Swarmlab quick-start guide!

Welcome to the vast world of Swarmlab!

Swarmlab is a powerful collaboration tool that connects tutors and students in a unique all-in-one environment that gets work done fast and easily.

In other words Swarmlab is the way to make your learning fun while at the same time getting to your goals faster!

But let's get you started...

Note: These instructions are general targeted at both students and tutors.

If you wish to skip to the specific instructions for tutors (lab creation, student invitation etc) click here.

⚠️ "General info WILL NOT be repeated so if you have basic questions regarding registration or labroom usage please read on from here.

1. The first steps

Before you have all of the powerfull tools of Swarmalb at your grasp you will have to create an account.

To do that navigate to the home page at https://api-client.swarmlab.io:8088

You should have been greeted by our login screen!

[login] | *images/registration/login.png*

To register click on the 'login with sso' option and accept the terms.

[terms] | *images/registration/terms.png*

Now you should be redirected to the login screen and you should be able to use the 'register' option at the bottom to make a new account!

[login main] | *images/registration/login_main.png*

Now fill in the required details

[register details] | *images/registration/register_details.png*

and follow the remaining instructions to activate your account.

Now that your acount is ready, lets get to know Swarmlabs main page.

2. Main Menou

2.1. The main page

Once loaded the main page should look like this:

[main page] | *images/run/main_page.png*

On the left we can see that there are two areas inside swarmlab:

1. the dashboard where our services live

2. the 'mylab' tab from which we can control our labrooms.

The dashboard page is pretty straight forward:

1. The left list shows all the available services and allows you to run them.

2. The other shows the available lab rooms and allows you to interact with them (p.e. subscribe).

With that out of the way lets enter the heart of our platform, 'my lab'!

2.2. My Lab

Nested under 'mylab' are both creation tools (bootstrap) and execution tools (run).

2.3. The run menu

Inside the run menu there are three areas:

1. 'Rub lab instance'

2. 'Enter lab room'

3. the console

As the names suggest we can use the first to **run** one of our available lab instances, the second to **enter** a created labroom, and the third to **execute** scripts.
Lets look at the options one by one.
For the purpose of this walkthrough 'testmpi2' and 'test 18' were created.
Since the console is explained along with the first area we dont feel there is a need to have seperate walkthrough.

2.3.1. Run lab instance area

Creating a lab

To create a lab instance simply add a repo to your git!
You will then be able to locate that repo inside the instance list and run it.
For more info on this please refer to the Bootstrap chapter here !

Running a lab instance

First we have to select the lab we want to run. To find a specific lab we can use the search bar at the top.

[lab instance list] | *images/run/lab_instance_list.png*

After we have located the desired lab we can run it by using the cloud icon. [cloud icon]

To request more info about the lab we use the info icon [info icon] and the trashcan [trash icon]

deletes the lab (forever).
Carefull, forever is a long time!

So lets go back to running the lab. After clicking on the according icon we should be redirected to the execution screen.

[execution screen] | *images/run/execution_screen.png*

Here we can use the command line on the bottom to execute the required commands. Our results will be visible in the console above.

[ls results console] | *images/run/ls_results_console.png*

Note that you can choose to use the editor of your liking from the drop down menu.

For more complicated tasks that may require responsivines from the machine you can use the terminal option to open a full-blown unix command line.

2.3.2. Enter lab room area

This is the area where you will find the available labrooms created by tutors and join the desired one.

Browsing labrooms

You can easily browse throught the available labrooms in the list using the search bar at the top.

[lab room list] | *images/run/lab_room_list.png*

You can get more info about a specific labroom by clicking the info button [info icon].

[info poppup] | *images/run/info_poppup.png*

Active labrooms are indicated using a green color while closed ones with orange.

Entering a labroom

Once you have found the labroom you were looking for you can enter by simply clicking on the cloud icon [cloud icon].

You should see a screen as follows:

[password big] | *images/run/password_big.png*

The passwords are randomly generated and you can use them to connect to your lesson (see picture bellow).

[password user] | *images/run/password_user.png*

If the tutor is sharing his screen you can watch by clicking on the according button and using the respective password.

[password tutor screen] | *images/run/password_tutor_screen.png*

Now, under the password area there are three other options:

'Meeting' [meeting button]

> Here you can join a meeting with your group (this will redirect you to a jitsi implementation).
> For more visit https://jitsi.org/

'Gui' [gui button]

> This loads a graphic unix environment.

'Console' [console button]

> This loads a unix terminal.

By now it should propably be clear how to run a lab from the list or enter a labroom.
But you are propably wondering 'My list is empty how am i supposed to run labs from it?!'
Well here is where the final but most basic menu comes into play: **'Bootstrap'** !

2.4. Bootstrap

choose myLAB → create

[bootstrap main] | *images/bootstrap/bootstrap_main.png*

Here we can create and add labs to our list. To do so we have to go through a series of steps, but lets first understand some basic things about how swarmlab works...

> Every swarmlab lab instance resides in a git repo created inside our git and owned by the lab creator. This repo then serves as a 'home' for all the work inside that lab.
> So for us to create and run a lab we have to link a service to a git repo.

Asuming we have an account (using the same email as our swarmlab account!) all our repos should be shown on the list in the top left corner.
To search for any public repo we can use the search bar.

[test search repo] | *images/bootstrap/test_search_repo.png*

From that list we can now select the desired repository that we want to use by clicking on the plug icon. [plug icon]
We can use any repo we like, but we will be able to commit only to those that we have access to.
If we wish to edit or contribute to the development of a project we are not part of we can use the clone option as usual!

Now we have to select the service that our lab will be using. We just select the desired one from the list on the right and it will be automatically linked to the project.

[service select] | *images/bootstrap/service_select.png*

Finally we have to save the lab instance by filling in the required info at the bottom.

[save lab details] | *images/bootstrap/save_lab_details.png*

3. Tutors Menou

Tutors have access to an extended 'my lab/bootstap' page where they can create labrooms from existing lab instances.

To do so we have to scroll all the way down whilst on the Bootstrap page.
There we should find two menus, 'My labrooms' and 'Bootstrap Labroom'

3.1. 'My labrooms'

[my labrooms] | *images/for_tutors/my_labrooms.png*

Here we can see a list of all the labrooms programmed to run today. Again like in the run tab we can distinguish open from closed ones by color (green/orange).

[todays labrooms] | *images/for_tutors/todays_labrooms.png*

We can also get info using the info icon [info icon] or delete the labroom using the trashcan icon [trash icon] .
To search for a specific labroom we can use the searchbar at the top.

> **!** The delete command is **FORCED** even if the labroom is running!!!

3.2. 'Bootstrap Labroom'

> **🔥** **choose** myLAB → Bootstrap

Now lets see how we can bootstrap a new labroom using one of our lab instances.
To do that lets navigate to the respective tab that should look like this:

[bootstrap labroom] | *images/for_tutors/bootstrap_labroom.png*

Here we can select the desired (existing!) lab instance that will be used to create the labroom using the plug icon [plug icon].

[lab instance choose] | *images/for_tutors/lab_instance_choose.png*

Then we have to specify starting date and time and also set the duration for which our lab room will be open.

[timeframe selection] | *images/for_tutors/timeframe_selection.png*

> **🔥** You can only start a labroom every whole or half hour (p.e. 12.00 or 14.30) and never inbetween so please plan ahead!!

Also note that the duration cannot be greater than three hours since we consider that the

maximum for any type of lesson!

Now using the searchbar that follows (you HAVE to search for students for them to come up) we search and select ([plug icon]) the students that we want to invite to the lesson.
Note that only students that have subscribed to our labroom will be available for invites to avoid confusion!

[student selection] | *images/for_tutors/student_selection.png*

After selecting all the desired students we have to fill in the last details at the bottom of the page. Please give concise titles and good descriptions to your labrooms!!

[bootstrap details] | *images/for_tutors/bootstrap_details.png*

Now we shall check and upon making shure that everything is right we click the 'bootstrap your lab' button!

⚠️ Bootstraping is final and no further edits are allowed!!

4. Learning Objects Notebooks

A LO Notebook **is a reusable web-based interactive chunks of e-learning** designed to explain a stand-alone learning objective, which can be used, reused or referenced during technology supported learning.

Swarmlab.io Notebooks includes the following key characteristics:

- **Learning objects are a new way of thinking about learning content**. Traditionally, content comes in a several hour chunk. Learning objects are much smaller units of learning, typically ranging **from 2 minutes to 15 minutes.**
- Are **self-contained** – each learning object can be taken independently
- Are **reusable** – a single learning object may be used in multiple contexts for multiple purposes
- Can **be aggregated** – learning objects can be grouped into larger collections of content, including traditional course structures
- Are **tagged with metadata** – every learning object has descriptive information allowing it to be easily found by a search

Learning Object Notebooks is:

> "A digital self-contained and reusable entity, with a clear educational purpose, with at least three internal and editable components: content, learning activities and elements of context".".[1]
>
> — IEEE

> "learning object any entity, digital or non-digital, that may be used for learning, education or training".[2]
>
> — Wikipedia

4.1. Usage

LO notebooks have a wealth of different uses including as a testing ground for development work, a presentation platform, and more.

- **Ad hoc nature.** - The ad hoc nature of notebooks is excellent for trying things out

  - Designing, developing, and testing solutions to problems.

- **Ready-to-Use Services.** - Easier start. Your just start with the cleaned data, the trained model, and get right to the analysis.

  - Presenting analyses, demonstrating both the code and the output, that can be easily turned into slides/pdf.

- **Built for collaboration.** - Swarmlab.io notebooks are easy to collaborate with using version control systems (git). The YAML index file makes it extremely easy to tell where things belongs to.

  - Providing hands-on walkthroughs of new library modules, visualization techniques, and strategies for attacking existing problems. They allow someone to mostly follow along while allowing them space to try out new things right in-line.

4.2. Run Learning Object Notebook

4.2.1. Find

RUN_LLO_INSTANCE ☰

| Filter your LLO searches by tags | Filter your LLO searches by Learning goals |
| Filter your LLO searches by tags | Filter your LLO searches by Author |

1 ⌄

| Object | Search | | Go | Reset |

🗁 LearningObject ⇕	🔗 URL ⇕	🔖 Tags ⇕	*CC* ⇕	▷ Actions
Linux	https://git.swarmlab.io:3000 /labs/examples-mpi2/raw /branch/master/llo-index.yml	Linux system network admin	CC BY-SA	i ⎙
Netstat	https://git.swarmlab.io:3000 /labs/examples-mpi2/raw /branch/master/index_test.yml	Linux	CC BY-SA	i ⎙

Displaying 1 to 5 of 2 items

Dashboard

After we have located the desired LO we can

- run it by using the cloud icon. [cloud icon]
- To request more info about the lab we use the info icon [info icon]

4.2.2. Run

	/branch/master/llo-index.yml			
Netstat	https://git.swarmlab.io:3000 /labs/examples-mpi2/raw /branch/master/index_test.yml	Linux	CC BY-SA	i

Displaying 1 to 5 of 2 items

sublime ⬍	i		Give it a try

```
1  # Try!
```

Netstat

netstat is available on all Unix-like Operating Systems and also available on Windows OS as well. It is very useful in terms of network troubleshooting and performance measurement. netstat is one of the most basic network service debugging tools, telling you what ports are open and whether any programs are listening on ports.

This tool is very important and much useful for Linux network administrators as well as system administrators to monitor and troubleshoot their network-related problems and determine network traffic performance. This article shows usages of netstat command with their examples which may be useful in daily operation.

Listing all the LISTENING Ports of TCP and UDP connections

```
netstat -antlupe
```

```
(No info could be read for "-p": geteuid()=33 but
you should be root.)
Active Internet connections (servers and
established)
Proto Recv-Q Send-Q Local Address Foreign Address
State User Inode PID/Program name
```

- After we have Run the desired LO we can run "try it" by using the edit icon.

4.2.3. Try

Netstat

netstat is available on all Unix-like Operating Systems and also available on Windows OS as well. It is very useful in terms of network troubleshooting and performance measurement. netstat is one of the most basic network service debugging tools, telling you what ports are open and whether any programs are listening on ports.

This tool is very important and much useful for Linux network administrators as well as system administrators to monitor and troubleshoot their network-related problems and determine network traffic performance. This article shows usages of netstat command with their examples which may be useful in daily operation.

Listing all the LISTENING Ports of TCP and UDP connections

```
netstat -antlupe
```

```
(No info could be read for "-p": geteuid()=33 but
you should be root.)
Active Internet connections (servers and
established)
Proto Recv-Q Send-Q Local Address Foreign Address
State User Inode PID/Program name
```

```
(No info could be read for "-p":
geteuid()=33 but you should be
root.)
Active Internet connections
(servers and established)
Proto Recv-Q Send-Q Local Address
Foreign Address State User Inode
PID/Program name
tcp 0 0 127.0.0.11:38898 0.0.0.0:*
LISTEN 0 682515990 -
tcp 0 0 0.0.0.0:8088 0.0.0.0:*
LISTEN 0 682521722 -
tcp 0 0 0.0.0.0:443 0.0.0.0:*
LISTEN 0 682521724 -
tcp 0 0 10.255.1.24:443
10.255.0.17:56744 ESTABLISHED 33
722582429 -
udp 0 0 127.0.0.11:56862 0.0.0.0:*
```

sublime | i | Clear | Give it a try

```
1   netstat -antlupe
```

4.3. Create Learning Object Notebook

4.3.1. Main Options

- **"My LabLearningObjects"** - LLO Admin panel

- **"Publish LearningObjects"** - Metadata and Publish your LLO

- **"Bootstrap LearningObjects"** - Create your LLO

- **"How to Contribute"**

4.3.2. Publish

Creative Commons

Get familiar with "The Creative Commons copyright licenses". 'About the Licenses'.footnote[http://creativecommons.org/about/licenses]

Creative Commons is an international nonprofit that offers flexible copyright management tools for creative work. Many learning objects have CC licences.

Bootstrap Lab Learning Object

My LabLearningObjects | Publish LearningObject | Bootstrap LearningObject | How to Contribute

Author | Subjects covered by LLO

Lab Learning Object | Subjects covered by LLO

1 beginer, 2 intermediate, 3 advanced, 4 expert, 5 guru

Difficulty | 1 | Online

CreativeCommons | CC BY-SA

Learning Object Tags | Here you can describe your goals

Learning Goals | Here you can describe your goals

URL | url_4_llo-index | Add LLO

Here you can describe your LLO

When you are finished creating your Learning Object, Use "git push" to push commits to swarmlab.io repository

- Find the url where the yaml-index resides.
- Whithin the "Publish Learning Object" menu you can choose to publish your LLO

After that you can admin your LLO using the "My Lab Learning Object".

4.3.3. Bootstrap

Bootstrap Lab Learning Object

My LabLearningObjects   Publish LearningObject   Bootstrap LearningObject   How to Contribute

| LLO name | Lab Learning Object | | Author | Author |

Intro — Here you can describe your LLO   i

Code spec [0]

sublime   i   Give it a try   i

| Service | Swarmlab service | i | Task Name | Name | i |

| Language | Programming lan | i | Exec | Full path | i | Options | Exec Options | i |

| Mimetype | Mimetype | i | Output | console | i | output mimetype | ascii | i |

index

Learning Objects Notebooks Based on the following structure:

```
--- # Lab Learning Object
  name: "test" ①
  author: "zeus"
  intro: "intro" # intro is a file ②
  base_url: "https://git.swarmlab.io:3000/labs/examples-mpi2/raw/branch/master" ③
  0_netstat:  ④
    display_name: "netstat" ⑤
    swarmlabservice: "os2" //  ⑥
    info:
      file: "0_netstat/info" ⑦
    code:
      lang: "bash" ⑧
      exec: "/bin/bash" ⑨
      file: "0_netstat/code" ⑩
      options: "" ⑪
      mimetype: "bash" ⑫
      output:
        log: "console" ⑬
        mimetype: "bash"
    challenge: ⑭
      question: "0_netstat/q-info"
      codehelp: "0_netstat/codehelp"
      codeanswer: "0_netstat/codeanswer"
      mimetype: "bash"
      points: "1" ⑮
      output:
        log: "ascii"
```

① LabLearningObject Name

② Here you can describe your LLO. Describe what students should know or be able to do at the end of the course that they couldn't do before!

③ The "base url" field is the landing page of your LLO.

④ Code spec

⑤ Task Name

⑥ A Swarmlab service is an dockerized microservice. It receives code snippets to execute, runs these code snippets, and returns the result and output of the execution (Click on Field [Service] will show all available services to use).

⑦ Task info. Here you can describe your Task

⑧ Set the programming language!

⑨ The full path of the executable

⑩ Here you can add the code snippet to be executed. (Using the "Give It A Try" option you can test your Code.)

⑪ Exec Options

⑫ Code block syntax highlighting

⑬ Output: Console, file

⑭ Challenge spec [Optional]

⑮ Difficulty - 1 beginer, 2 intermediate, 3 advanced, 4 expert, 5 guru

This example LO is organized as follows

```
/test # The ''base url'' https://git.swarmlab.io:3000 ①
├──── 0_netstat ②
│      ├──── code ③
│      ├──── codeanswer ④
│      ├──── codehelp ⑤
│      ├──── info ⑥
│      └──── q-info ⑦
└──── intro ⑧
```

① Base Directory - based on: LabLearningObject Name. (This Directory Location is the desired URL in "base_url" above)

② Code spec - based on: display_name

③ Code snippets to execute

④ Contain the answer.

⑤ Hints.

⑥ Task info file.

⑦ Question file

⑧ Intro file.

> ℹ️ Swarmlab.io notebooks are organized in files easy to collaborate using version control systems like git.

Text

Text can be added to swarmlab.io Notebooks using Ascidoctor

> ℹ️ Asciidoctor is a fast text processor and publishing toolchain for converting AsciiDoc content to HTML5, DocBook 5, EPUB3, PDF and other formats. Asciidoctor is the leading implementation of the AsciiDoc syntax, first introduced and implemented in the Python-based AsciiDoc project.

Why AsciiDoc?

- Standardized format

    ◦ The formatting of Asciidoc is standardized so there is only one 'flavor' unlike in Markdown. The definitive user guide is here

- Asciidoctor can convert directly to HTML or DocBook.

AsciiDoc is about being able to focus on expressing your ideas, writing with ease and passing on knowledge without the distraction of complex applications or angle brackets.

AsciiDoc works because:

- It's readable
- It's concise
- It's comprehensive
- It's extensible
- It produces beautiful output (HTML, DocBook, PDF, ePub and more)

AsciiDoc is easy to write and easy to read (in raw form). It's also easy to proof and edit. After all, it's plain text, just like that familiar e-mail.

Best of all, it only requires a text editor to read or write.

AsciiDoc vs Markdown

The defacto lightweight markup language is Markdown. (At least, that's what you call it at first). **The main advantage of Markdown lies in its primitive syntax:** its manual and cheatsheet are one and the same. **But this advantage is also its greatest weakness.**

As soon as authors need something slightly more complex than basic prose (e.g., tables, cross references, footnotes, embedded YouTube videos, etc.), they find themselves **resorting to embedded HTML** or seeking out more feature-rich implementations. Markdown has become a maze of different implementations, termed "flavors", which make a universal definition evasive.

> The IETF has declared "there is no such thing as "invalid" Markdown." See This Is Markdown! Or: Markup and Its Discontents.

Here's how the story inevitably goes.

- You start out with Markdown.
- Then it's Markdown + X.
- Then Markdown + X + Y.
- And down the rabbit hole you go.

What's worse, X and Y often require you to sprinkle in HTML, unnecessarily coupling content with presentation and wrecking portability. Your instinct to choose Markdown is good. There are just better options.

Code

The Swarmlab Services offers an easy place to run your Source code.

you can create, run and share your live code right away without worrying about the system that's running

Swarmlab Learning Object Notebooks are language agnostic

As you see it's really easy to create a LO you may not even need the "Bootstrap LearningObject" menou.

Follow the instructions given in the "yaml-index" file and you'll be ready to start.

You can allways create a prototype LO from the menu and edit it later on.

5. Get Involved

Swarmlab.io is a open source project born to support interactive Virtual Labs and interactive Learning Objects across all programming languages and systems.

Swarmlab.io is open and welcoming to all.

And we need your help.

We invite anyone to come and participate in the creation of Swarmlab.io. Regardless of your skillset, we're sure there is something you can add to our community and project.

5.1. Reporting Security Issues

The Swarmlab.io maintainers take security seriously. If you discover a security issue, please bring it to their attention right away!

Please DO NOT file a public issue, instead send your report privately to security@swarmlab.io.

Security reports are greatly appreciated and we will publicly thank you for it, although we keep your name confidential if you request it.

5.2. Reporting other issues

A great way to contribute to the project is to send a detailed report when you encounter an issue. We always appreciate a well-written, thorough bug report, and will thank you for it!

Check that our issue database doesn't already include that problem or suggestion before submitting an issue. However, if you have ways to reproduce the issue or have additional information that may help resolving the issue, please leave a comment.

When reporting issues, always include: as much information as you can

Create a new issue

Also include the steps required to reproduce the problem if possible and applicable. This information will help us review and fix your issue faster. Don't forget to remove sensitive data

from your logfiles before posting (you can replace those parts with "REDACTED"). Quick contribution tips and guidelines contributions can make a docker-compose prototype

5.2.1. Creating an issue

- On git.swarmlab.io , navigate to the main page of the feedback repository.
- Under repository name, click **Issues.**
- Click **New issue.**
- Type a title and description for your issue.
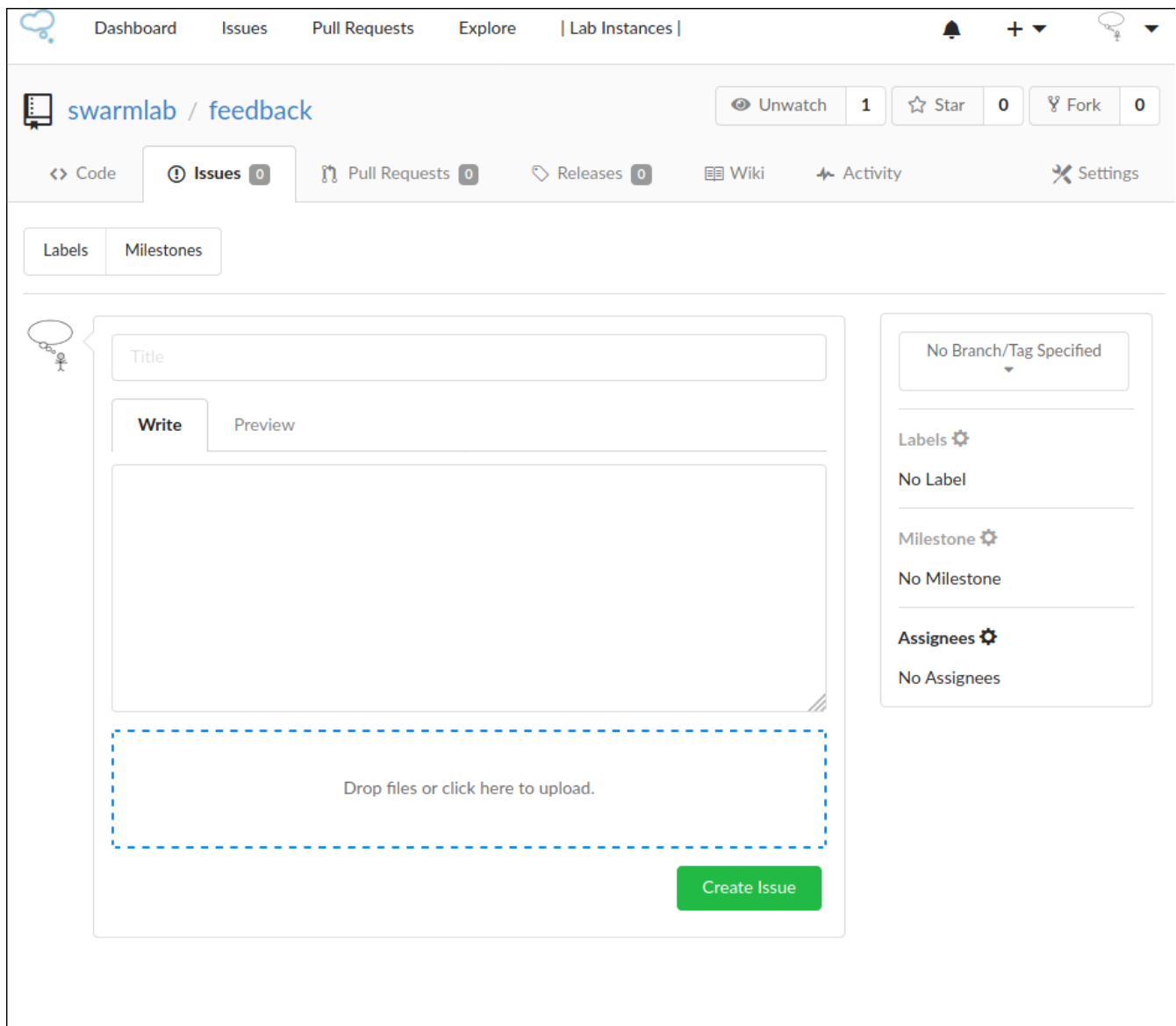
- When you're finished, click Submit **Create issue.**

5.3. Develop swarmlab services

All developers already familiar with docker-compose and that have already set up interesting stacks that may be useful to others, are highly encouraged to add them to the repository and share them. Improvements on the current samples are also very much appreciated!

5.4. Create Learning Object Notebooks

See here

[1] Learning Technology Standards Committee (2002), Draft Standard for Learning Object Metadata. IEEE Standard 1484.12.1 (PDF), New York: Institute of Electrical and Electronics Engineers, retrieved 2008-04-29

[2] https://en.wikipedia.org/wiki/Learning_object#Definitions

See here