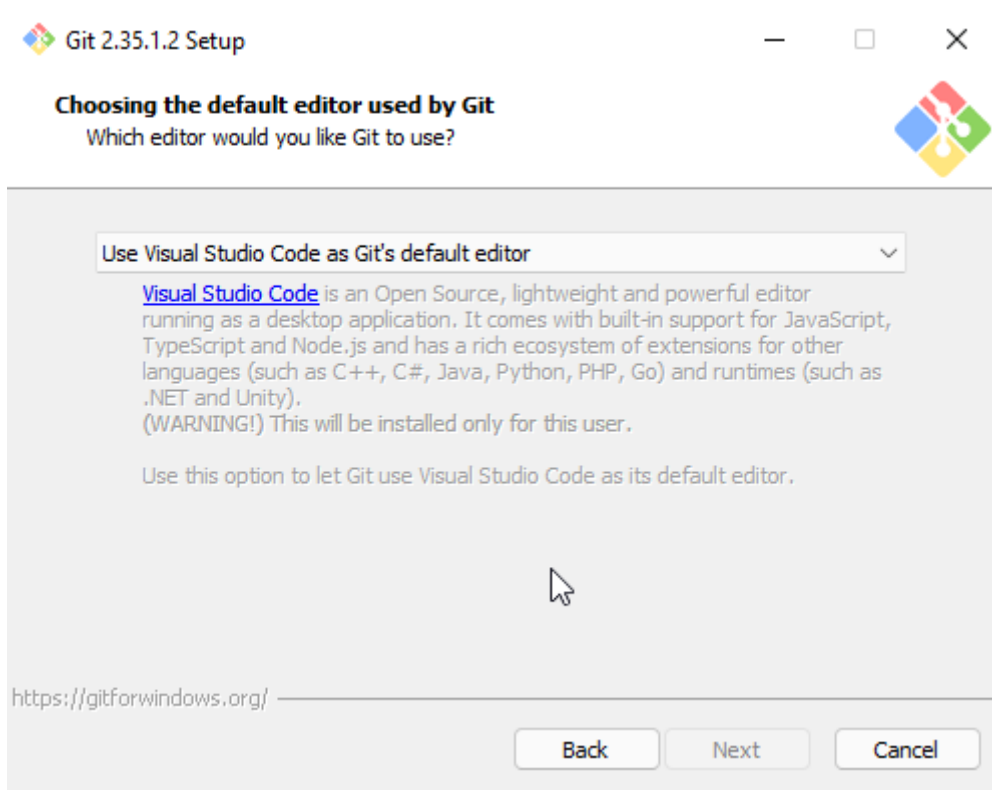


# Configure Workspace

## Install git for windows (Git Bash)

<https://gitforwindows.org/>

You can use everywhere the default settings except for the default editor, select there "Use Visual Studio code as Git's default editor"



## Generate an ssh-key

1. Open "Git Bash" and type the following command to create an SSH-Key

```
ssh-keygen.exe -t <type> -b <bit ammount> -c <Your identifying comment>

# For example

ssh-keygen.exe -t ed25519 -b 4096 -C "janic @ windows pc"
```

2. Now you are prompted to enter the path to the key. You can use the default. (just press enter and do not enter anything)

3. Now you are prompted to enter a password for your key. The default is no password. (just press enter and do not enter anything)

4. Now you are prompted to confirm your password, if you have entered none, just hit enter again.

```

MINGW64:/c/Users/Janic Voser
Janic Voser@win11 MINGW64 ~
$ ssh-keygen.exe -t ed25519 -b 4096 -C "janic @ windows vm"
Generating public/private ed25519 key pair.
Enter file in which to save the key (/c/Users/Janic Voser/.ssh/id_ed25519):
Created directory '/c/Users/Janic Voser/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/Janic Voser/.ssh/id_ed25519
Your public key has been saved in /c/Users/Janic Voser/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:Xwo4zxcgVZITg1XxfzPTaF1yrT05e5JxrfbzLE2znons janic @ windows vm
The key's randomart image is:
+---[ED25519 256]---+
|    ...*=    +. . |
|    . ...o..  =.o |
|    .  .. . o +. |
|    o    = .o. |
|    + S    .+o.o |
|    * o o. +++ |
|    . o o  ..B+ |
|    . Eo* |
|    o=.*= |
+-----[SHA256]-----+
Janic Voser@win11 MINGW64 ~
$ |

```

# Always use your custom ssh-key

1. Navigate to your home directory with the comand `cd`

2. Navigate in your ssh config directory with the command `cd .ssh`

3. List the content of this directory with the command `ls` you should now see you custom key in here.

4. Create a config file which defines to always use your custom ssh-key you can use the command below

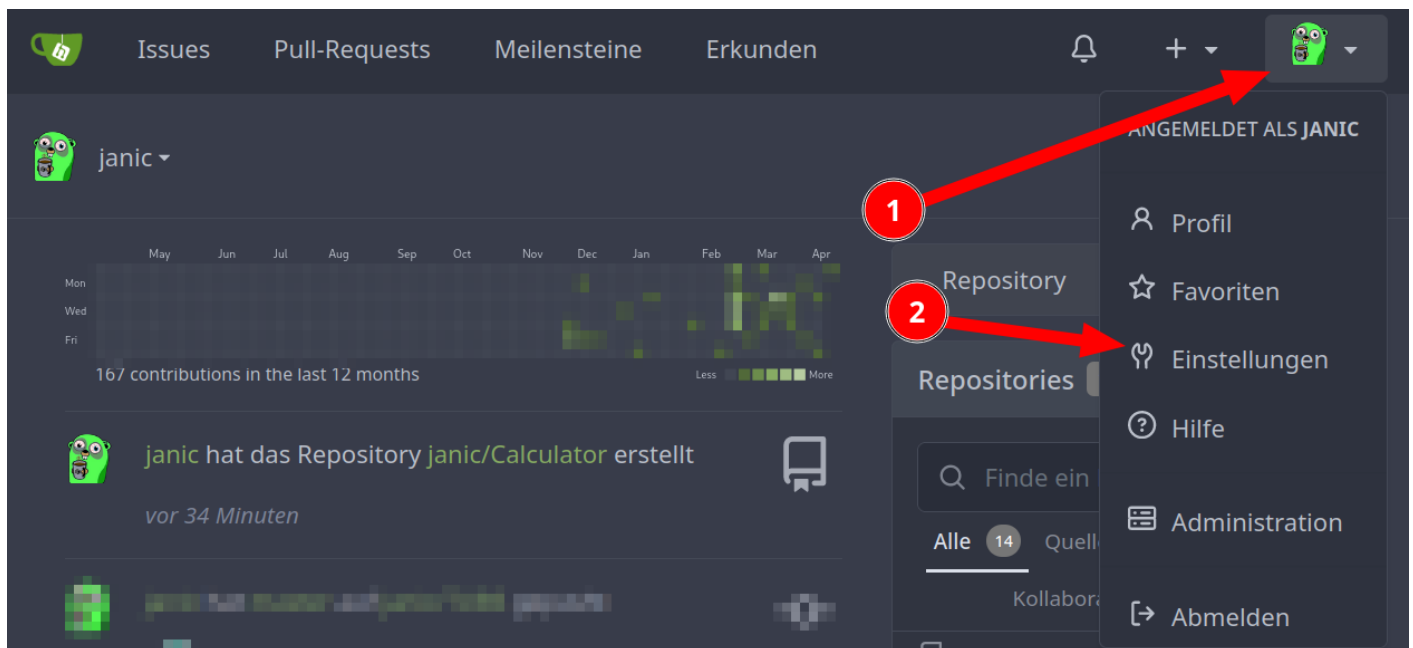
```
echo "IdentityFile ~/.ssh/<your ssh key>" > config
# For example:
```

```
echo "IdentityFile ~/.ssh/id_ed25519" > config
```

```
Janic Voser@win11 MINGW64 ~  
$ cd .ssh  
  
Janic Voser@win11 MINGW64 ~/.ssh  
$ ls  
id_ed25519  id_ed25519.pub  
  
Janic Voser@win11 MINGW64 ~/.ssh  
$ echo "IdentityFile ~/.ssh/id_ed25519" > config  
  
Janic Voser@win11 MINGW64 ~/.ssh
```

# Add your Public SSH-Key to Gitea

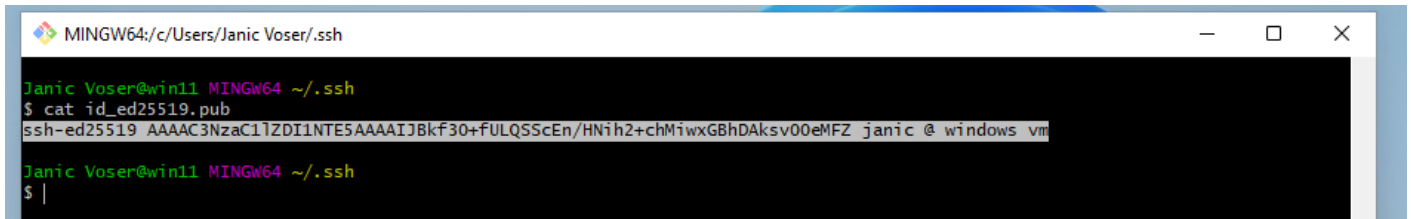
1. Login to your gitea account
2. Click in the top right on your profile and there on "Settings"



3. Get the content of your public key

You can use the cat command to open the content of the public key.

```
cat <your public key>  
  
# For example:  
cat id_ed25519.pub
```

A terminal window titled 'MINGW64:/c/Users/Janic Voser/.ssh' with standard window controls. The prompt is 'Janic Voser@win11 MINGW64 ~/.ssh'. The user enters '\$ cat id\_ed25519.pub', and the terminal displays the public key content: 'ssh-ed25519 AAAAC3NzaC11ZDI1NTE5AAAAIJBkf30+fULQSScEn/HNih2+chMiwxGBhDAksv00eMFZ janic @ windows vm'. The prompt returns to '\$ |'.

4. Follow the steps below

Go to "SSH- / GPG-Key"

Click on "add key"

Enter the name of this key

Enter the content of the public key

Click on "Add Key"

The screenshot shows the GitHub 'SSH- / GPG-Schlüssel' page. Red arrows and numbers indicate the following steps:

1. Click on the 'SSH- / GPG-Schlüssel' tab in the top navigation bar.
2. Click on the 'Schlüssel hinzufügen' button in the top right corner of the 'SSH-Schlüssel verwalten' section.
3. Enter a key name in the 'Schlüsselname' field.
4. Paste the SSH key into the 'Inhalt' field.
5. Click on the 'Schlüssel hinzufügen' button at the bottom left of the form.

Below the form, there is a list of existing SSH keys, each with 'Entfernen' and 'Verifizieren' buttons. At the bottom, there is a section for 'GPG-Schlüssel verwalten' with a 'Schlüssel hinzufügen' button and a link to GitHub's help page for GPG.

# Configure Git

You need to set a default username and email for git. You can do this with the following commands

```
git config --global user.name "Name"
git config --global user.email "e@mail.com"
#For example:
git config --global user.name "Janic Voser"
```

```
git config --global user.email "janic@voser.cloud"
```

```
Janic Voser@win11 MINGW64 ~/.ssh
$ git config --global user.name "Janic Voser"

Janic Voser@win11 MINGW64 ~/.ssh
$ git config --global user.email "janic@voser.cloud"

Janic Voser@win11 MINGW64 ~/.ssh
$ |
```

---

Revision #2

Created 10 April 2022 10:00:50 by Janic Voser

Updated 10 April 2022 10:56:55 by Janic Voser