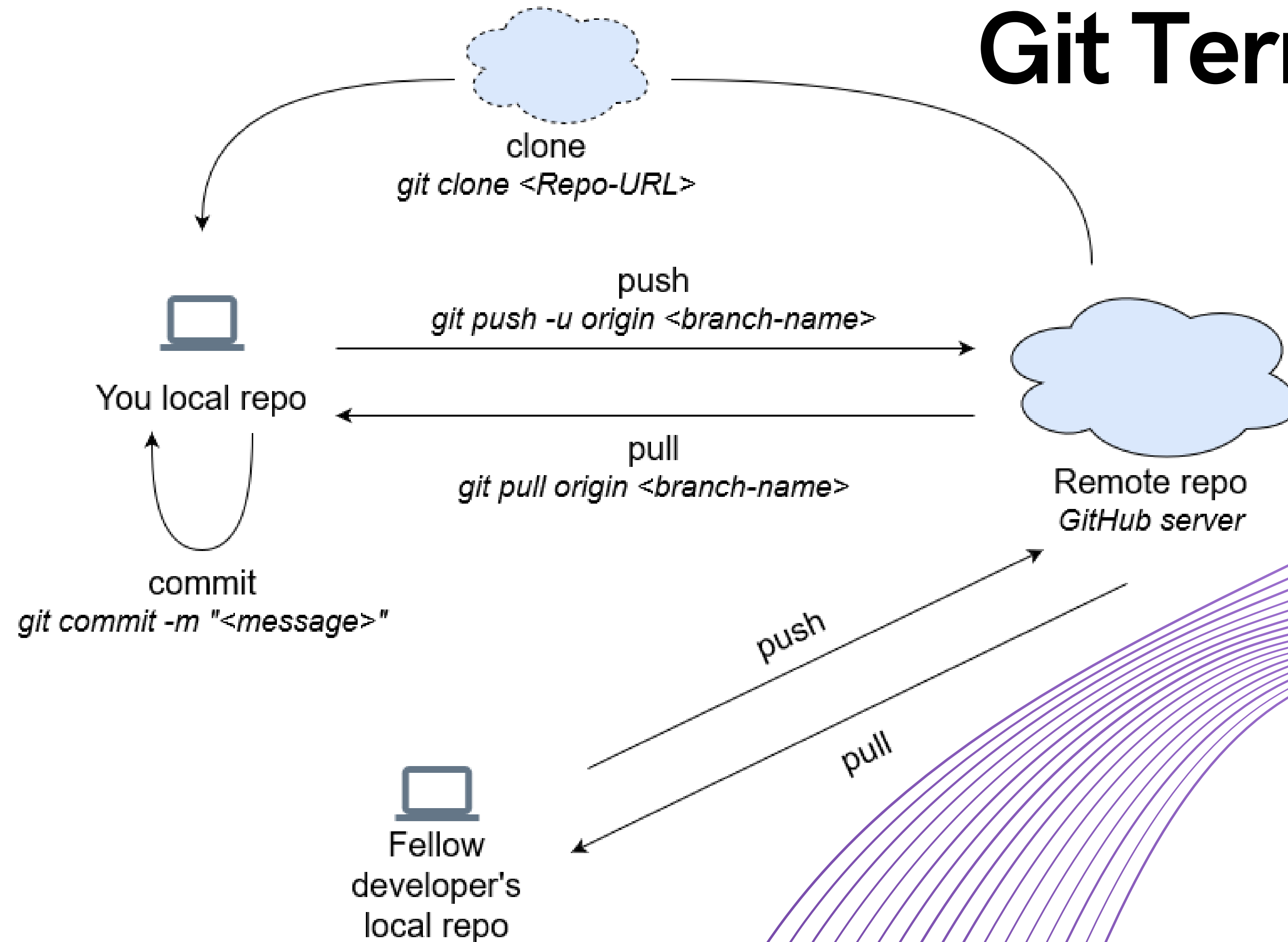


# What is GitHub Desktop?

GitHub Desktop allows us to easily start using version control. GitHub Desktop offers a Graphical User Interface (GUI) to use Git. A GUI allows users to interact with a program using a visual interface rather than relying on text commands.

Though there are some potential advantages to using the command line version of Git in the long run, using a GUI can reduce the learning curve of using version control and Git.

# Git Terminologies



# Repository

Repository or Repo is a folder that contains all the project files and the history of the revisions made to each file.

## REMOTE REPO

It contains your project that can be accessed from anywhere and by anyone. Your remote repository lives on the GitHub server and anybody can access them

## LOCAL REPO

It is a copy of the remote repo that resides on your local machine. All the changes you make will be saved in your local repo.



## CLONING

Cloning means creating a copy of the remote repo on your local machine. Now you can make changes to the project on your local machine.

```
git clone <Repo-URL>
```

## COMMIT

When you commit a change, you save the changes you made to your files in the repo. When working with Git from your local machine, using the commit command will save your files in the local repo.

```
git commit -m "<commit message>"
```

## PUSH

Push command allows you to transfer all the changes on your local repo to the remote repo. Now all the fellow developers will have access to the changes you made and they can update their local repositories.

```
git push origin <branch>
```



## BRANCHES

Branches are an important part of working with Git. Any commits you make will be made on the branch you're currently "checked out" to.

Use `git status` to see which branch that is.

```
git branch [branch-name]  
git checkout [branch-name]
```

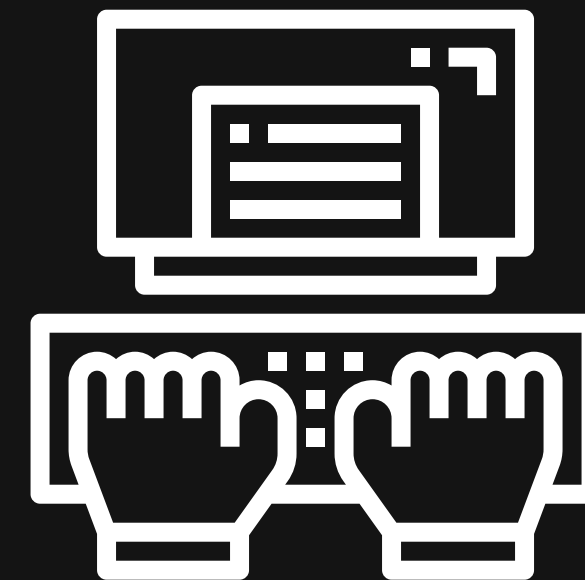
## FORK

A copy of a repository on GitHub owned by a different user.

## PULL REQUEST

a place to compare and discuss the differences introduced on a branch with reviews, comments, integrated tests, and more

It's Hands-On Time!!



# Find all the used git commands

at: [dscommunity.in/gitcommands](https://dscommunity.in/gitcommands) 



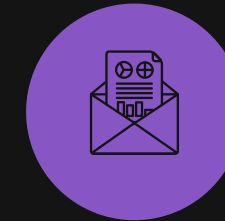
# Resources:

To learn more about Git and GitHub, check out these resources.

1. [GitHub for Developers](#)
2. [Understanding the GitHub Flow](#)
3. [How to write a good Commit Message](#)
4. [How to write the perfect pull request](#)
5. [GitHub Training Kit](#)

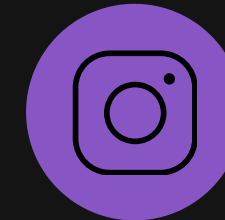
# Thank You!

## GET IN TOUCH WITH US!



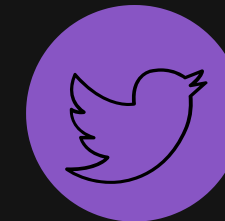
### EMAIL

[datasciencecommunity@gmail.com](mailto:datasciencecommunity@gmail.com)



### INSTAGRAM

[dscommunity\\_srm](https://www.instagram.com/dscommunity_srm)



### TWITTER

[dscommunity\\_srm](https://twitter.com/dscommunity_srm)



### LINKEDIN

[Data Science Community SRM](https://www.linkedin.com/company/data-science-community-srm)



### YOUTUBE

[Data Science Community SRM](https://www.youtube.com/channel/UC...)