

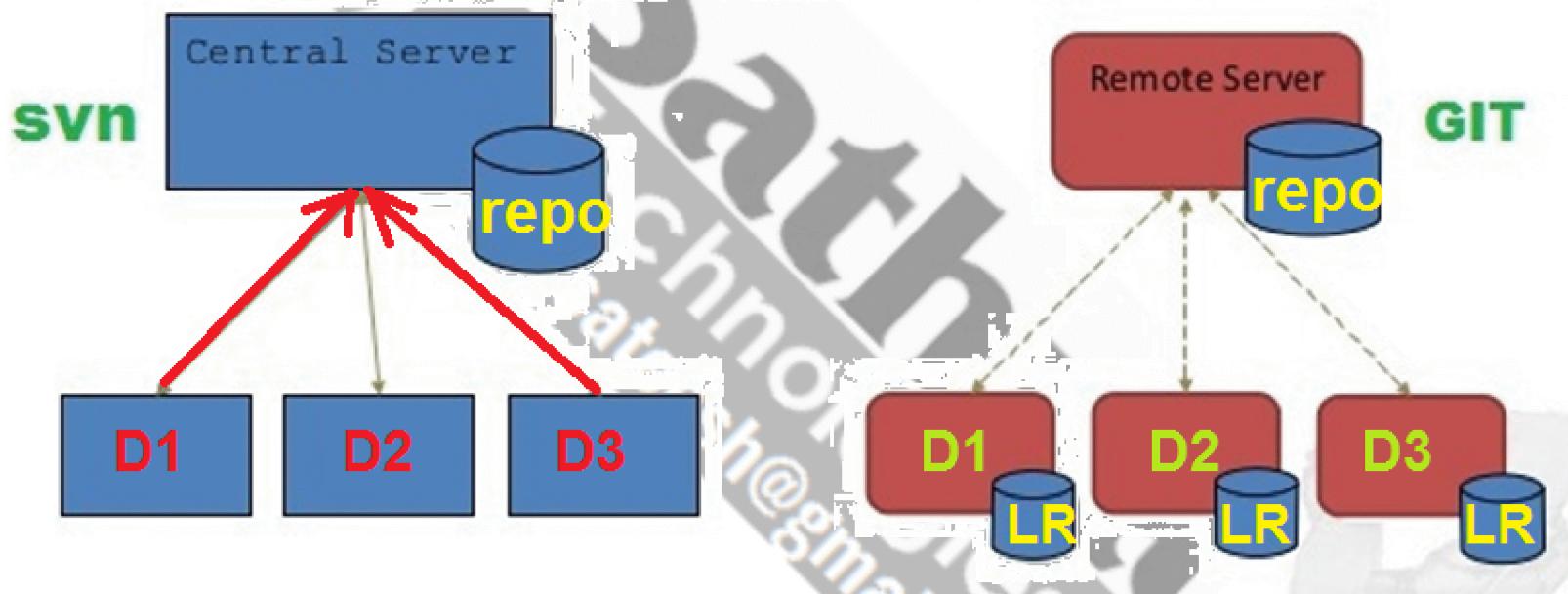


git



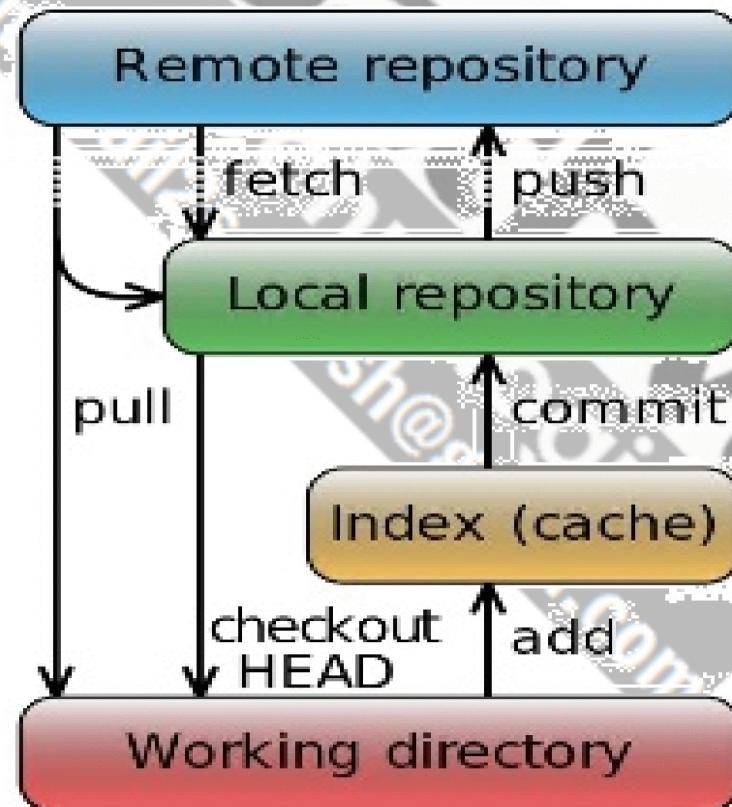
GitHub

Centralized vs Distributed



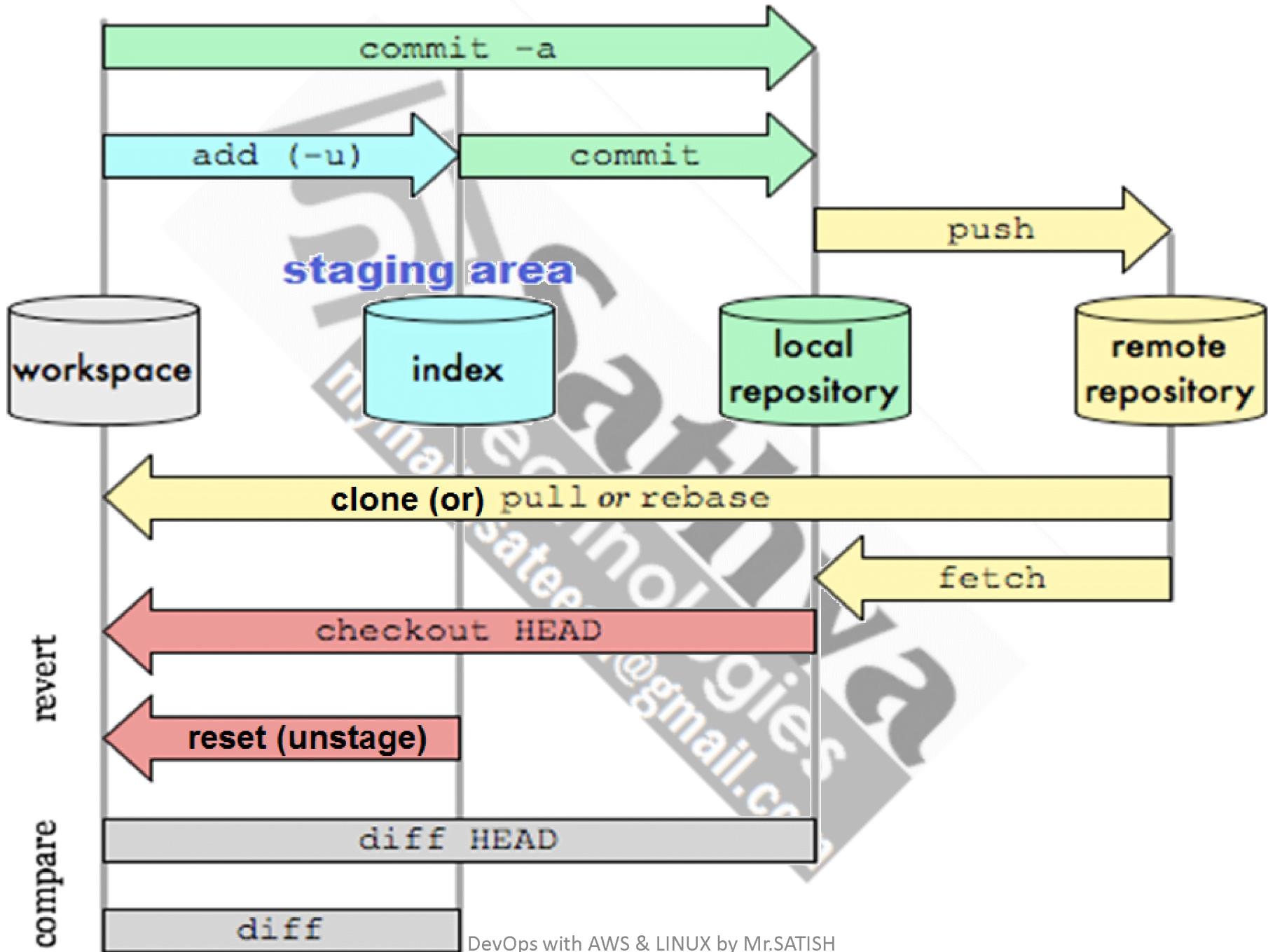


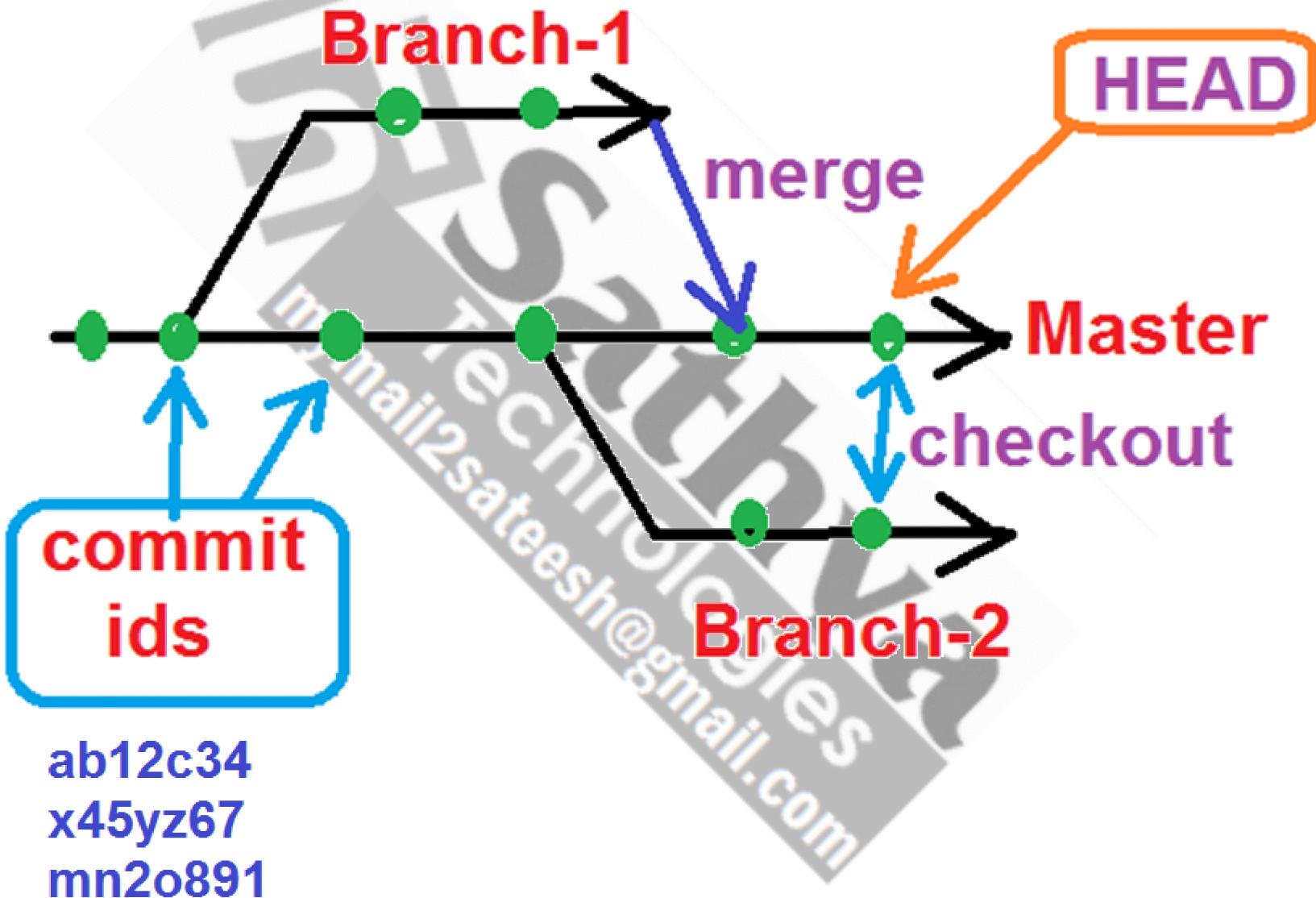
Git Data Flow



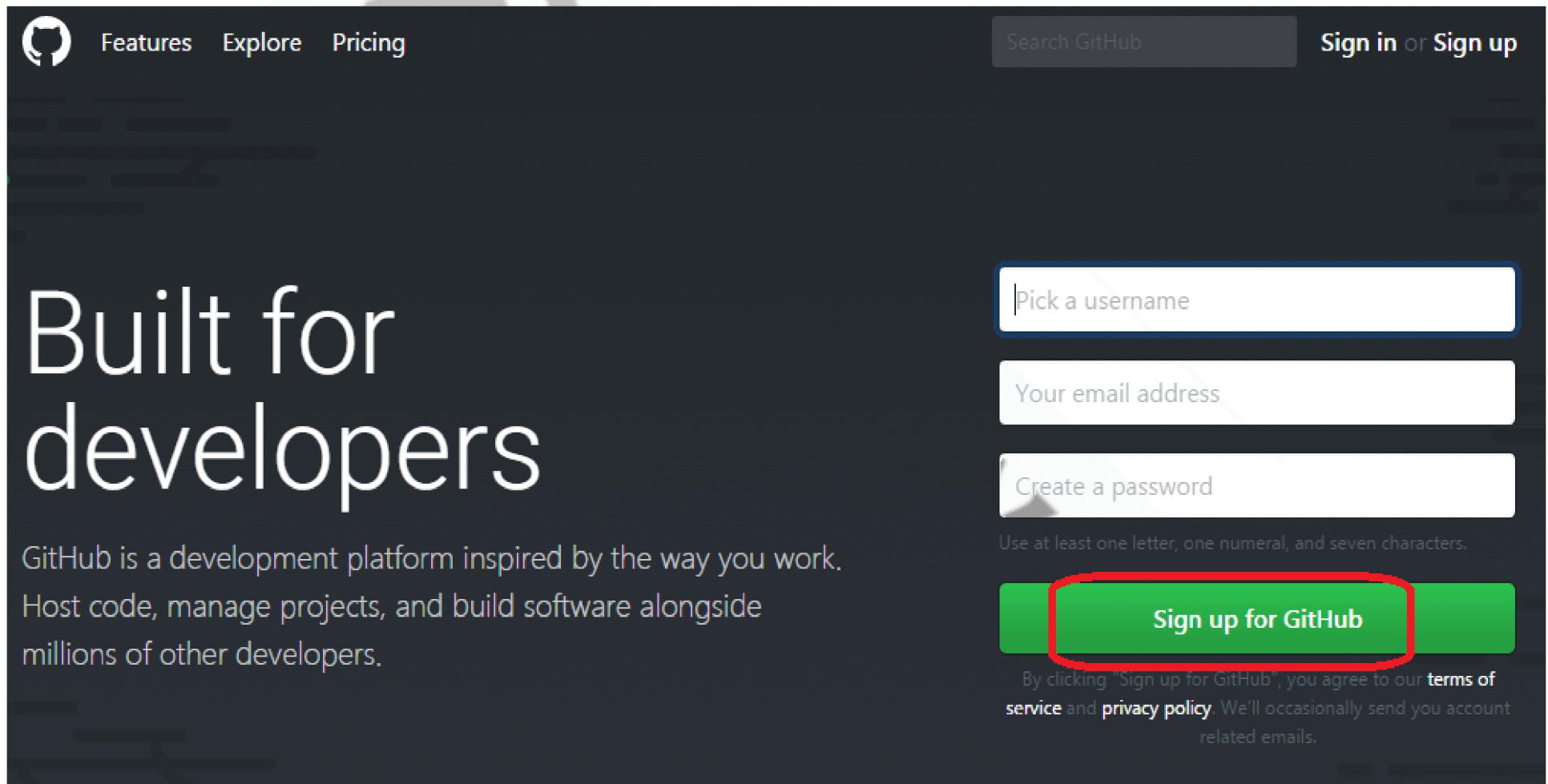
Git Workflow

- Fetch or clone (create a copy of the remote repository) (compare to cvs check out)
- Modify the files in the local branch
- Stage the files (no cvs comparison)
- Commit the files locally (no cvs comparison)
- Push changes to remote repository (compare to cvs commit)





Create Account in GitHub



The image shows the GitHub sign-up page. At the top, there is a navigation bar with links for 'Features', 'Explore', and 'Pricing'. On the right side of the bar are search and sign-in/sign-up buttons. Below the navigation bar, the text 'Built for developers' is displayed in large white font. To the right, there are three input fields: 'Pick a username', 'Your email address', and 'Create a password'. A note below the password field says 'Use at least one letter, one numeral, and seven characters.' At the bottom right is a green 'Sign up for GitHub' button, which is highlighted with a red rectangle. Below the button, a small note states: 'By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We'll occasionally send you account related emails.'

Features Explore Pricing

Search GitHub

Sign in or Sign up

Built for developers

GitHub is a development platform inspired by the way you work. Host code, manage projects, and build software alongside millions of other developers.

Pick a username

Your email address

Create a password

Use at least one letter, one numeral, and seven characters.

Sign up for GitHub

By clicking "Sign up for GitHub", you agree to our [terms of service](#) and [privacy policy](#). We'll occasionally send you account related emails.

Create Repository

Create a new repository

A repository contains all the files for your project, including the revision history.

Owner

 sathyadevops ▾

Repository name

/ myproj 

Great repository names are short and memorable. Need inspiration? How about [automatic-octo-guacamole](#).

Description (optional)

 Public

Anyone can see this repository. You choose who can commit.

 Private

You choose who can see and commit to this repository.

Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

Add .gitignore: **None** ▾

Add a license: **None** ▾



Create repository

Install Git on Ubuntu 14.04

Step 1: Installation

```
#apt-get update  
#apt-get install git-core -y  
#git --version
```

Step 2: Configuration

```
#git config --global user.name sathyadevops  
#git config --global user.email  
sathyadevops1@gmail.com  
#cat .gitconfig (or) #git config --list
```

Step 3: Create GIT repository

```
#mkdir /repos  
#cd /repos  
#git init  
#ls -a  
#git clone
```

<https://github.com/sathyadevops/myproj.git>

Step 4: Working with Git Repository

```
#echo "Welcome to Git" >> README.md  
#git status
```

to add a file to cache (staging Area)

```
#git add README.md  
#git status
```

to move a file from Staging Area to Local Repo

```
#git commit -m "initial commit"
```

to Add and Commit a file at a time

```
#git commit -a -m "initial commit"
```

to push the code to Central Repo(master)

```
#git push -u origin master
```

To changed files in your working repository

```
#git status -s
```

To show all git commits

```
#git log
```

```
#git log -p
```

```
#git log --since=12-03-2017 --until=13-03-2017
```

```
#git log --oneline
```

To push Code to GitHub by using key:

Step1: Generate key

```
#ssh-keygen
```

Step2: add public key to github project

github → project → settings → deploy keys

Step3: set remote github url

Syntax:

```
git remote set-url origin  
git@github.com:<Username>/<Project>.git
```

Ex:

```
#git remote set-url origin  
git@github.com:sathyadevops/newproj.git
```

To made changes to tracked files

#git diff

#git log

#git log -1

#git diff 57af6s43d..9wg5c2ys3

To list all branches

#git branch

to work with branches:

#git branch branch1

#git checkout branch1

#git branch

#vi index

new line from branch

#git commit -am "new line from branch"

#git push -u origin branch1

check in browser → github

to merge the branch code into master

```
#git checkout master
```

```
#git merge branch1
```

```
#cat index.html
```

```
#git push -u origin master
```

to delete a Branch:

```
#git branch -d branch1
```

to delete a Branch without merging the Data:

```
#git branch -D branch1
```

```
#git push origin --delete br1
```

(to delete remote branch)

Git - Review Changes

```
# git diff
```

```
# git log
```

```
# git show
```

```
c0f455906befd100192848233fb896d081e22  
84
```

Git – Remote Server

```
#git remote -v
```

```
#git checkout -- .    (to revert all the changes)
```