git process

# **🚀 GitHub Collaboration with SSH + Branch Workflow**

## **🔑 Step 1: Set up SSH (one-time setup)**

This avoids entering username/password every time.

1. **Generate SSH Key**

ssh-keygen -t ed25519 -C "your\_email@example.com"

* Press Enter for defaults.

1. **Start SSH agent & add your key**

eval "$(ssh-agent -s)"

ssh-add ~/.ssh/id\_ed25519

1. **Copy your public key**

cat ~/.ssh/id\_ed25519.pub

* Copy the whole output (starts with ssh-ed25519...).

1. **Add to GitHub**
   * Go to 👉 [GitHub SSH Keys](https://github.com/settings/keys)
   * Click **New SSH Key** → Paste the key → Save.
2. **Clone with SSH**

git clone git@github.com:VellankiMahesh7779/mahesh\_vellanki.git

✅ Done → No passwords/tokens needed anymore!

## **🌱 Step 2: Work with Remote Branches**

When your friend wants to create/access branches:

1. **Fetch all branches**

git fetch origin

1. **See available remote branches**

git branch -r

Example:

origin/main

origin/dev

origin/feature/new-ui

1. **Check out a remote branch locally**

git checkout -b dev origin/dev

* This creates a local branch dev that tracks the remote origin/dev.

1. **Verify tracking**

git branch -vv

Example output:

\* dev abc1234 [origin/dev] latest commit message

## **📝 Step 3: Push Your Own Branch**

If your friend creates a new branch and pushes it to your repo:

# Create and switch to a new branch

git checkout -b vellanki557

# Do changes → commit them

git add .

git commit -m "Added new feature"

# Push to your GitHub repo

git push origin vellanki557

Or if already on the branch:

git push origin HEAD:vellanki557

## **⚡ Bonus: Compile & Run C++ (your example)**

g++ c++.cpp -o c++

./c++

✅ **Summary**

* Use SSH once → no passwords.
* Your friend can create **feature branches** in your repo.
* Push their branch → then merge into main via Pull Request (PR).