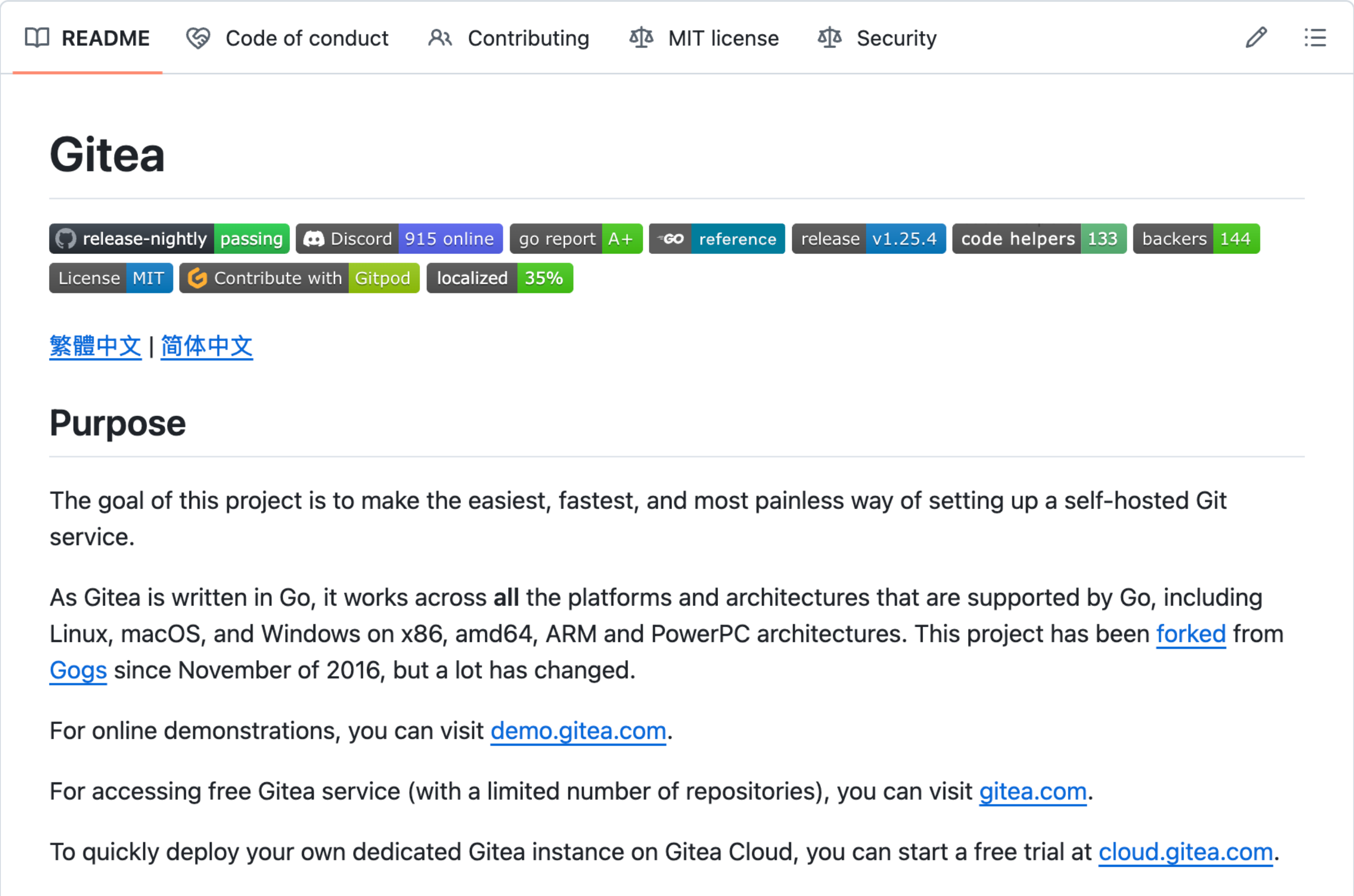
# Demo: Creating a GitHub Alternative on EC2

CSC 410/510

Richard Kelley

# GitHub Alternatives



# Gitea

繁體中文 | 简体中文

## Purpose

The goal of this project is to make the easiest, fastest, and most painless way of setting up a self-hosted Git service.

As Gitea is written in Go, it works across **all** the platforms and architectures that are supported by Go, including Linux, macOS, and Windows on x86, amd64, ARM and PowerPC architectures. This project has been forked from Gogs since November of 2016, but a lot has changed.

For online demonstrations, you can visit demo.gitea.com.

For accessing free Gitea service (with a limited number of repositories), you can visit gitea.com.

To quickly deploy your own dedicated Gitea instance on Gitea Cloud, you can start a free trial at cloud.gitea.com.

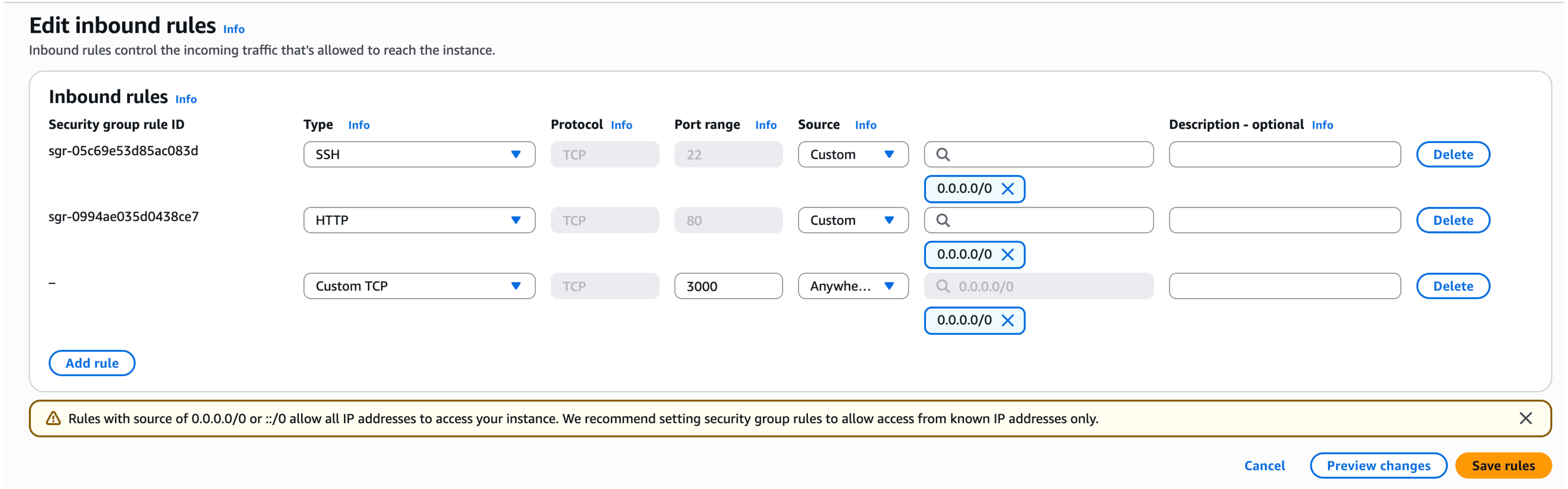
# Plan For Today

- Set up an EC2 Instance
  - Running Ubuntu
- Create a Gitea Service User
- Set up directories for new user
- Install Gitea
- Set up a Systemd service
- Install a database
- Create a git repo on our personal service!

# Create an EC2 Instance

- Give it a name
- Choose Ubuntu 24.04 LTS (HVM), SSD Volume Type
  - Free-tier eligible!
- Choose the t3.micro instance type
- Create a new key pair (or use one if you already have one)
  - Give it a name,
  - Choose ED25519
  - Use .pem from Linux/Mac
- Set up a VPC (with defaults) if you need to.
- Set up 1 20GB gp3 Root volume.
- Click “Launch Instance”

# Update Security Group Inbound Rules



## Edit inbound rules

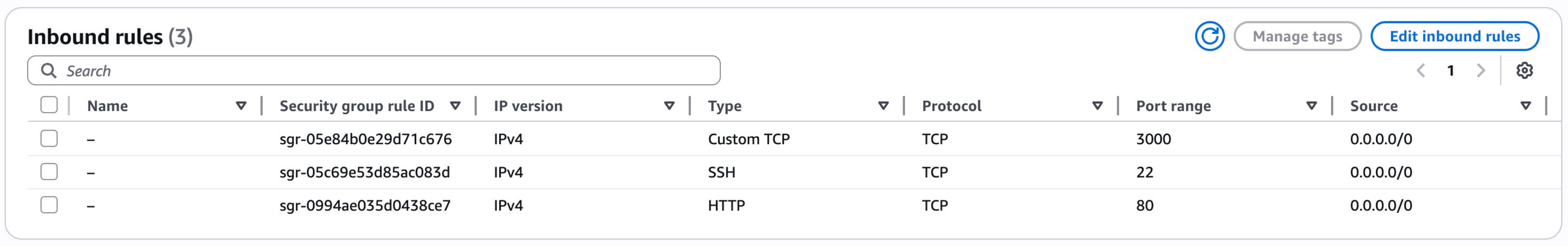
Inbound rules control the incoming traffic that's allowed to reach the instance.

### Inbound rules

| Security group rule ID | Type | Protocol | Port range | Source | | Description - optional |
|---|---|---|---|---|---|---|
| sgr-05c69e53d85ac083d | SSH | TCP | 22 | Custom | 0.0.0.0/0 | |
| sgr-0994ae035d0438ce7 | HTTP | TCP | 80 | Custom | 0.0.0.0/0 | |
| – | Custom TCP | TCP | 3000 | Anywhe... | 0.0.0.0/0 | |

⚠ Rules with source of 0.0.0.0/0 or ::/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

# Update Security Group Inbound Rules



| □ | Name | Security group rule ID | IP version | Type | Protocol | Port range | Source |
|---|---|---|---|---|---|---|---|
| □ | – | sgr-05e84b0e29d71c676 | IPv4 | Custom TCP | TCP | 3000 | 0.0.0.0/0 |
| □ | – | sgr-05c69e53d85ac083d | IPv4 | SSH | TCP | 22 | 0.0.0.0/0 |
| □ | – | sgr-0994ae035d0438ce7 | IPv4 | HTTP | TCP | 80 | 0.0.0.0/0 |

# SSH into the new instance

```
chmod 400 <pem file>
```

```
ssh –i <pem file> ubuntu@<ip address>
```

You get the public IPv4 address from the instance summary.

# Update Ubuntu

```
sudo apt update
sudo apt upgrade -y
sudo apt install -y git sqlite3 curl ca-certificates
```

# Create Gitea service user with real home

```
sudo adduser \
  --system \
  --shell /bin/bash \
  --home /home/git \
  --gecos 'Git Version Control' \
  --group \
  git
```

# Create directories

```
sudo mkdir -p /var/lib/gitea/{custom,data,log}
sudo mkdir -p /etc/gitea

sudo chown -R git:git /var/lib/gitea
sudo chown -R root:git /etc/gitea
sudo chmod 770 /etc/gitea
```

# Install Gitea binary

```
curl -L https://dl.gitea.com/gitea/1.22.3/gitea-1.22.3-linux-amd64 \
 | sudo tee /usr/local/bin/gitea > /dev/null
```

```
sudo chmod +x /usr/local/bin/gitea
```

# Installing emacs

```
sudo apt install emacs
```

# systemd service

```
sudo emacs /etc/systemd/system/gitea.service

[Unit]
Description=Gitea
After=network.target

[Service]
Type=simple
User=git
Group=git
WorkingDirectory=/var/lib/gitea
ExecStart=/usr/local/bin/gitea web --config /etc/gitea/app.ini
Restart=always
Environment=USER=git HOME=/home/git GITEA_WORK_DIR=/var/lib/gitea

[Install]
WantedBy=multi-user.target
```

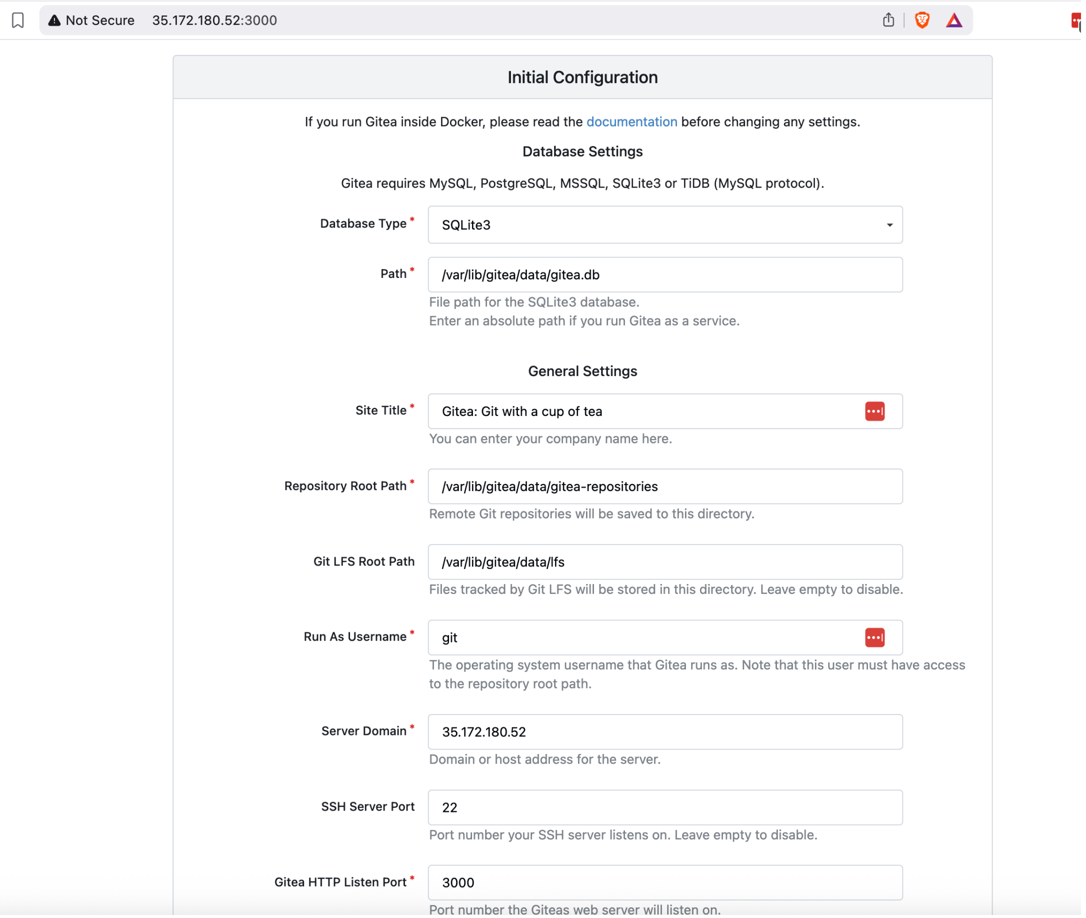
# Enable and start service

```
sudo systemctl daemon-reload
sudo systemctl enable gitea
sudo systemctl start gitea
```

# Debugging

```
sudo journalctl -u gitea -n 200 --no-pager
```

# Set up Database



Not Secure 35.172.180.52:3000

Initial Configuration

If you run Gitea inside Docker, please read the documentation before changing any settings.

Database Settings

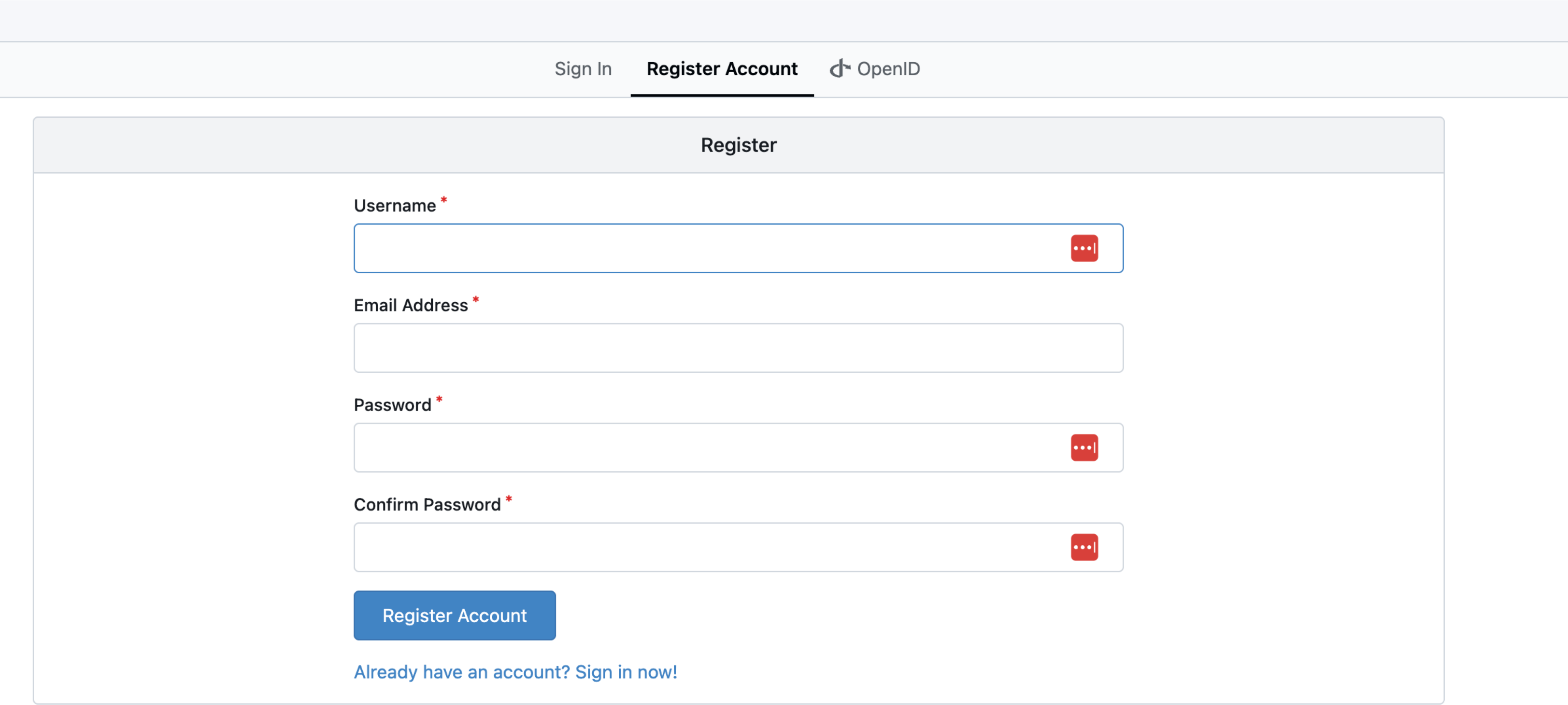
Gitea requires MySQL, PostgreSQL, MSSQL, SQLite3 or TiDB (MySQL protocol).

| | |
|---|---|
| Database Type * | SQLite3 |
| Path * | /var/lib/gitea/data/gitea.db<br>File path for the SQLite3 database.<br>Enter an absolute path if you run Gitea as a service. |

General Settings

| | |
|---|---|
| Site Title * | Gitea: Git with a cup of tea<br>You can enter your company name here. |
| Repository Root Path * | /var/lib/gitea/data/gitea-repositories<br>Remote Git repositories will be saved to this directory. |
| Git LFS Root Path | /var/lib/gitea/data/lfs<br>Files tracked by Git LFS will be stored in this directory. Leave empty to disable. |
| Run As Username * | git<br>The operating system username that Gitea runs as. Note that this user must have access to the repository root path. |
| Server Domain * | 35.172.180.52<br>Domain or host address for the server. |
| SSH Server Port | 22<br>Port number your SSH server listens on. Leave empty to disable. |
| Gitea HTTP Listen Port * | 3000<br>Port number the Giteas web server will listen on. |

# Register a new user



Register

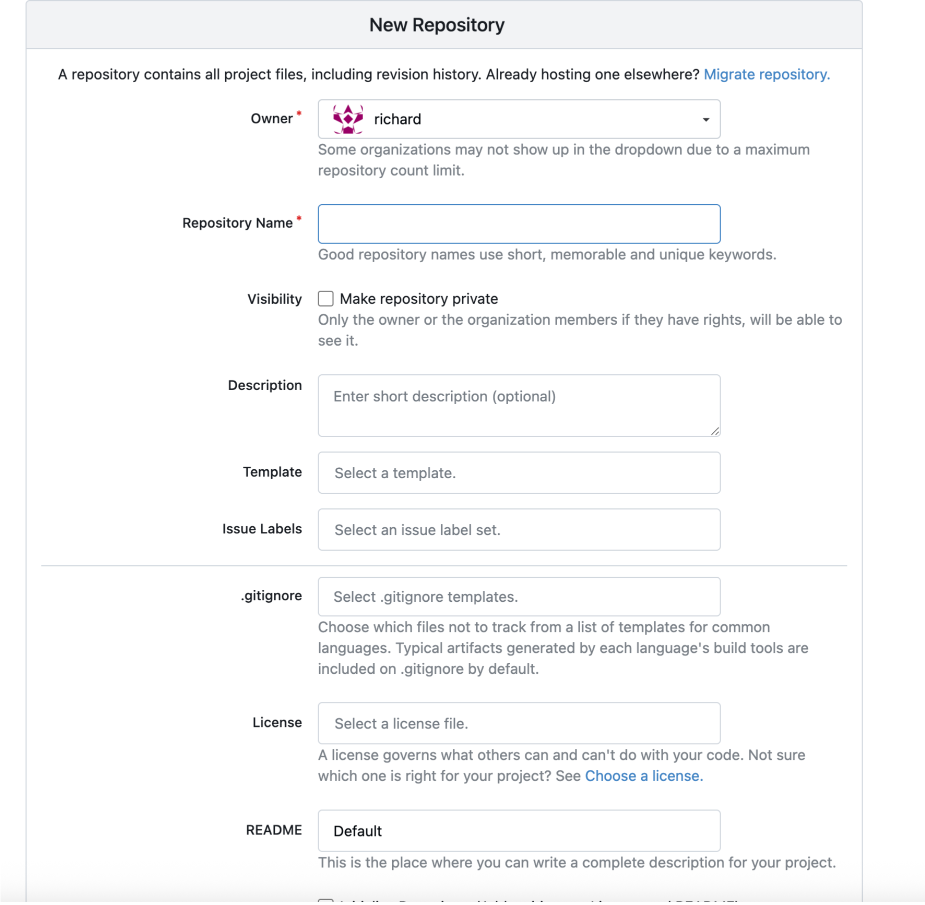
Username *

Email Address *

Password *

Confirm Password *

# Create a repository



New Repository

A repository contains all project files, including revision history. Already hosting one elsewhere? Migrate repository.

Owner * richard

Some organizations may not show up in the dropdown due to a maximum repository count limit.

Repository Name *

Good repository names use short, memorable and unique keywords.

Visibility □ Make repository private

Only the owner or the organization members if they have rights, will be able to see it.

Description: Enter short description (optional)

Template: Select a template.

Issue Labels: Select an issue label set.

.gitignore: Select .gitignore templates.

Choose which files not to track from a list of templates for common languages. Typical artifacts generated by each language's build tools are included on .gitignore by default.

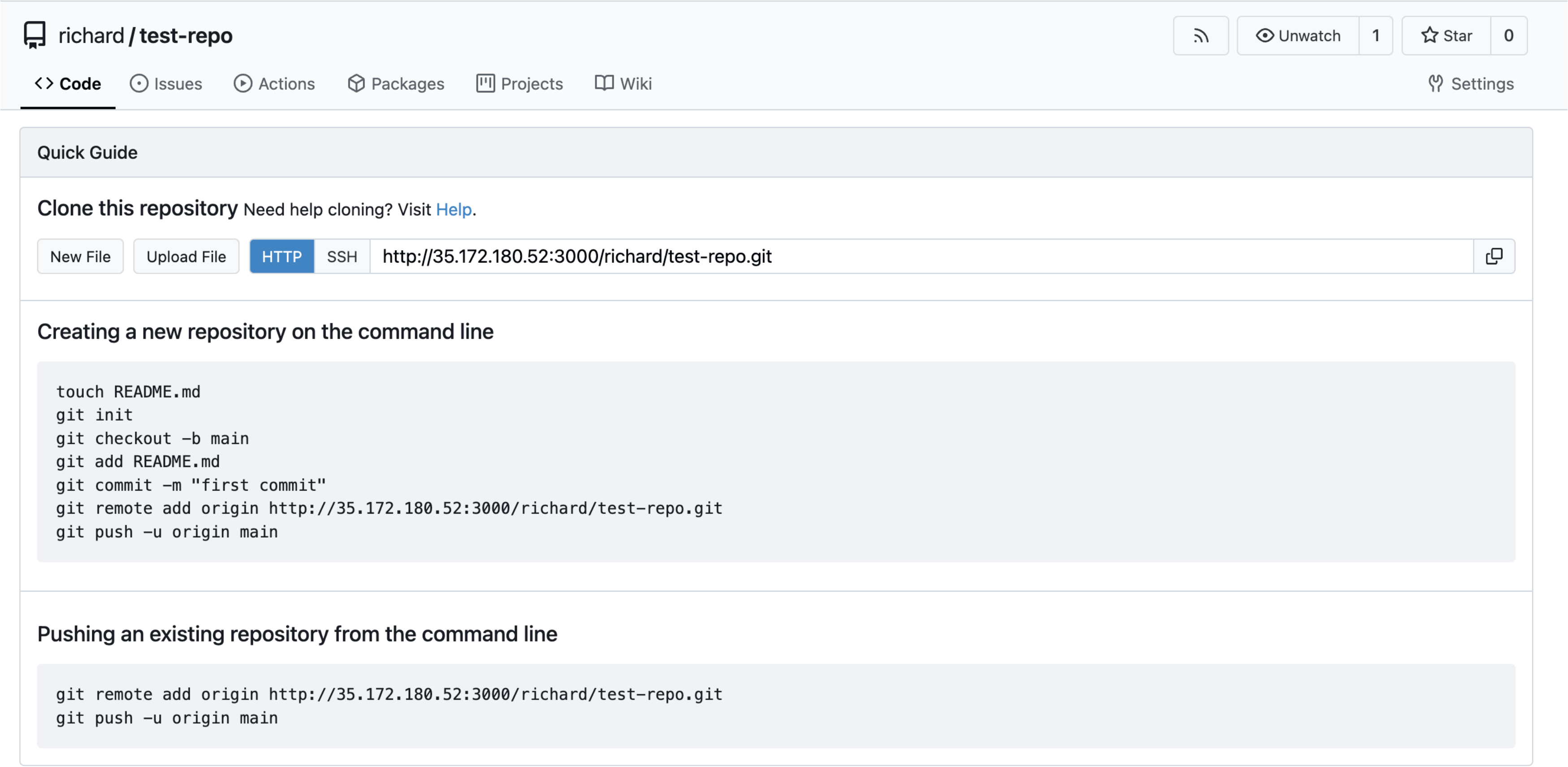
License: Select a license file.

A license governs what others can and can't do with your code. Not sure which one is right for your project? See Choose a license.

README: Default

This is the place where you can write a complete description for your project.

# Create a repository



Quick Guide

**Clone this repository** Need help cloning? Visit Help.

http://35.172.180.52:3000/richard/test-repo.git

**Creating a new repository on the command line**

```
touch README.md
git init
git checkout -b main
git add README.md
git commit -m "first commit"
git remote add origin http://35.172.180.52:3000/richard/test-repo.git
git push -u origin main
```

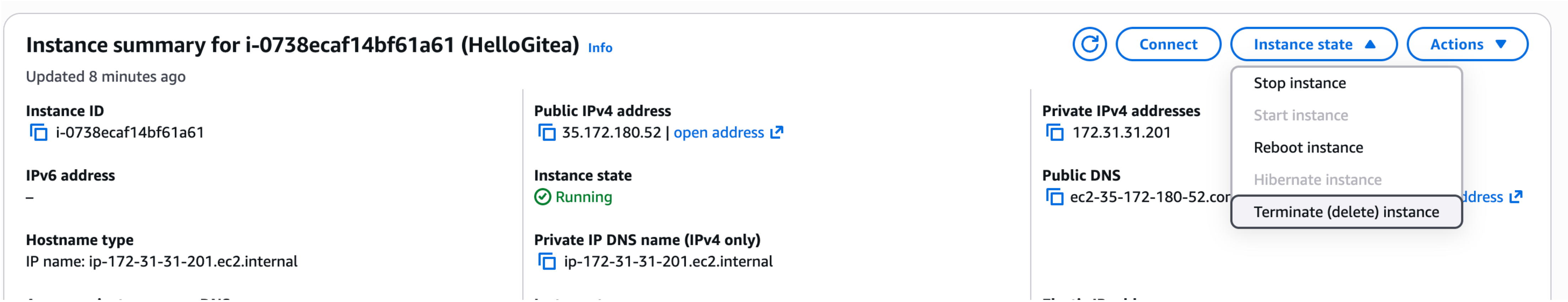
**Pushing an existing repository from the command line**

```
git remote add origin http://35.172.180.52:3000/richard/test-repo.git
git push -u origin main
```

# Lock down config after install

```
sudo chmod 750 /etc/gitea
sudo chmod 640 /etc/gitea/app.ini
sudo chown root:git /etc/gitea/app.ini
sudo systemctl restart gitea
```

# Terminate the instance (when you’re done)



Instance summary for i-0738ecaf14bf61a61 (HelloGitea)

Updated 8 minutes ago

**Instance ID**
i-0738ecaf14bf61a61

**IPv6 address**
–

**Hostname type**
IP name: ip-172-31-31-201.ec2.internal

**Public IPv4 address**
35.172.180.52 | open address

**Instance state**
Running

**Private IP DNS name (IPv4 only)**
ip-172-31-31-201.ec2.internal

**Private IPv4 addresses**
172.31.31.201

**Public DNS**
ec2-35-172-180-52.cor

# Going Beyond the Basics

- Next steps to get a GitHub replacement:
    - Put it behind HTTPS
    - Fix the public URL permanently
    - Enable backups
        - SQLite DB
        - Repositories
        - Config
    - Decide on Git auth method
    - Configure email
    - Lock down the instance
        - Restrict ports, force HTTPS, disable registration

# Programs to know

- chmod
  - Change the permissions of a file or directory
- ssh
  - Log in to a remote machine.
- sudo
  - Run a command as the administrative user
- apt
  - Install, update, and remove software (binary packages)
- adduser
  - Create a new user and/or group.
- mkdir
  - Create a new directory
- chown
  - Change the owner and group of a file or directory
- curl
  - Download a file from a remote source.

# Programs to know

- tee
  - Split
- emacs
  - Text editor. Very powerful.
- systemctl
  - Control operating system services and long-running processes.
- journalctl
  - Inspect and analyze log files collected by systemd-journald.