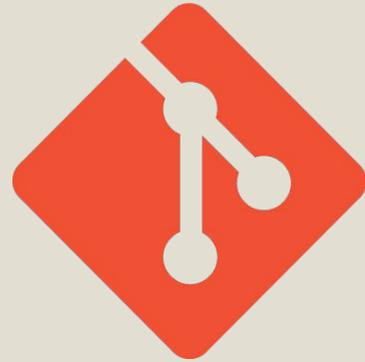


# Self Hosted Git



# git

What is git?

Version control - tracking, organizing, managing changes to a project (solo or collaboratively)

<https://git-scm.com/>

What is GitHub?

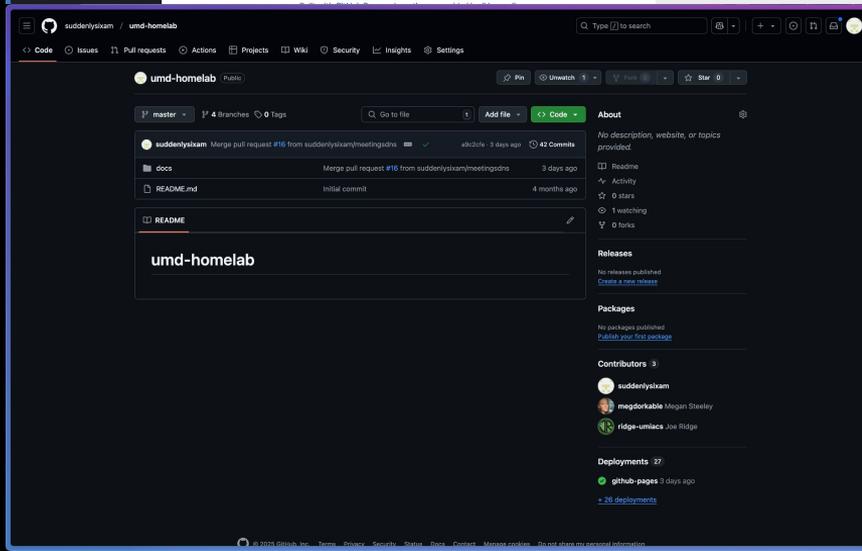
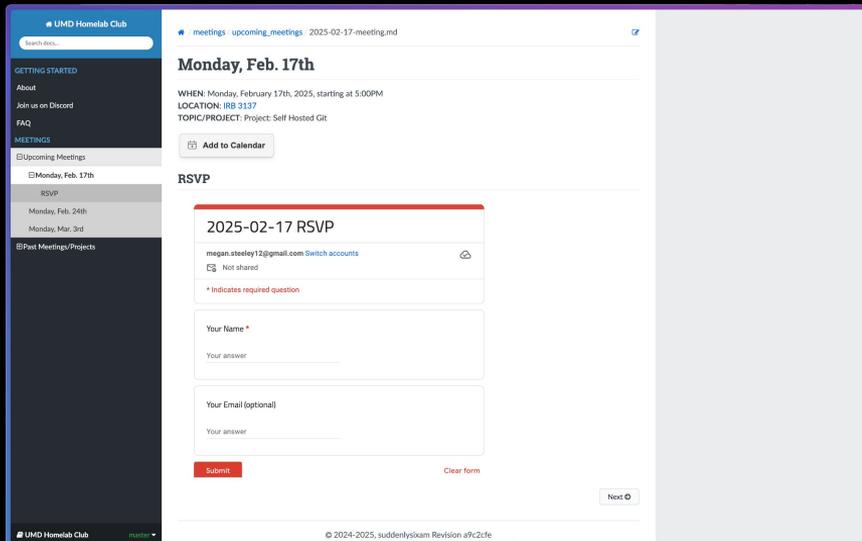
Online software development platform that uses git for version control, and provides many other additional features.

<https://github.com/>

# GitHub Example:

→ GitHub Pages used to build the club's website

<https://suddenlysixam.club/>



## Why use git in general?

- **Version control & history**
  - See when changes were made and by who
  - Rollback to previous versions
- **Backup & disaster recovery (DR)**
- **Collaboration or multi-device (multiple working copies)**
  - Team workflows / approval processes
- **etc.**

## Why self host?

- **Services like GitHub may be paid (or paid for additional features or resources)**
- **Control your own data & who can access it**
- **Local homelab services can access without general access to the internet**

Let's get git set up  
on the Raspberry Pi!

Install packages:

```
sudo apt update  
sudo apt upgrade
```

```
sudo apt install git
```

(Note: git should already be installed, but we should make sure!

(Note: if you run into problems, make sure your internet connection is working properly, and then check the next slide)

(If needed) Time setup:

Note - if your time is not syncing like mine was, stop and start NTP:

```
sudo timedatectl set-ntp False  
timedatectl status  
sudo timedatectl set-ntp True  
timedatectl status
```

Ensure the time is correct, then go back and retry the previous commands

Enable SSH:

*sudo raspi-config*

*Arrow key to '3 Interface Options' and hit Enter*

*Arrow key to 'I1 SSH' and hit Enter*

*Arrow key to 'Yes' and hit Enter*

(Note: You can also do this through the Preferences menu under 'Raspberry Pi Configuration')  
(Note: There are other protocols that you can use other than SSH if you want - more info at <https://git-scm.com/book/en/v2/Git-on-the-Server-The-Protocols>)  
(See also: the Remote Access presentation for how you would access this away from home)

Set up the repo:

```
mkdir -p ~/devel/test-repo  
cd ~/devel/test-repo  
git init --bare
```

(Note: `pwd` and `ls` are also helpful commands here just to see what we are doing)

(Note: The `-p` flag creates any intermediate directories as required)

(Note: You can choose a different directory path, I have chosen to put this under `~/devel/` for this demo.)

Find IP address:

Find the pi's IP address:

*ifconfig*

(Note: we are looking for the IP address for its ethernet or wifi connection, e.g. *eth0*)

git setup:

On a different device:

*(move to or make a directory for the repo,  
method depends on OS)*

```
git init
```

```
git remote add test-repo <username>@<pi-ip>:~/devel/test-repo
```

```
git remote -v
```

(Note: make sure you substitute the values in <> with the appropriate ones based on your setup)

(Note: you can also use this repository locally on the same device - this demonstration is adding the additional step of accessing it on a different device)

(Note: There are other protocols that you can use other than SSH if you want - more info at

<https://git-scm.com/book/en/v2/Git-on-the-Server-The-Protocols>)

git setup:

Still on a different device:

```
vi test.txt
```

*(Add any text you want, save & quit)*

```
git status
```

```
git add test.txt
```

```
git commit -m "My first commit"
```

```
git push test-repo master
```

(Note: if you run into access trouble, make sure you have correctly enabled SSH)

(Note: if you run into other problems, make sure you check the next slide)

`git setup:`

If you have never used git before on the device you are pushing from, and run into problems, you may need to configure git

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

(Note: make sure you substitute the values in <> with the appropriate ones based on your setup)

git setup:

Still on a different device:

*(We can create a different folder to test that it is working)*

```
git clone <username>@<pi-ip>:~/devel/test-repo
```

(Note: if you run into access trouble, make sure you have correctly enabled SSH)

(Note: There are other protocols that you can use other than SSH if you want - more info at

<https://git-scm.com/book/en/v2/Git-on-the-Server-The-Protocols>)

Some GUI options - *there are many more!*:



**GitLab**

<https://about.gitlab.com/>



**Gitea**

<https://about.gitea.com/>



**GitBucket**

<https://gitbucket.github.io/>

**Let's briefly demo Gitea  
on the Raspberry Pi**

Install Docker:

Install docker via the apt repository:

<https://docs.docker.com/engine/install/debian/#install-using-the-repository>

(Note: If you are using Raspberry Pi OS (64-bit) like we are, you will need to use the Debian instructions as linked above, and not the Raspberry Pi OS (32-bit) instructions. You will also need to substitute in the OS version as described in the steps, which you can find on the pi by running `hostnamectl`)

(Note: If you are running a different OS or version, follow that OS's instructions)

Configure Gitea:

```
sudo mkdir -p /opt/stacks/gitea/  
cd /opt/stacks/gitea  
sudo vi compose.yaml
```

## Configure Gitea:

```
services:
  server:
    image: gitea/gitea:latest
    container_name: gitea
    environment:
      - USER_UID=1000
      - USER_GID=1000
      - GITEA__database__DB_TYPE=mysql
      - GITEA__database__HOST=db:3306
      - GITEA__database__NAME=gitea
      - GITEA__database__USER=gitea
      - GITEA__database__PASSWD=gitea
    restart: always
    networks:
      - gitea
    volumes:
      - ./gitea:/data
      - /etc/timezone:/etc/timezone:ro
      - /etc/localtime:/etc/localtime:ro
    ports:
      - "3000:3000"
      - "222:22"
    depends_on:
      - db

  db:
    image: mysql:8
    restart: always
    environment:
      - MYSQL_ROOT_PASSWORD=gitea
      - MYSQL_USER=gitea
      - MYSQL_PASSWORD=gitea
      - MYSQL_DATABASE=gitea
    networks:
      - gitea
    volumes:
      - ./mysql:/var/lib/mysql

networks:
  gitea:
    external: false
```

For copy & paste-ability: [https://suddenlysixam.club/meetings/upcoming\\_meetings/2025-02-17-meeting.html](https://suddenlysixam.club/meetings/upcoming_meetings/2025-02-17-meeting.html)

(Note: normally you would want to do some additional configuration here, such as changing the username and password. We are not for the sake of this demo)

Configure Gitea:

```
sudo docker compose up -d
```

*Navigate to <http://localhost:3000> in a web browser - create a user & configure as you wish!*

# Thank you!

## Don't forget to join the Discord!

<https://discord.gg/H6MsBwSTwZ>

