

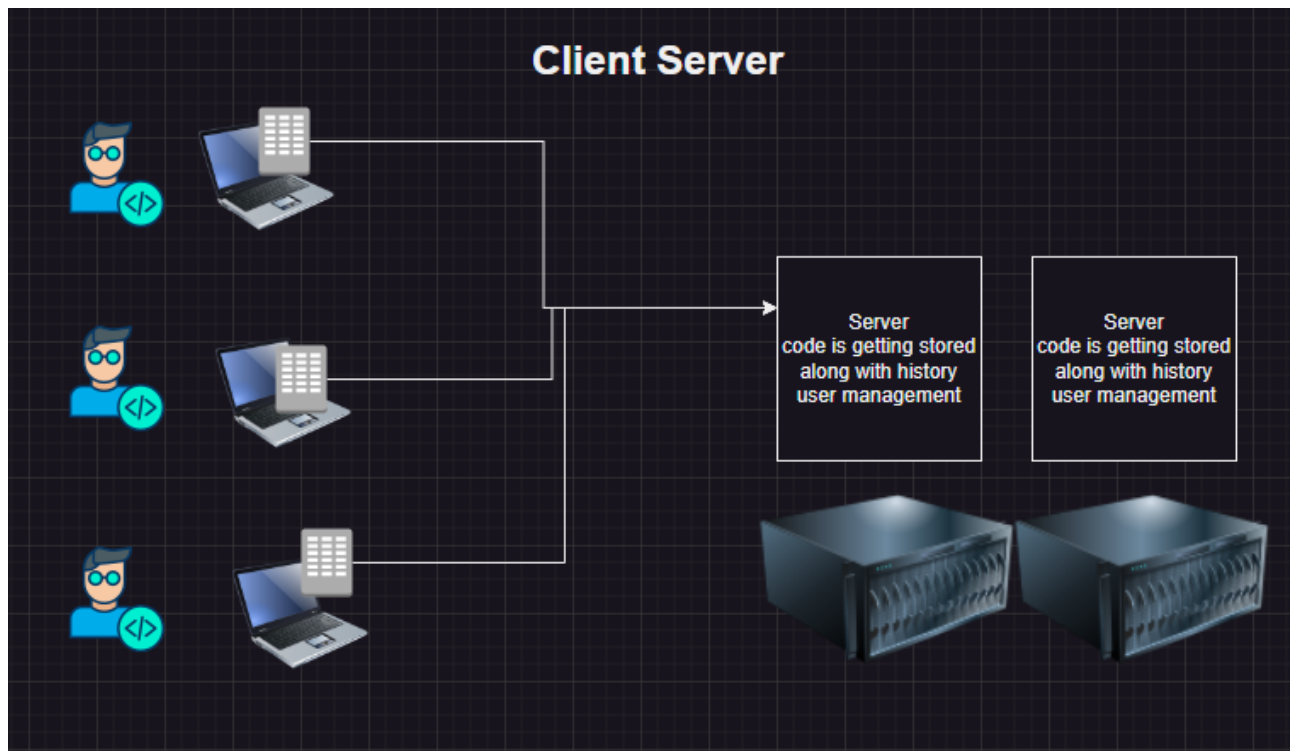
# Git Version Control

## overview

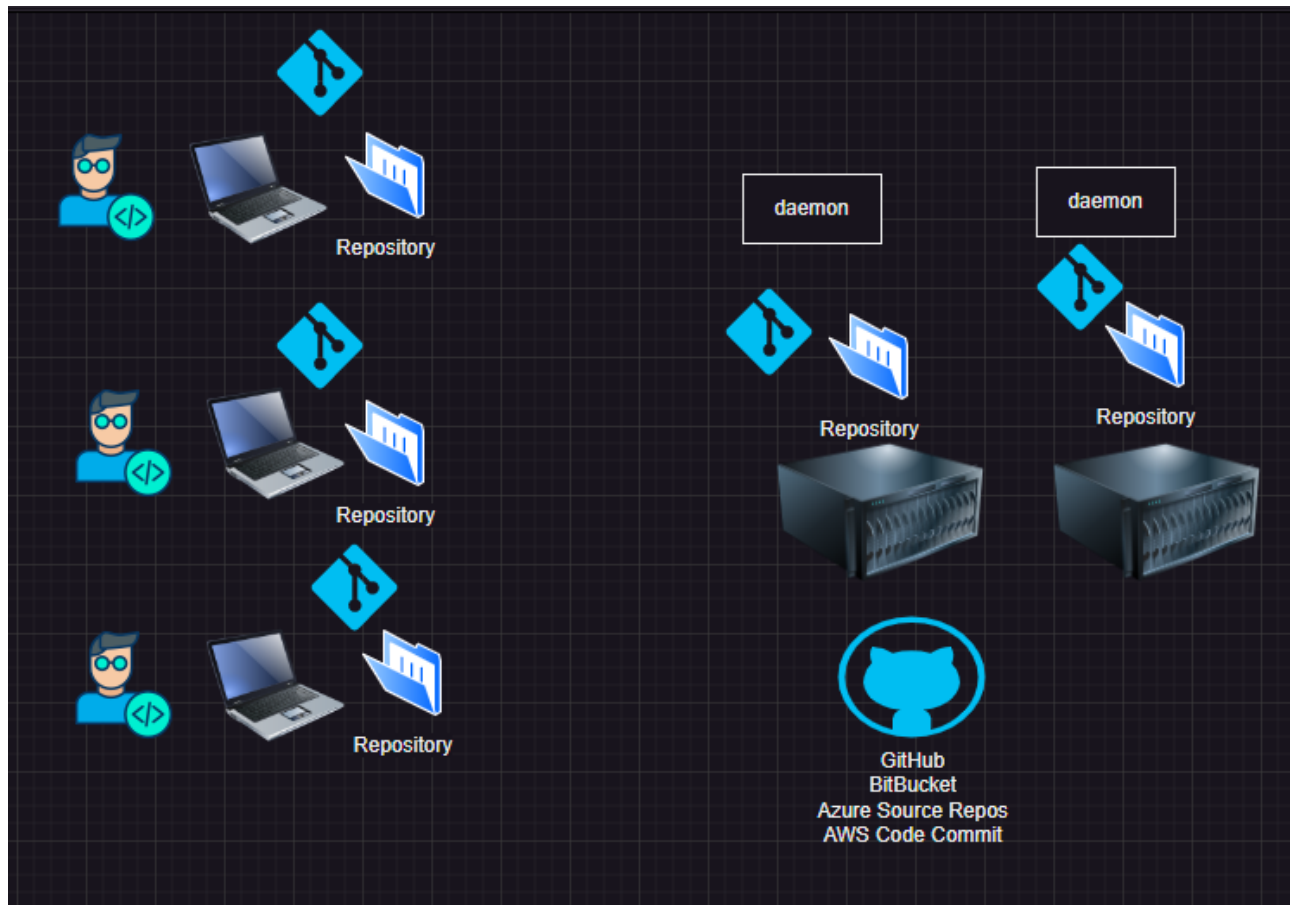
- Use cases for DevOps Engineers:
  - This is where code is hosted for which we are supposed to build pipelines
  - Any work which we do let it be terraform, k8s, docker, ansible etc... we need to store it in version control
  - In some cases, we might be asked to do advanced tasks such as backups etc in git.

## basics

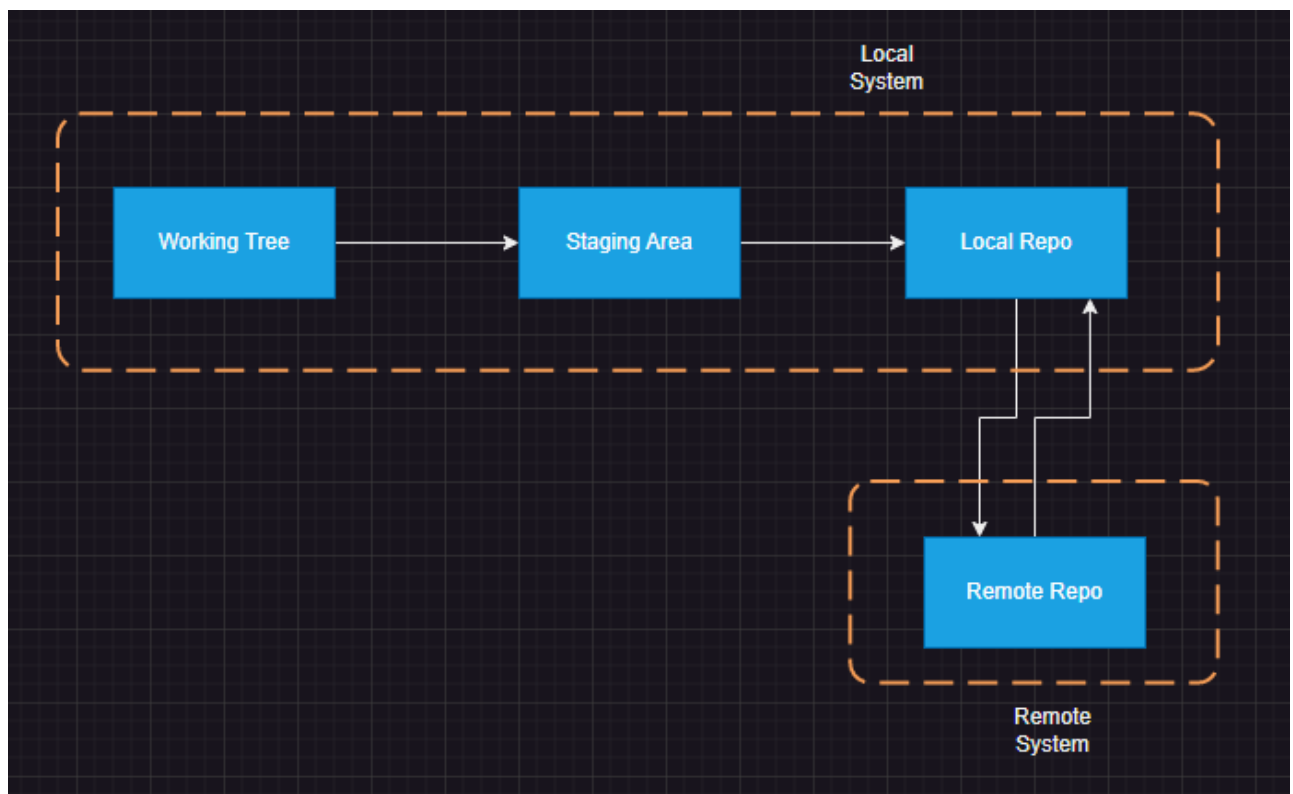
- Client Server VCS



- Distributed VCS



- In Git we deal with two repositories
  - one in local system => local repo
  - other on server/hosted git => remote repo
- Workflow of Git



## Git Learning Plan

- Local working
  - working tree
  - staging area
  - local repo
  - commit
  - branch
  - tag
  - Finding changes
  - modify
  - cherry pick
  - rebase
  - stash
- How Git works ?
- Remote Repository
  - GitHub
  - Azure Source Repos
  - Code Commit
- Branching Strategy (After Pipelines)
- Git Hooks (After Pipelines)
- Pull request (During Pipeline )

## Setup local repository

- Git to maintain history needs to know your username and email id

```
git config --global user.name "<username>"  
git config --global user.email "<email>"
```

- Create a new folder and initialize a git repository

```
git init
```

- Create a new file 1.txt

```
touch 1.txt  
New-Item 1.txt
```

- Now view the status

```
PS C:\temp\GitPractice\hello-git> git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
       1.txt

nothing added to commit but untracked files present (use "git add" to track)
PS C:\temp\GitPractice\hello-git>
```

- Lets add 1.txt to staging area

```
PS C:\temp\GitPractice\hello-git> git add .\1.txt
PS C:\temp\GitPractice\hello-git> git status
On branch master

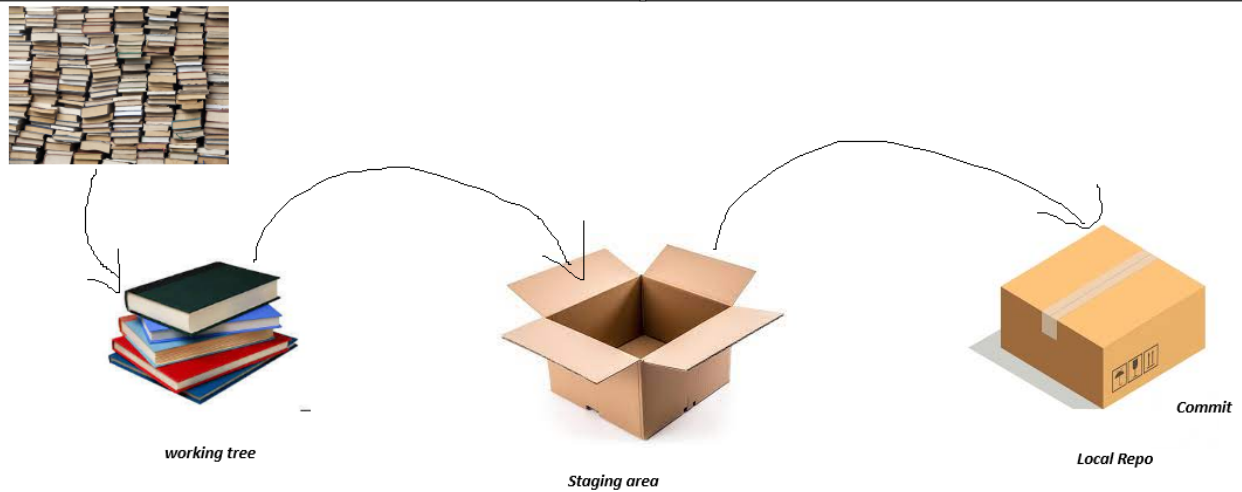
No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
       new file:   1.txt

PS C:\temp\GitPractice\hello-git>
```

- Let commit the changes. To commit we need give a message, username, email id

```
Windows PowerShell
PS C:\temp\GitPractice\hello-git> git commit -m "first commit"
[master (root-commit) 632aa01] first commit
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 1.txt
PS C:\temp\GitPractice\hello-git> git status
On branch master
nothing to commit, working tree clean
PS C:\temp\GitPractice\hello-git>
```



- Lets add more than one change

```

Windows PowerShell
PS C:\temp\GitPractice\hello-git> git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   1.txt
    new file:   2.txt
    new file:   3.txt

PS C:\temp\GitPractice\hello-git> git commit -m "second change"
[master 787fdd4] second change
 3 files changed, 1 insertion(+)
 create mode 100644 2.txt
 create mode 100644 3.txt

PS C:\temp\GitPractice\hello-git> git log
commit 787fdd4272e7d4bf18d1166b9cd3c2caf85a8384 (HEAD -> master)
Author: shaikkhajaibrahim <qtkhajadevops@gmail.com>
Date:   Wed Mar 27 08:10:56 2024 +0530

    second change

commit 632aa01b91fd4bc782491ff4513334bbeed86562
Author: shaikkhajaibrahim <qtkhajadevops@gmail.com>
Date:   Wed Mar 27 08:01:42 2024 +0530

    first commit

PS C:\temp\GitPractice\hello-git>

```

```
PS C:\temp\GitPractice\hello-git> git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   2.txt
        new file:   4.txt

PS C:\temp\GitPractice\hello-git> git restore --staged .\4.txt
PS C:\temp\GitPractice\hello-git> git status
On branch master
Changes to be committed:
  (use "git restore --staged <file>..." to unstage)
        modified:   2.txt

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        4.txt

PS C:\temp\GitPractice\hello-git> git commit -m "third change"
[master 3e42d4e] third change
1 file changed, 1 insertion(+)
PS C:\temp\GitPractice\hello-git> git status
On branch master
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        4.txt

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

- Find a git command to
  - To remove untracked files `git clean -fd`
  - To remove all the changes in working tree `git reset --hard`
- To view history `git log`