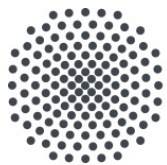


# GitLab Instructions for Students



University of Stuttgart

**Brian Setz, Robin Pesl**

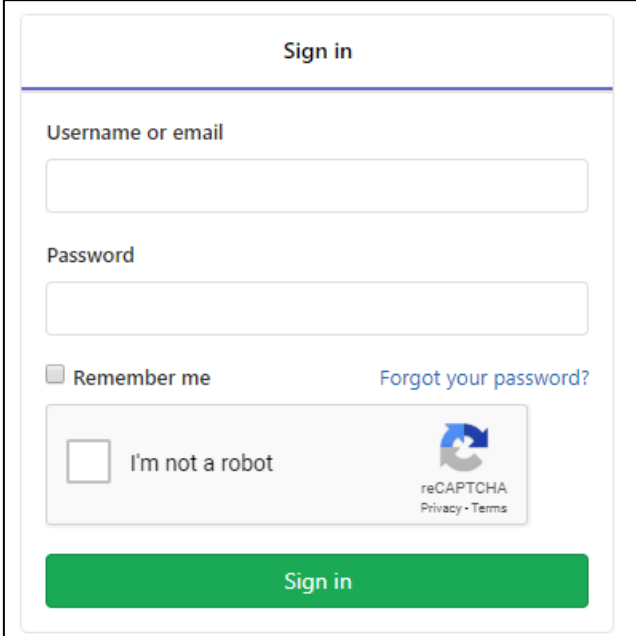
[robin.pesl@iaas.uni-stuttgart.de](mailto:robin.pesl@iaas.uni-stuttgart.de)

Service Computing Department

Institute of Architecture of Application Systems

# Organisation of Student Repositories

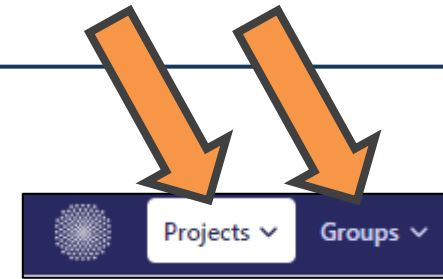
- GitLab <https://gitlab.svccomp.de/>
  - Web-based Git repository manager (like GitHub)
    - Git → distributed version control system
    - Documentation: <https://gitlab.svccomp.de/help/user/index.md>
  - Robin Pesl will create an account for you
    - Send your full name, student email, project type, and title
    - Follow the email instructions to set your password
- Login with your username / email and password once you have configured your password
- Fill in the reCAPTCHA
- Complete the login process



The image shows a screenshot of a web-based sign-in form. At the top, it says "Sign in". Below that, there are two input fields: "Username or email" and "Password". Under the "Password" field, there is a checkbox labeled "Remember me" and a link "Forgot your password?". At the bottom of the form, there is a reCAPTCHA widget with a checkbox and the text "I'm not a robot", and a reCAPTCHA logo with the text "reCAPTCHA Privacy - Terms". A green "Sign in" button is located at the bottom right of the form.

# GitLab Usage

- Browse all groups assigned to you
  - <https://gitlab.svccomp.de/dashboard/groups>
- Browse all repositories assigned to you
  - <https://gitlab.svccomp.de/dashboard/projects>
- Setup your SSH keys
  - <https://gitlab.svccomp.de/profile/keys>
  - Generate SSH keys:
    - <https://gitlab.svccomp.de/help/ssh/README#generating-a-new-ssh-key-pair>
- How to use git
  - <https://rogerdudler.github.io/git-guide/>
  - <https://product.hubspot.com/blog/git-and-github-tutorial-for-beginners>
  - <https://git-scm.com/docs/gittutorial>



# Branches

---

- Master branch should contain only the **final version** of your work
- By default the **master branch is protected**
  - This means you cannot commit to it
- Create a **develop branch**
  - `$ git checkout -b develop`
  - `$ git push -u origin develop`
- Add your commits to the develop branch and push them
- Modifications to this branching model may be made by your supervisor

# Final Version

- Before finishing your project or thesis, the **final version has to be placed in the master branch**. This is mandatory

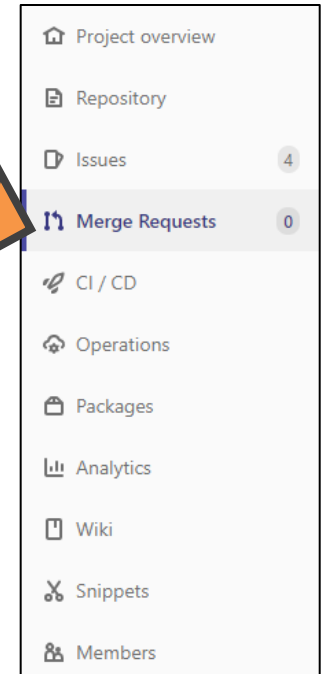
- Create a Merge Request

- Source: develop
- Target: master

New Merge Request

Source branch: svccomp-students/entwicklu... develop

Target branch: svccomp-students/entwicklu... master



- Assign your **supervisor** and **Robin Pesl** to the merge request

Assignee: Unassigned

- Finalize the merge request