GIT & GITHUB

- DONI PRATHAMESH

WHAT IS VERSION CONTROL SYSTEM?

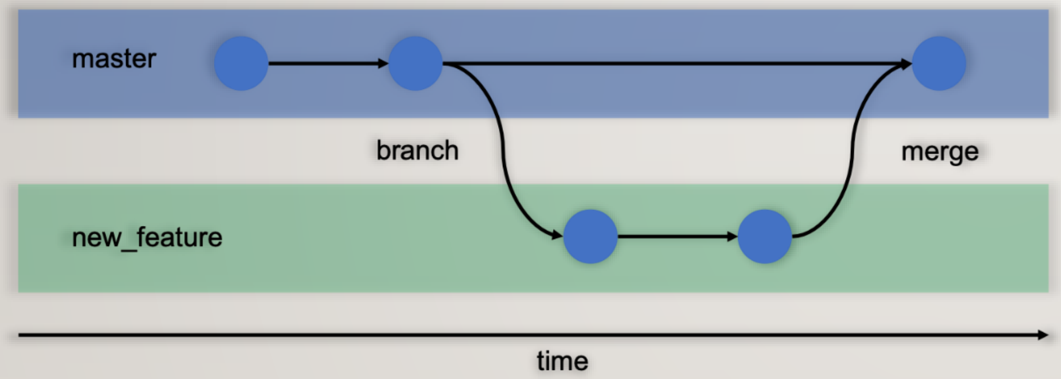
Definition:

- A **Version Control System (VCS)** is a tool that helps software developers track and manage changes to their code over time.

Key Functions of a Version Control System:

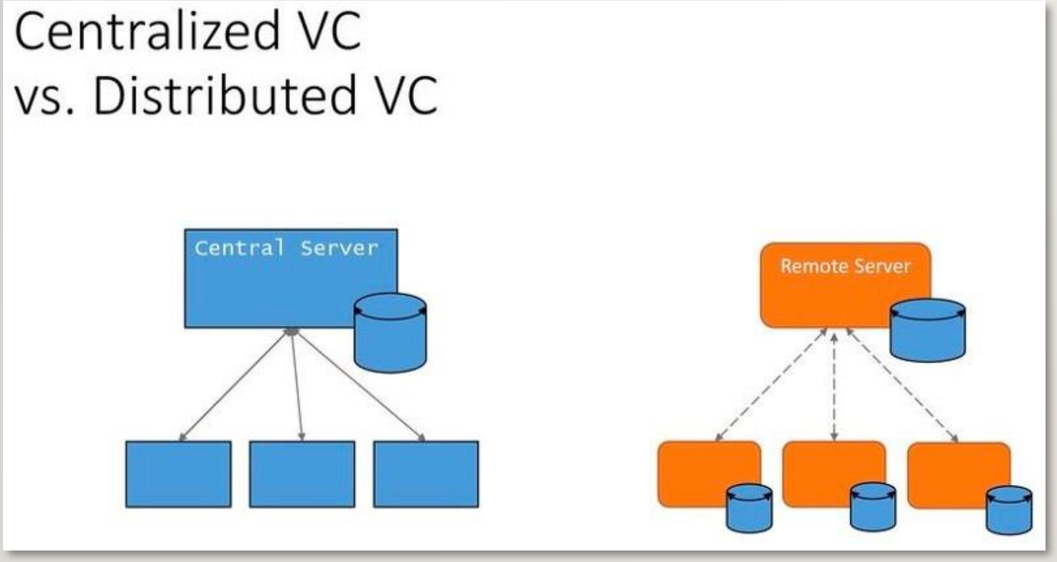
- Track Changes: It keeps a record of every modification made to the files, including what changes were made, by whom, and when.
- Collaboration: Multiple people can work on the same project at once without conflicting with each other's work.
- Revert Changes: If something goes wrong, you can easily roll back to a previous, stable version of the code.
- Branching and Merging: Developers can work on different parts of a project in isolation (branches), then later merge their changes back into the main project

BRANCHING AND MERGING



- **What is Branching?**
  - **Branching** allows developers to work on different parts of a project simultaneously, without affecting the main codebase (often called the **main** or **master** branch).
- **What is Merging?**
  - **Merging** is the process of integrating changes from one branch into another.

TYPES OF VERSION CONTROL SYSTEM (VCS)



- Centralized Version Control (CVCS):
  - A central repository stores all the files and their versions. Developers check out files to work on them and check them back in when done.
  - Example: Subversion (SVN), CVS.
- Distributed Version Control (DVCS):
  - Every developer has a full copy of the entire repository, including its history, on their local machine.
  - Developers can work offline and sync changes when connected to internet.
  - Example: Git, Mercurial.

WHAT IS GIT?

Definition:

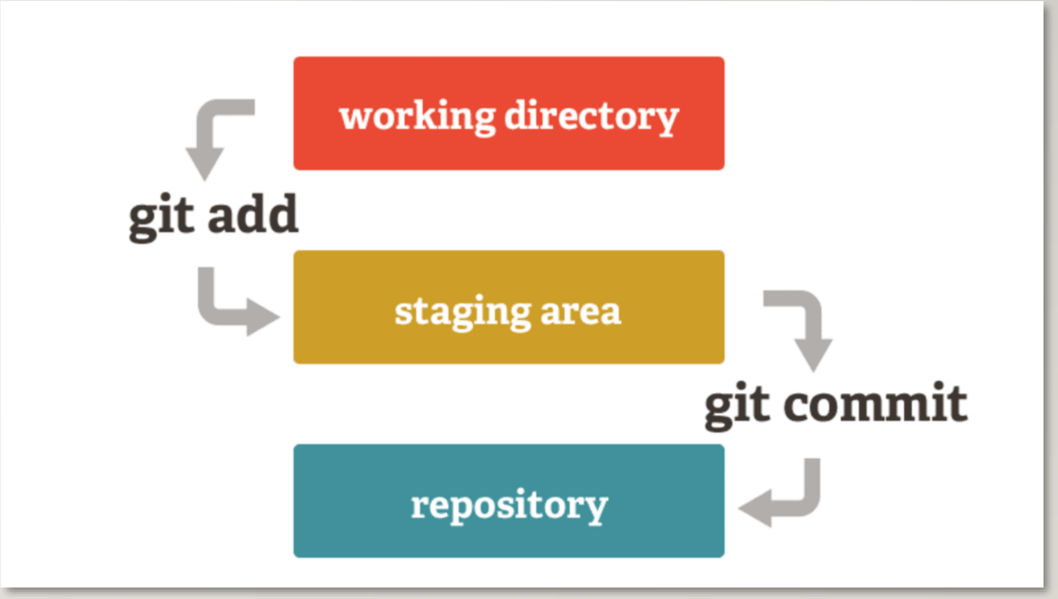
- **Git** is a **distributed version control system (VCS)** that allows multiple developers to track changes in their code, collaborate efficiently, and maintain a history of their work.

Key Functions of Git:

- **Distributed Version Control-** Every developer has their own local repository
- **Collaboration**: Multiple developers can work on the same project simultaneously
- **Staging Area**: Git introduces a staging area between your local changes and the final commit.

GIT – LIFE CYCLE

- **Working Directory**:
  - The working directory is the folder on your local machine where the actual files of your project reside.
- **Staging Area**:
  - The staging area is a temporary area where you can stage changes before committing them to the repository.
- **Repository**:
  - A **repository** (or **repo**) is a storage space where your project and its files are stored.



WHAT IS GITHUB?

Definition:

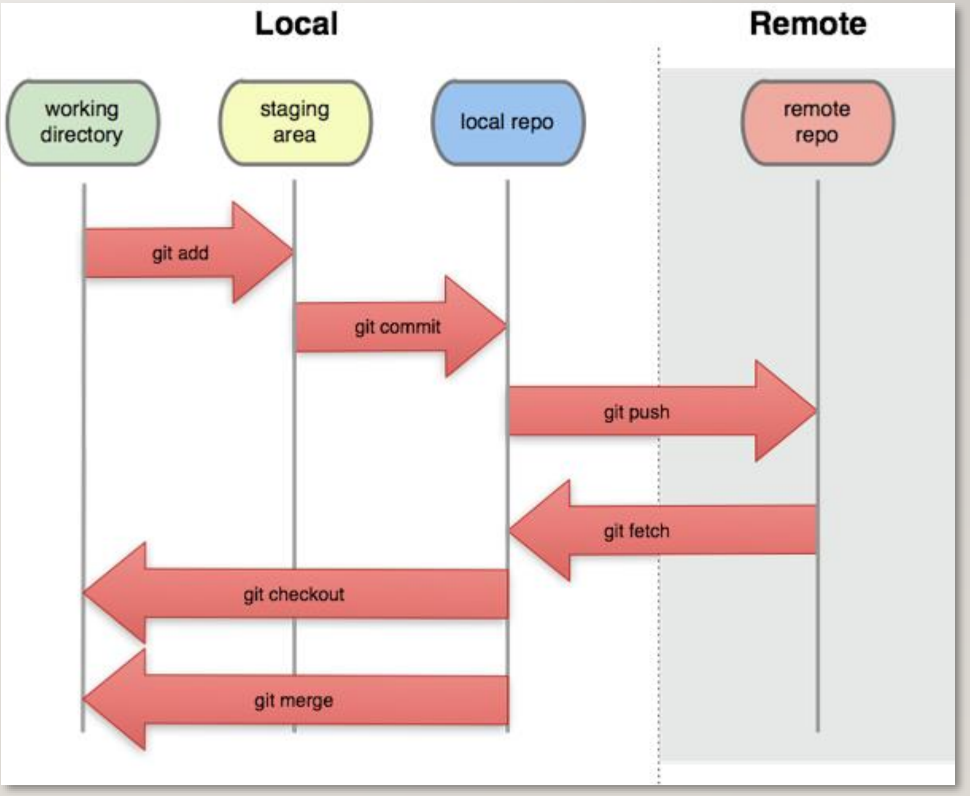
- **GitHub** is a **web-based platform** that provides **hosting for Git repositories** and supports version control and collaboration features.

Feature:

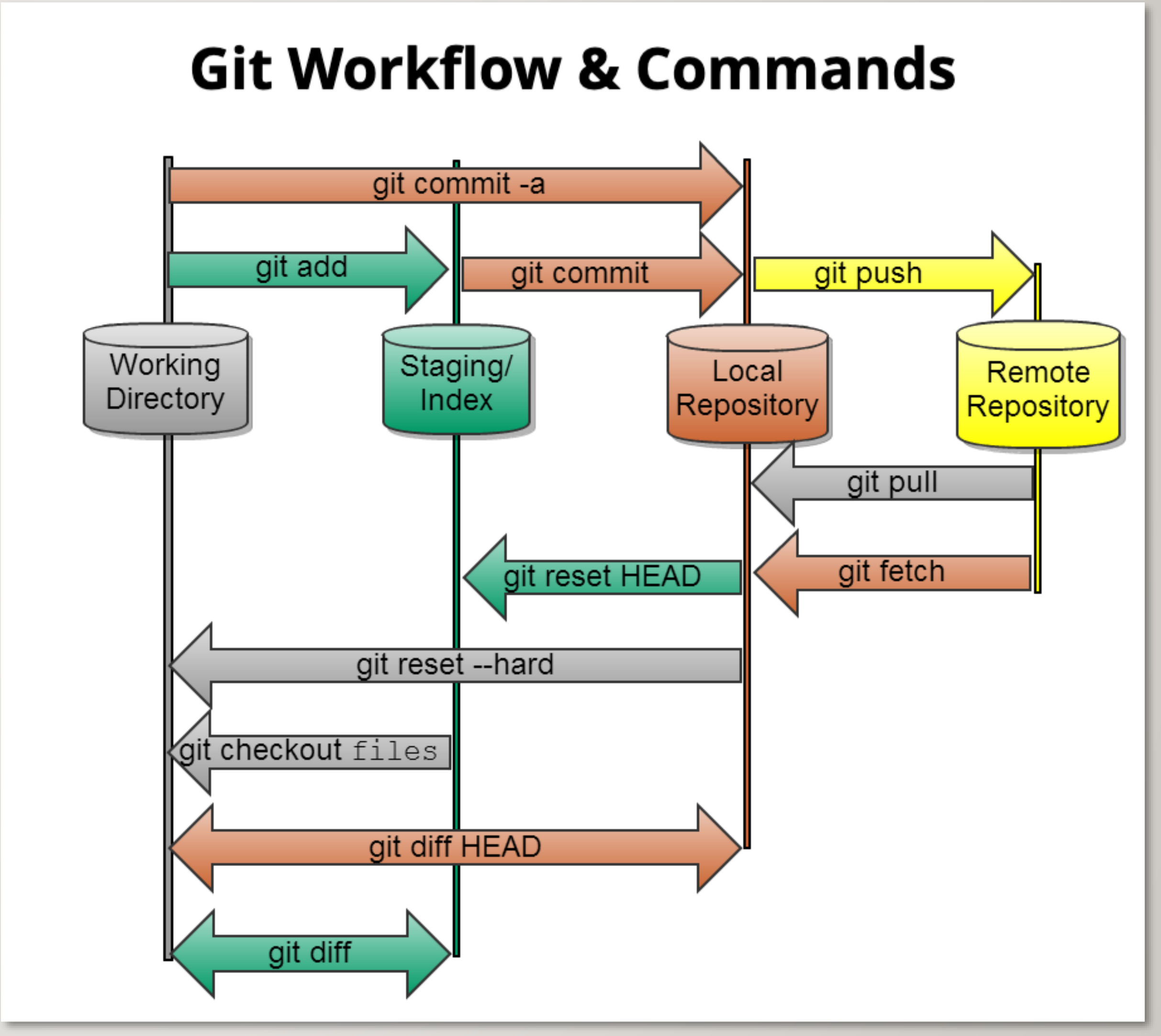
- **Team Collaboration**: Pull requests and code reviews facilitate teamwork.
- **Open Source**: Fork, contribute and improve projects globally.
- **Social Features**: Follow others, star projects, and showcase your work.
- **Community**: Engage in discussions, report issues and work with developers worldwide.

REMOTE GIT REPOSITORIES

- **What is a Remote Repository?**
  - A **remote repository** is essentially a **cloud-based version** of a local Git repository.
  - It allows multiple users to **collaborate on the same project** by providing access to a centralized location where the project code is stored.
  - You can interact with remote repositories by pushing your changes to them and pulling changes from them.



GIT WORKFLOW AND COMMANDS



KEY APPLICATIONS FOR DEVELOPERS

- **Open-Source Contributions**:
  - GitHub has become the home for **open-source development**. Developers can contribute to global projects, fix bugs add features and share code that benefits others.
- **Showcase Your Work**:
  - Developers can showcase their projects by creating **public repositories** on GitHub. This acts as a **portfolio** that potential employers, clients or collaborators can explore to see your skills and coding style.

THANK YOU