

Quick guide to Git & GitHub

Agenda

- Git first steps
- GitHub first steps
- Your SWP

What is Git?

- Git = tool for software version control
- Changes to a code base are applied via «commits» (= patches)
- Each contributor/participant has its own repository to which the commits are applied

Git Setup Step 1: Installation

- Linux: package management of your choice
 - with Debian for example `sudo apt-get install git`
- Mac OS X: DMG from <http://git-scm.com/download/mac>
- Windows: EXE from <http://git-scm.com/download/win>
- There are also GUIs
- TortoiseGit (Windows-only)
<http://code.google.com/p/tortoisegit/wiki/Download>
- Eclipse with Egit
- ... there are others...

Git Setup Step 2: Configuration

- Name and e-mail address (ID for commits)

```
git config --global user.name "Your name"
```

```
git config --global user.email "you@e-mail.de"
```

Git Setup Step 3: Cloning

- Cloning: set up a local copy of the repository

```
git clone https://github.com/RIOT-OS/RIOT.git
```

Committing Code Changes with Git

- Changes committed are based on so-called Git Index
- Index: data structure tracking files that Git is managing
- Updating the index to track your recent changes to a file:

```
git add <file>
```

or you can also use

```
git add -p <file>
```

- Commit the changes to all the files managed by Git:

```
git commit -am "<commit message>"
```

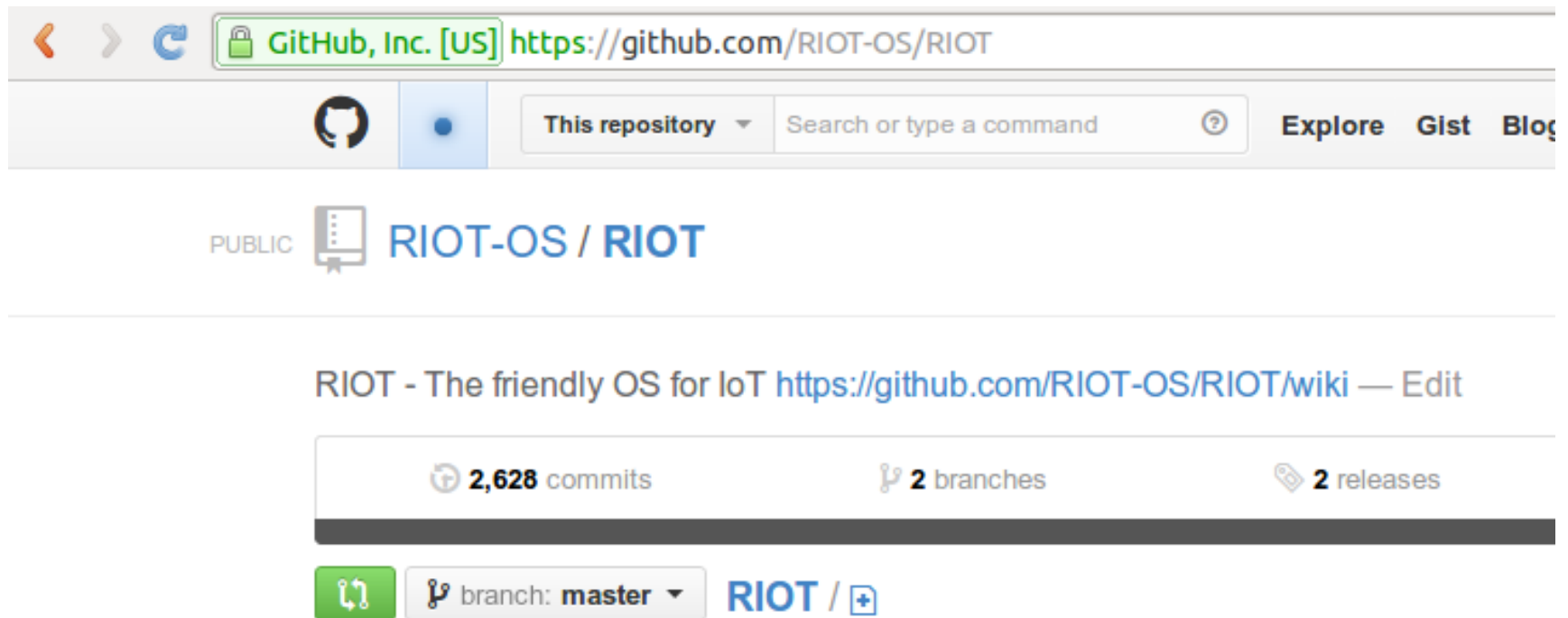
- Commit message should concisely describe the change(s)

Agenda

- Git first steps
- GitHub first steps
- Your SWP

What is GitHub?

- Web interface for Git, and for
 - project management
 - social network for developer
 - RIOT on GitHub: <https://github.com/RIOT-OS/RIOT/>



GitHub Setup Step 1: Account

- Create an account on <https://www.github.com>

GitHub Setup Step 2: generate SSH Key

- In Windows with PuTTYgen:
<http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html>
- In other environments:

ssh-keygen

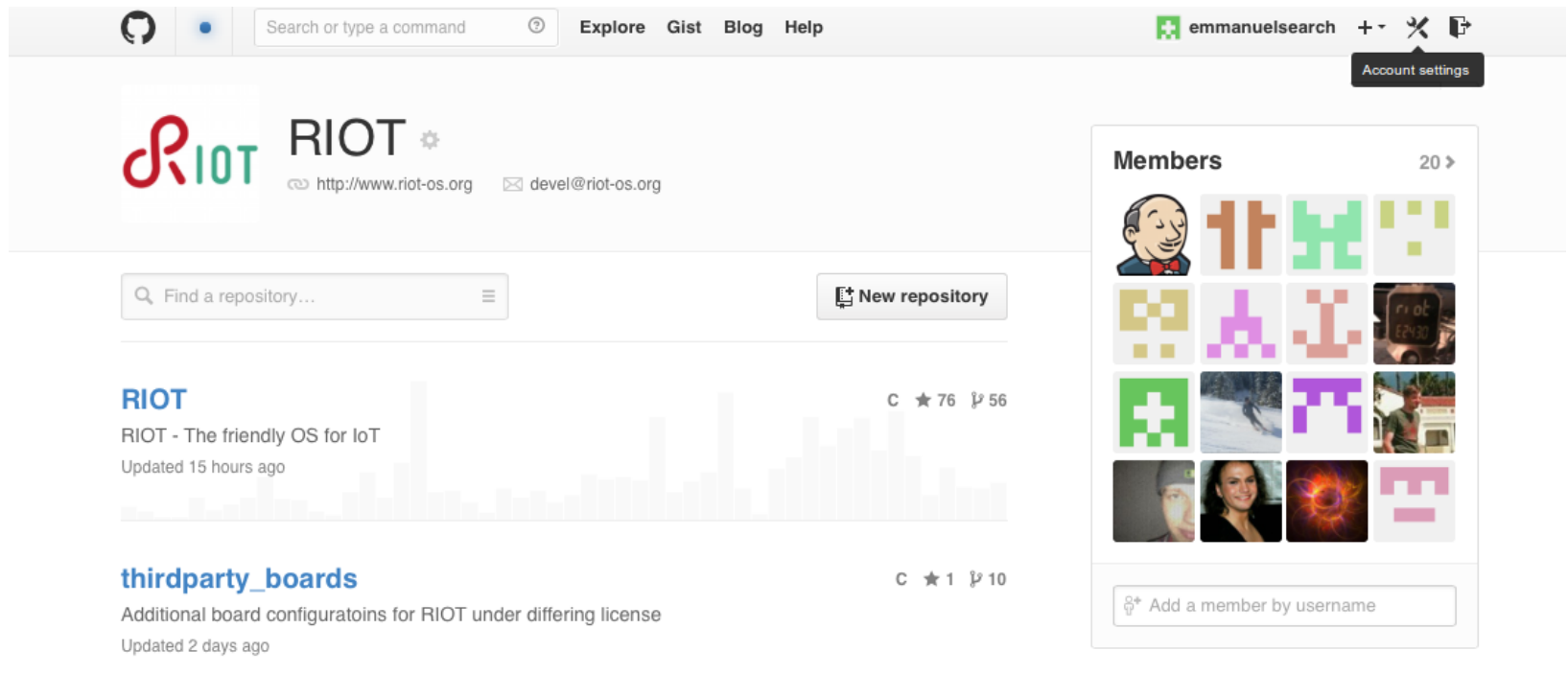
- Choose password, other parameters can stay default
- Copy generated public key

```
cat ~/.ssh/id_rsa.pub
```

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDAz4zddu7La6++ytxDD2gNIz0  
y0FkrnaqxcKiym0sGkNz+4+CoERmgH6NW5TwIopEGW0lwqZQUj9kz8zEh6QBzdr  
Oh0DhJ68L4okME3++FOSRStkSZschbm2XxYAxKbUktEEyMQ9YnQICAQUOYozyOX  
JJuzii6yITTXEbeQZMQEQc+bVe4e+BQ7QcvZ78GbWkVk/eqSC3KrIyJYPoGEMKV  
rTfba49DNt9tgArSuhxrF1vwhRSkOa2LjOg3xTokOyA4vTPvFUEL/757Si0PhHd  
UycAajjstcA9Yylm9nIu2IKNeKR4Uu+CEjXDnznckTED1zYL+cwNAwWVFCxN0fT  
xh YourName@somewhere
```

GitHub Setup Step 3: Set up Public Key

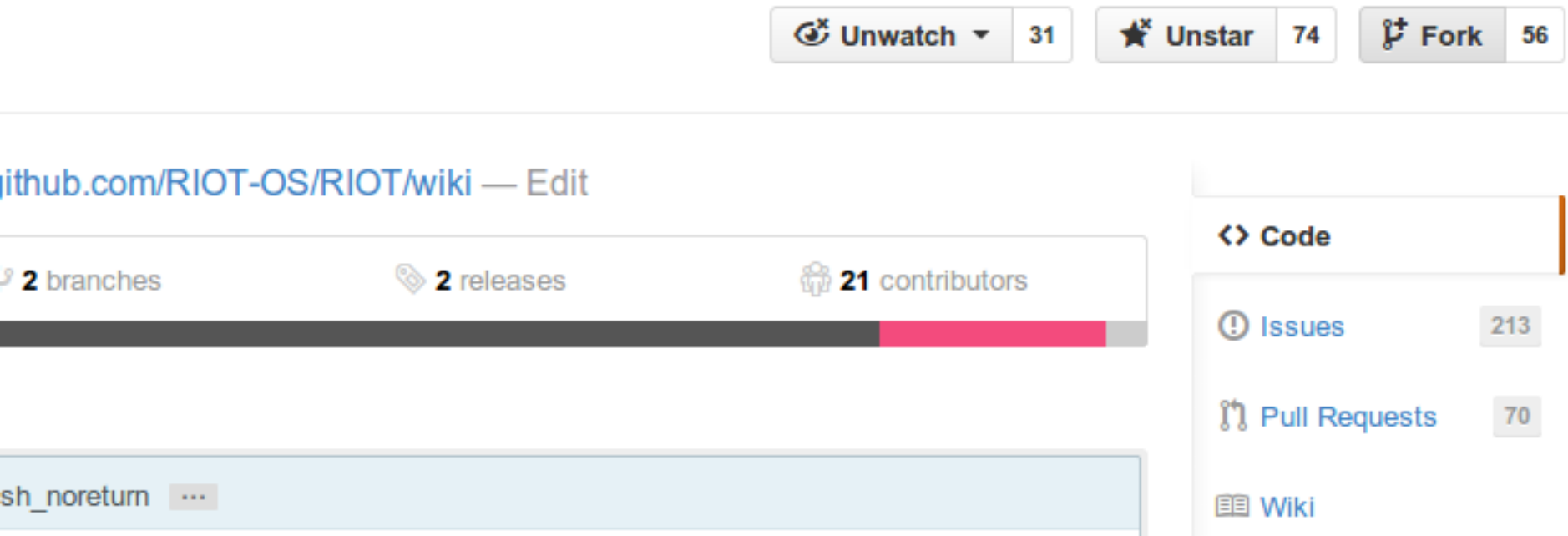
- Click on "Account settings" (upper-right corner)



- In "SSH keys" -> "Add SSH key":
 - title: e.g. your name
 - key: here copy your public key

GitHub Setup Step 4: Fork

- Fork: set up a copy of the repo in Github
- Prerequisite: Github account & being logged in
- Click on "Fork" button



Pushing Changes to your Fork

- Set up a local repository forking your GitHub repo

```
git remote add fork git@github.com:username>/RIOT-OS
```

- Edit your files, commit your changes locally
- Then push local commits to GitHub

```
git push -u fork master
```


- Remark: **always push to your own fork only!**
- From now on, you can simply use:

```
git push
```

Creating a Pull Request (PR)

- Pull Request: propose your changes upstream, from your fork to the upstream branch
- Click “Compare, review, create a pull request”
 - Choose appropriate branch
 - Click on “create pull request”


RIOT OS - The revolutionary IoT OS — Edit

 2,446 commits

 22 branches

 0 releases



 branch: master ▾

RIOT / 

Compare, review, create a pull request

This branch is 0 commits ahead and 0 commits behind master

More Info on Git and GitHub

- We'll see some more details as we go
- In the mean time:
 - Book on Git: <http://git-scm.com/book/de>
 - Github online help: <https://help.github.com>

Agenda

- Git
- GitHub
- Your SWP

Your Software Project

- By default base yourself on this fork:
<https://github.com/emmanuelsearch/RIOT>

(this will simplify your life because you won't have to rebase on the state of the master branch)

Any questions about this: ask us