Setup your VM on surf research cloud

R. Grouls

Version 0.4

February 14, 2025

Contents

Chapter 1

Create a VM

1.1 Introduction

For this semester, we will be using the SURF Research Cloud to run virtual machines. This has a lot of advantages:

1. Uniformity of hardware and software, so we can help you better.
2. Lesson material is tested on the VMs, so you can be sure that everything will work.
3. Instructions are optimized for the VMs, so you can follow them step by step.
4. We will have GPUs available on the VMs, so you can run deep learning models with considerably less time.
5. It is pretty common to run code on a VM in production; getting familiar with setting up VMs is a useful skill to have.

This manual will guide you through the process of setting up your VM on the SURF Research Cloud. The first requirement is that you should receive an invite to the SURF Research Cloud. If you have not received an invite, please contact us. We will need your @student.han.nl email address to send you an invite.

1.2 Creating a VM

Click on the invite link in your email. Log in with your @han email account. Note: make sure to drop the student part of your email address. So if your email is name@student.han.nl, you should use name@han.nl to log in.

1.2.1 Requesting a wallet

The first screen you encounter is an overview where you can enter the SURF Research Cloud, see Figure 1.1. Click on "Open".

After this, you should see a screen similar to Figure 1.2. Here you can request a wallet.

The wallet

The wallet is a way to pay for the VMs. You will get a personal budget, and it depends on which VM you choose how many credits you will use. To give you an idea: it costs around 50 credits per 24 hours for a 2 core CPU with 16GB of RAM, and a GPU will cost you about 400 credits per 24 hours. Put as a subject "SRC account for MADS-HAN" and the SURF helpdesk will approve your request, typically during workhours on the same day.
So the first thing you need to think about, is how many RAM and cores do you need? Some rough guidelines:

1. For the processing of small datasets (most datasets we use are considered small) 8GB is enough. Note that the pandas library has a really unhealthy relationship to RAM usage; often it is usefull to switch to the more modern polars library for preprocessing if you run into RAM problems because of pandas.

2. For most data visualisation, 8GB RAM is enough

3. For most simple programs in deployment, 8GB (or even less) is often more than enough to run the program.

4. When doing machine learning, some models require more RAM. Typically, image models (CNN architectures) require more RAM, typically about 16GB. The same for Transformer architectures for the bigger models, and most LLMs will need 16GB or even more.

As a rule of thumb: make it easy to setup your VM, so setting up a new VM is just a few minutes of work. Personally, I like to use scripts to install the things I need on a VM. Take a look at my repo for an example: `https://github.com/raoulg/serverinstall`. You can use the scripts there, or build your own script to automate setting up things you like. If setting up is easy, you can just start low (eg with 8GB and 2 cores) and scale up if it doesnt work out for what you want to do.
It is also very common to, for example, write code locally, and only spin up a GPU VM at the moment you need it and simple ‘git clone‘ your work to the GPU machine.

1.2.2 Workspace access setup

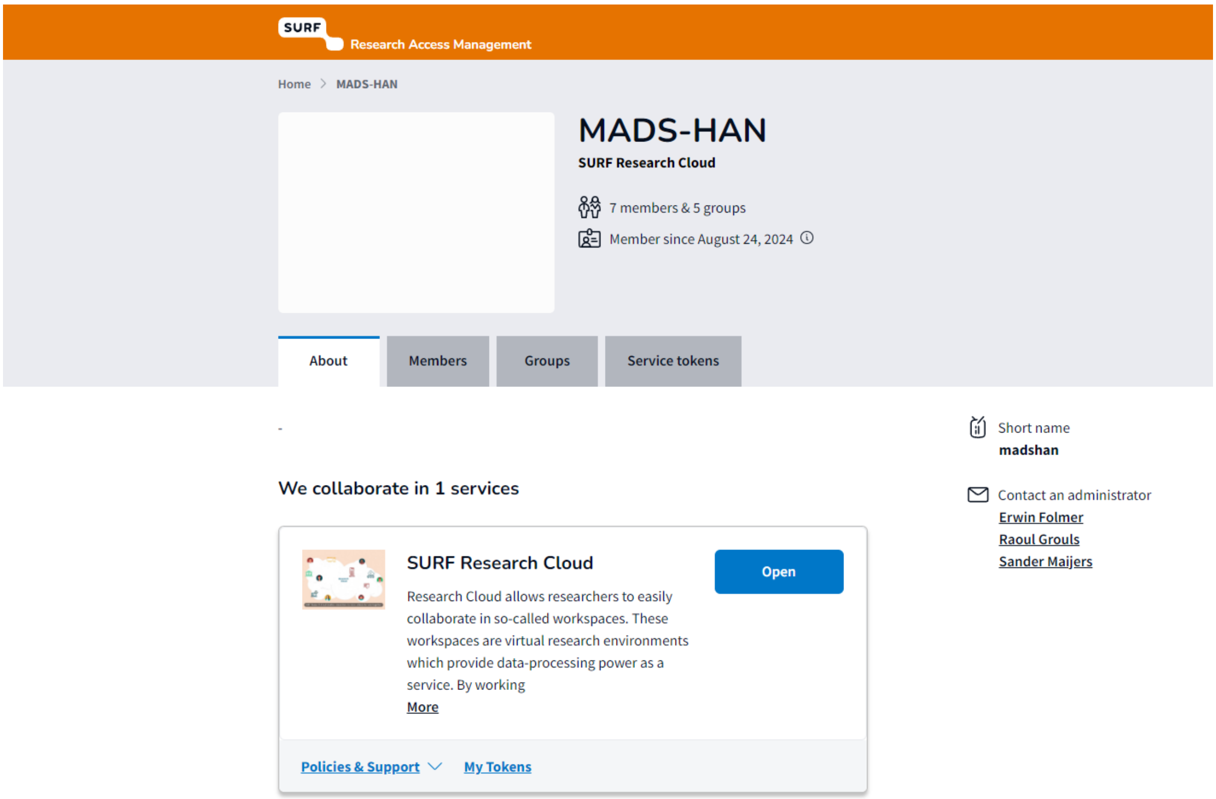
After you get your wallet approved, you should see something as shown in Figure 1.3. Choose “workspace access setup”. This will bring you to a screen where you need to set up your SSH key.

1.2.3 SSH key

You will need a SSH key. If you don’t already have one and need a refresher on how to create one, see the following link: `https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent`.

If you don’t really have an idea what an SSH key is, please watch this 10 minute video `https://youtu.be/dPAw4opzN9g?si=3k8Rvvd5uYgeQEfd`. Also, if you decide to skip it, but you get all sorts of issues with the ssh-keys, it is probably because you dont understand what you are doing. So the 10 minutes is a good investment of your time, you will earn it back!

You should add your public key to the workspace setup.



MADS-HAN

SURF Research Cloud

7 members & 5 groups

Member since August 24, 2024

-

We collaborate in 1 services

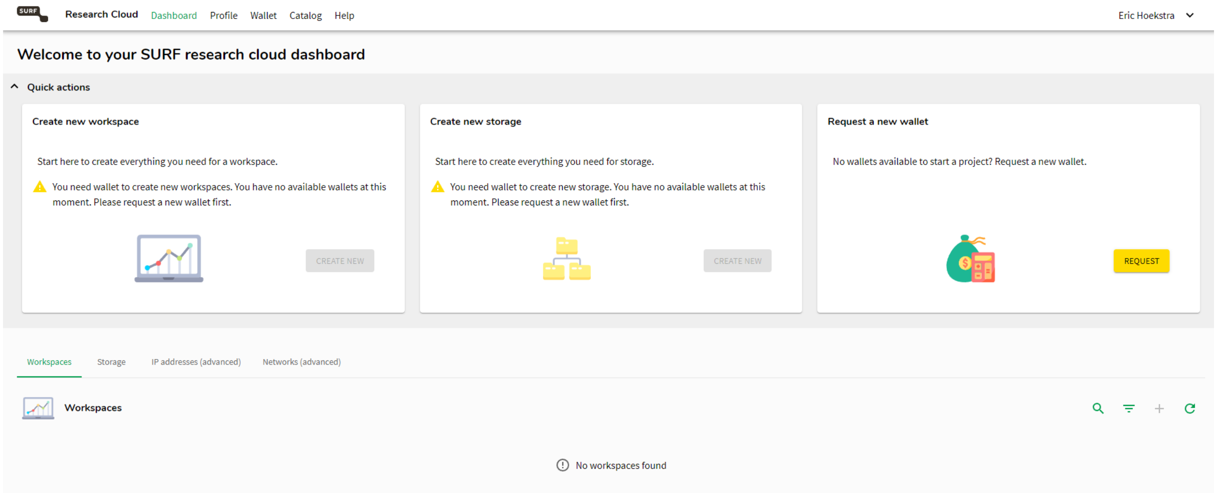
SURF Research Cloud

Research Cloud allows researchers to easily collaborate in so-called workspaces. These workspaces are virtual research environments which provide data-processing power as a service. By working

Short name
madshan

Contact an administrator
Erwin Folmer
Raoul Grouls
Sander Maijers

Figure 1.1: Open the SURF Research Cloud



Welcome to your SURF research cloud dashboard

Create new workspace

Start here to create everything you need for a workspace.

⚠ You need wallet to create new workspaces. You have no available wallets at this moment. Please request a new wallet first.

Create new storage

Start here to create everything you need for storage.

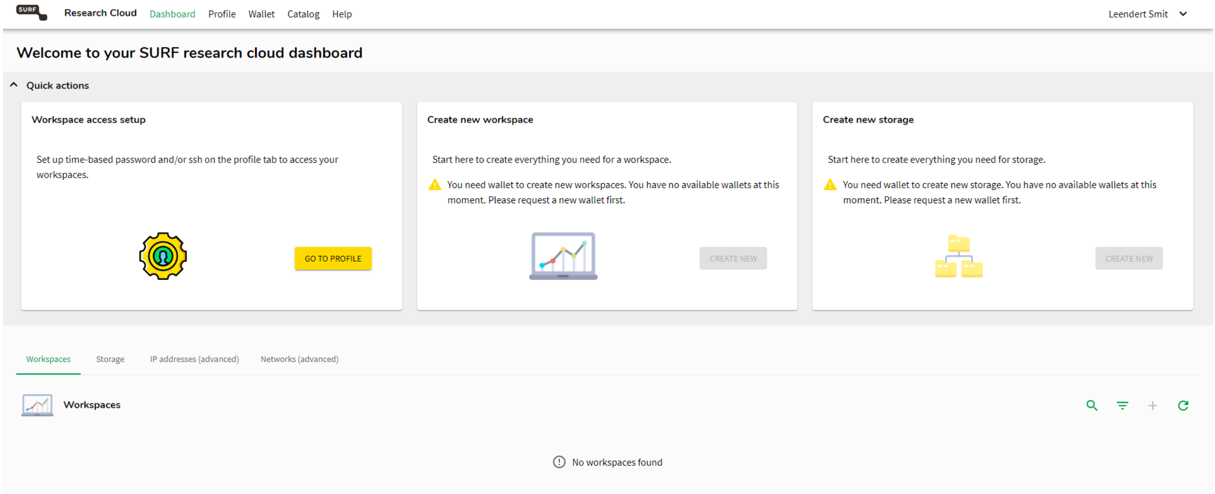
⚠ You need wallet to create new storage. You have no available wallets at this moment. Please request a new wallet first.

Request a new wallet

No wallets available to start a project? Request a new wallet.

No workspaces found

Figure 1.2: Request a wallet



Welcome to your SURF research cloud dashboard

Workspace access setup

Set up time-based password and/or ssh on the profile tab to access your workspaces.

Create new workspace

Start here to create everything you need for a workspace.

You need wallet to create new workspaces. You have no available wallets at this moment. Please request a new wallet first.

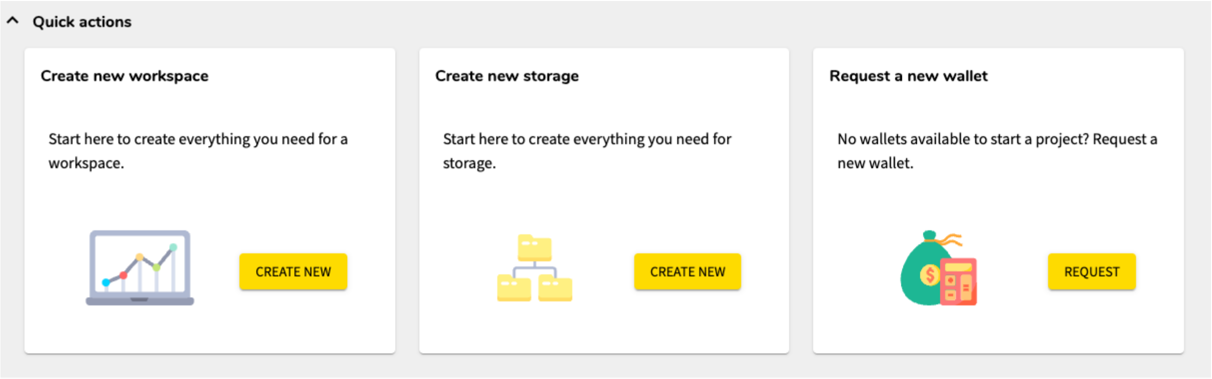
Create new storage

Start here to create everything you need for storage.

You need wallet to create new storage. You have no available wallets at this moment. Please request a new wallet first.

No workspaces found

Figure 1.3: Choose 'workspace access setup'.



Create new workspace

Start here to create everything you need for a workspace.

Create new storage

Start here to create everything you need for storage.

Request a new wallet

No wallets available to start a project? Request a new wallet.

Figure 1.4: Choose 'Create new' under workspace.

The SSH keys

1. Please, pay attention **where** you store your keys on your system. If necessary, write down the location. On linux and mac it will typically be in ~/.ssh, on windows it can be anywhere depending on your setup. Your public key is a string of characters inside a text file; if you open the .pub file with a text editor, you should see something like `ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDQ....` This is what you need to copy-paste into the SURF Research Cloud.

2. Never share your private key, only the public key (ending in .pub).

1.2.4 New workspace

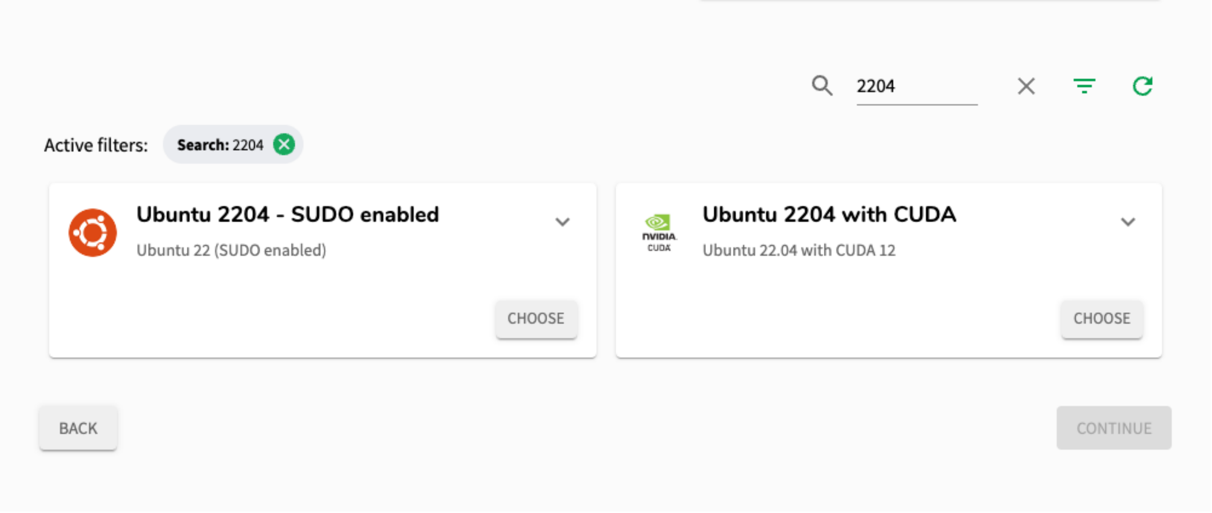
After setting up SSH, you should be able to create a workspace like in Figure 1.4.

You will run to some choices. Note that probably step 1 and 2 will be automatically selected and thus skipped. Choose the following:

1. Choose a collaborative organisation: MADS-HAN

2. Choose a wallet: SRC account for MADS-HAN

3. Catalog item: search for 2204 and choose Ubuntu 2204 - SUDO enabled, see Figure 1.5.



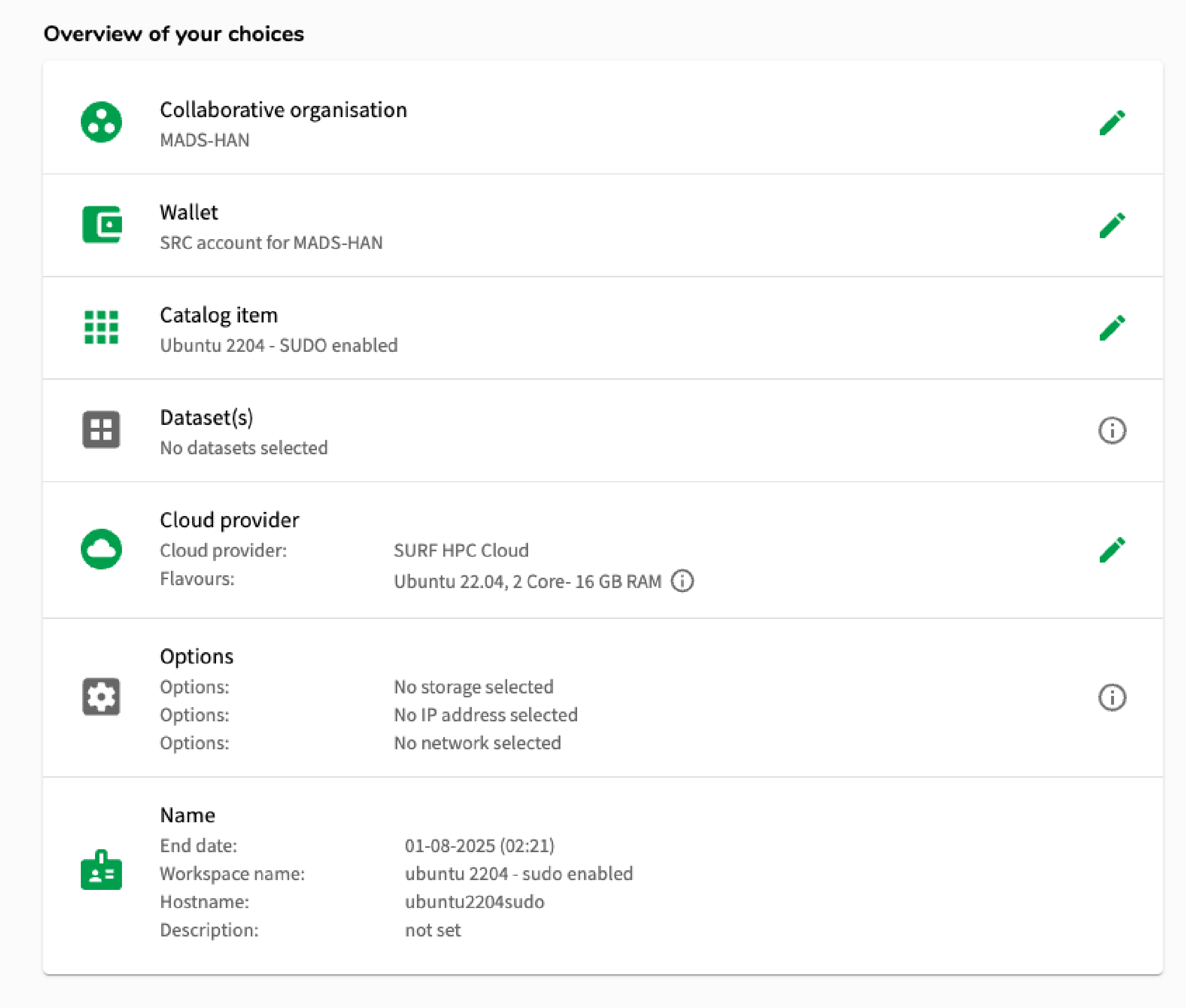
2204

Active filters: Search: 2204

Ubuntu 2204 - SUDO enabled
Ubuntu 22 (SUDO enabled)

Ubuntu 2204 with CUDA
Ubuntu 22.04 with CUDA 12

Figure 1.5: Choose Ubuntu 2204



Overview of your choices

Collaborative organisation
MADS-HAN

Wallet
SRC account for MADS-HAN

Catalog item
Ubuntu 2204 - SUDO enabled

Dataset(s)
No datasets selected

Cloud provider
Cloud provider: SURF HPC Cloud
Flavours: Ubuntu 22.04, 2 Core- 16 GB RAM

Options
Options: No storage selected
Options: No IP address selected
Options: No network selected

Name
End date: 01-08-2025 (02:21)
Workspace name: ubuntu 2204 - sudo enabled
Hostname: ubuntu2204sudo
Description: not set

Figure 1.6: Your overview should look like this

4. Choose cloud provider: SURF HPC Cloud

5. Choose size: eg for machine learning, 2 Core - 16GB RAM, or for data exploration, 8GB.

6. Choose the expiration date of the machine; 3 months is a reasonable time for the course.

7. Workspace name: use your LastnameInitial-semestername. For example, mine would be groulsr-UOS2 or groulsr-UOS3.

8. Workspace Hostname: use your LastnameInitial. For example, mine would be groulsr.

After you went through these steps, you should get an overview like in Figure 1.6. We will be experimenting with GPUs in the machine learning course; but because you have a limited budget of compute, start with the CPU first. For GPU, you should

a choose ‘Ubuntu 2204 with CUDA‘ in step 3 (see Figure 1.5) and

b for size choose ‘RTX2080 - 1 GPU‘ in step 5 (there are not enough A10s for everyone available, the RTX should be available for everyone).

c To discern them, add ‘GPU‘ to the workspace name and hostname in step 7. For example, mine would be ‘groulsr-UOS3-GPU‘ and ‘groulsrgpu‘.

After clicking "submit" you should see your VM being created. This can take a while.

Naming conventions

1. UOS2 (second semester class) and UOS3 (third semester class) will be creating VMs in the same workspace, so it is **really important** to stick to the naming conventions! We will **delete** VMs that do not adhere to the naming conventions!

2. use your LastnameInitial-semester for the workspace name and LastnameInitial for the hostname. Add -GPU for a GPU VM, eg. groulsr-UOS1 or groulsr-UOS3-GPU etc.

Chapter 2

Connect to your VM

2.1 Pausing and Resuming your VM

You will get some credits, probably for around 100 hours of compute to start with. Running a small VM (2 cores, 16GB) will cost you about 50 credits per day, so keep track of your usage. Please note that even if you don't actually use the VM, it will **still cost you credits**: it is still "on", regardless if you are logged in. So, when you are not using the VM, **please pause** it. You can do this in two ways:

1. Log in to the SURF Research Cloud, navigate to your Dashboard, and click "Resume" or "Pause" on your workspace. Note that it might very well be possible we will have more than 60 VMs here, which is the reason why we are strict on the naming conventions, see 1.2.4 for more details.

2. Use surfcontroller, a package we developped in Python. You can install it by running `pip install surf-controller`. You can then run `surfcontroller` from your terminal and select your VMs to pause/resume, see Figure 2.1.

Autoshutdown @ 21:00

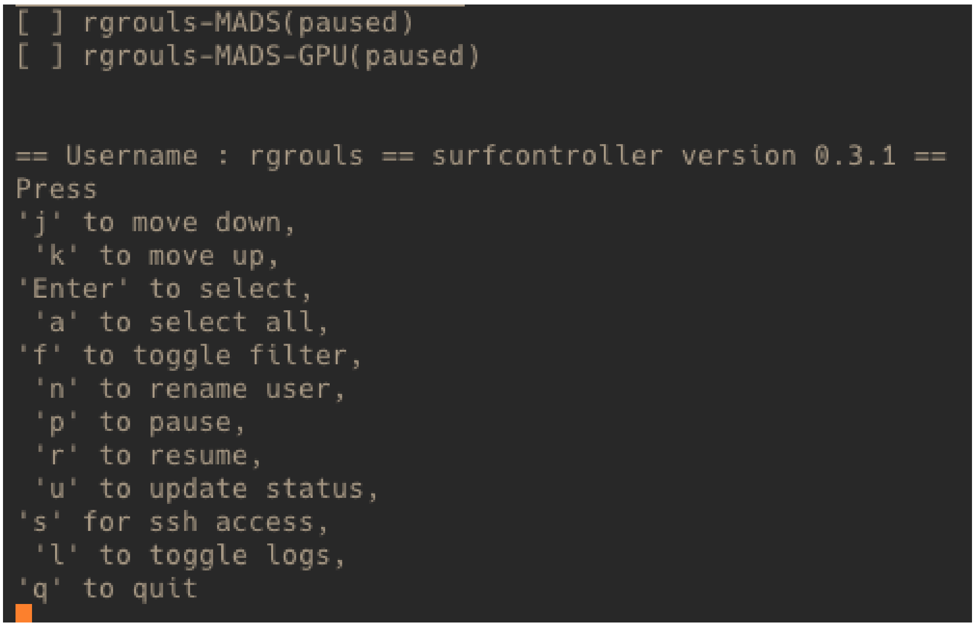
1. We will **automatically pause all VMs every day at 21:00**. Experience learns that there will always people who forget to shutdown their VM, and if you forget your GPU for a week you will spend most of your credits.

2. If you want to do long computations (eg for hypertuning your model) you can request a lift from the autoshutdown with your teacher.

2.2 Finding your IP address

After the VM is created, you can use the dropdown menu to find the URL with the IP address of your VM. You can use this IP address to connect to your VM, see Figure 2.4. Make sure your VM is running before you try to connect. You can open VScode and when you have installed the "Remote Development" extension from Microsoft, you will have a small blue icon in the bottom left corner, see Figure 2.2. Click on it and choose "Remote-SSH: Connect to Host...", see Figure 2.3. You can now enter `user@IP-address` of your VM and connect to it. You will find this under 'URL' in the dropdown menu. For example, I would enter `rgrouls@145.38.192.86` to connect.

The first time you will get a question saying this is an unknown host and if you want to add its fingerprint. You can accept this and continue. After this, you should have entered your VM. You can now start working on your VM.



```
[ ] rgrouls-MADS(paused)
[ ] rgrouls-MADS-GPU(paused)


== Username : rgrouls == surfcontroller version 0.3.1 ==
Press
'j' to move down,
 'k' to move up,
'Enter' to select,
 'a' to select all,
'f' to toggle filter,
 'n' to rename user,
 'p' to pause,
 'r' to resume,
 'u' to update status,
's' for ssh access,
 'l' to toggle logs,
'q' to quit
```

Figure 2.1: Surfcontroller screenshot

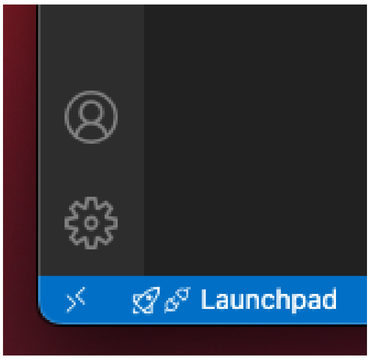


Figure 2.2: Click the small blue icon in the bottom left corner

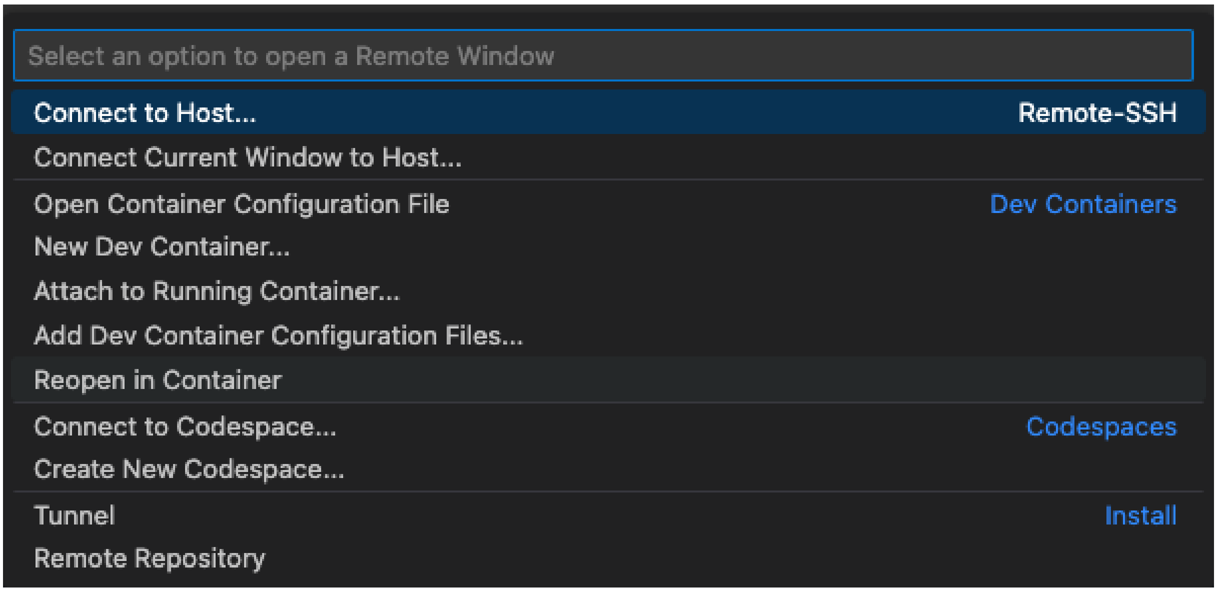
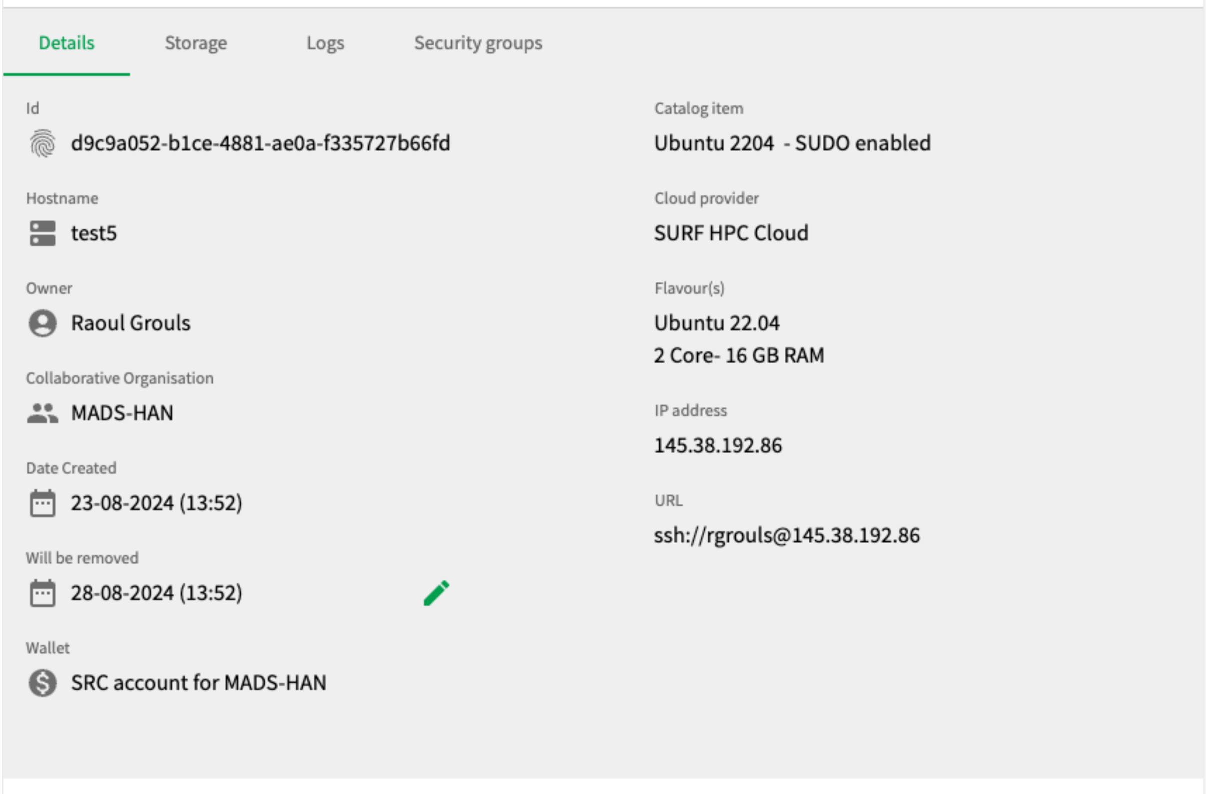

Select an option to open a Remote Window
Connect to Host...
Remote-SSH
Connect Current Window to Host...
Open Container Configuration File
Dev Containers
New Dev Container...
Attach to Running Container...
Add Dev Container Configuration Files...
Reopen in Container
Connect to Codespace...
Codespaces
Create New Codespace...
Tunnel
Install
Remote Repository

Figure 2.3: connect to host



Id
d9c9a052-b1ce-4881-ae0a-f335727b66fd

Hostname
test5

Owner
Raoul Grouls

Collaborative Organisation
MADS-HAN

Date Created
23-08-2024 (13:52)

Will be removed
28-08-2024 (13:52)

Wallet
SRC account for MADS-HAN

Catalog item
Ubuntu 2204 - SUDO enabled

Cloud provider
SURF HPC Cloud

Flavour(s)
Ubuntu 22.04
2 Core- 16 GB RAM

IP address
145.38.192.86

URL
ssh://rgrouls@145.38.192.86

Figure 2.4: Find you IP address and url

2.3 checking authorized keys

If you still need to enter a password when connecting, you might need to check your ssh key. The ssh-key will serve as a way to prove you are you, and you can connect to the VM without a password. To do this, you need to add your public key to the authorized keys on the VM.

On the VM:

1. navigate with cd to '~/.ssh'
2. check if there is a file called 'authorized_keys'
3. if not, create one with 'touch authorized_keys'
4. open the file with 'nano authorized_keys'
5. copy-paste your public key in this file [1] and save the file.

[1] see `https://docs.github.com/en/authentication/connecting-to-github-with-ssh/generating-a-new-ssh-key-and-adding-it-to-the-ssh-agent` for details

Chapter 3

Setup your VM

3.1 Clone and install the course

In the `README` of the course `https://github.com/raoulg/MADS-MachineLearning-course`, you can find the instruction to 1. install python and rye 2. clone the repo 3. setup git and install the requirements.

3.2 Optional: eyecandy

I typically like to install some packages for eyecandy.

3.2.1 starship

1. run `curl -fsSL https://starship.rs/install.sh | bash` in your terminal.
2. open your `~/.bashrc` file with `nano ~/.bashrc`
3. add `eval "$(starship init bash)"` to the file

For more info, or screenshots, got to `https://starship.rs/`

Chapter 4

TLDR

TLDR

please read all the boxes similar to these and you're good.