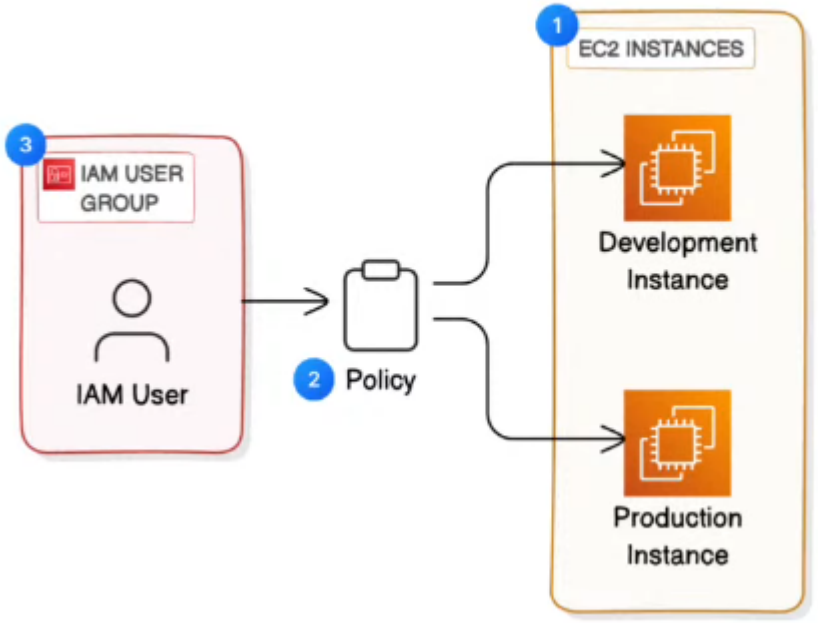
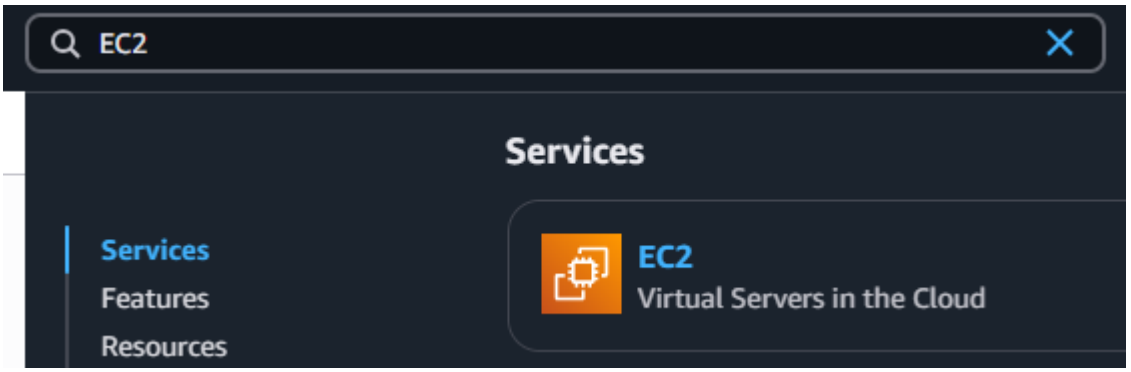
We'll launch an EC2 instance, then control who has access to it by creating some IAM policies and user groups. It will look something like this :



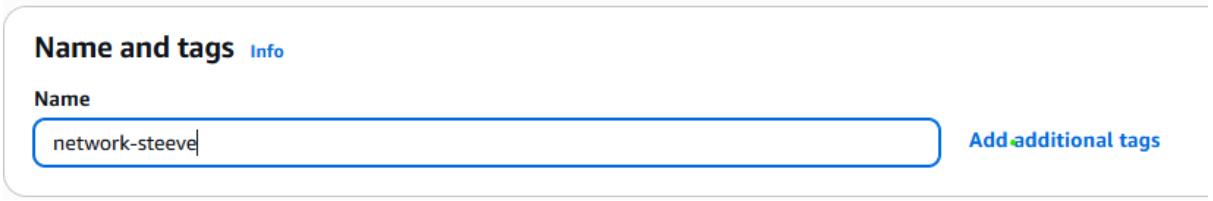
Step #1: LAUNCH INTANCE EC2

- Log in to your AWS Management Console.
- Open your EC2 console - search for it at the search bar.



Let's set up your EC2 instance!

In Name, enter the value.

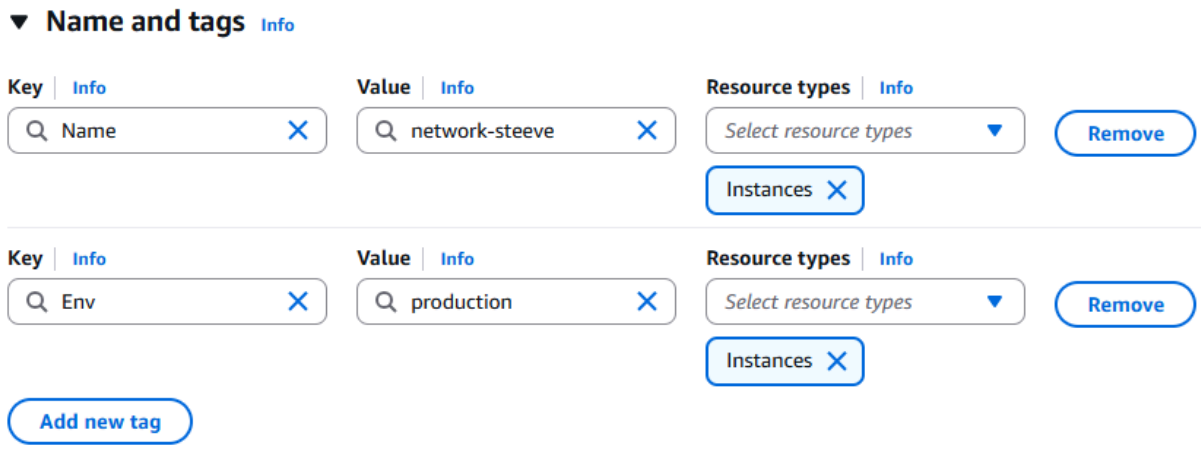


Click on “**Add additional tags”**

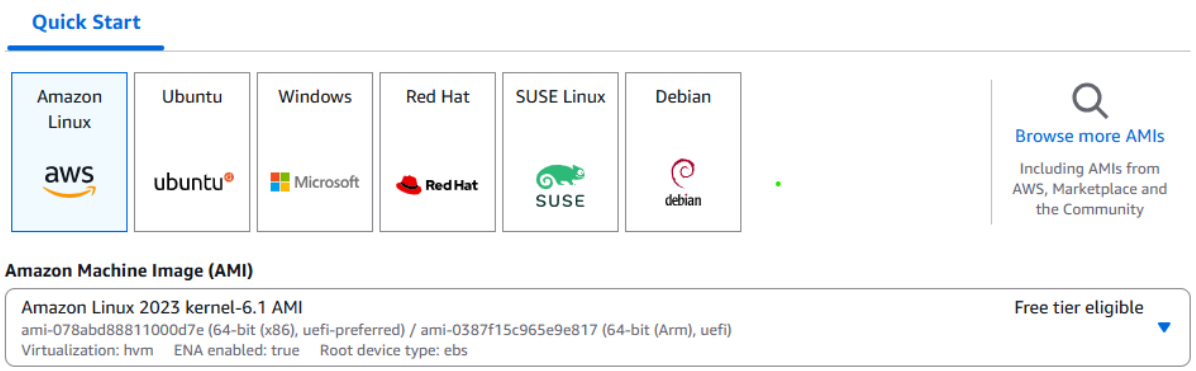
For the next tag, use this information:

- Key: Env
- Value: production

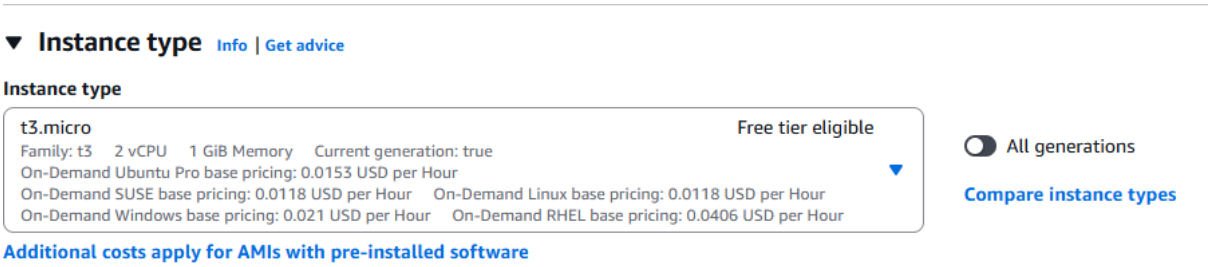
The tags are like labels you can attach to AWS resources for organization that helps you to organise your ressources or make search easily..



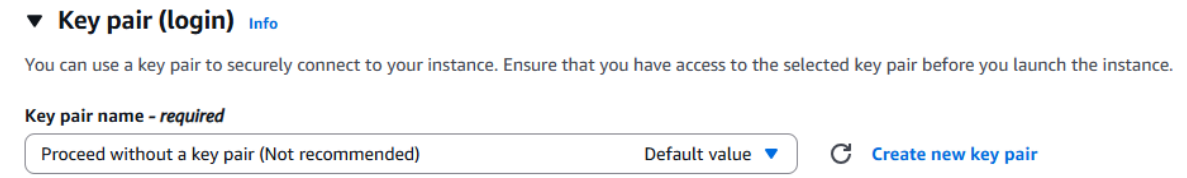
Head down to see your EC2 instance settings, and make sure the Amaon Machine Image using is “**Free tier eligible**”



For the **instance type** make sur you are using **free tier eligible**

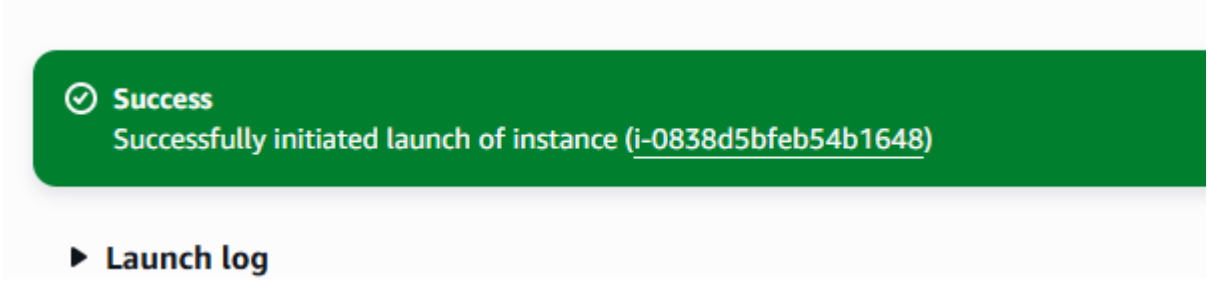


- For **Key pair, choose processing without a key pair.** key pair is used to increase security access to your instance by allowing access just by ssh connection.



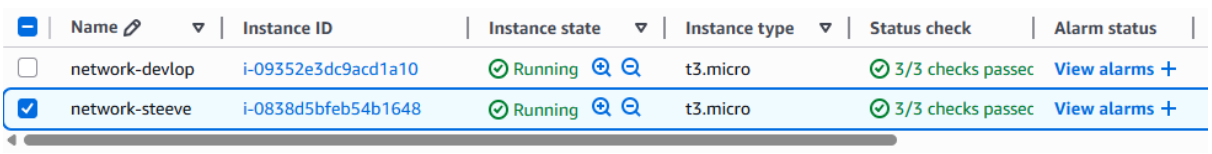
you're ready! click on launch instance

you get that message:



- Now you're going to create one more EC2 instance for development service

Create an IAM Policy

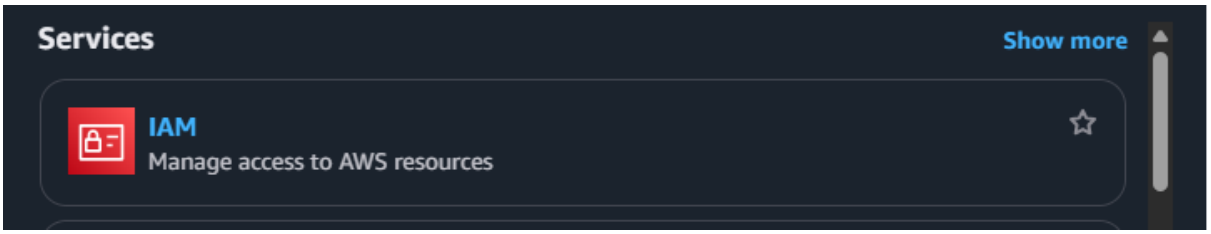


Woooo! you've deployed an EC2 instance for your production environment and development.

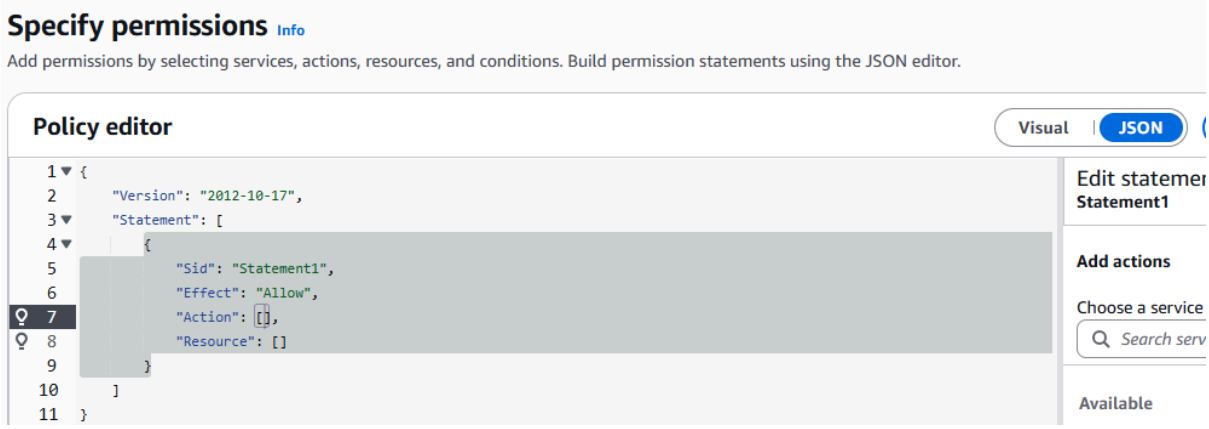
Step #2: CREATE IAM POLICY

Let’s move into our second task as NextWork's engineer, it's time to onboard the team's new intern and set up permission policies.
Our intem should have permission to the development EC2 instance but not the production instance. We don’t want them to accidentally shut down the platform or push their changes to the production environment while they’re just testing things!

Go in the search console, type IAM



- Now on the left-hand navigation panel of your IAM console, choose **Policies.**
- **Create policy**
- Switch on **Json**

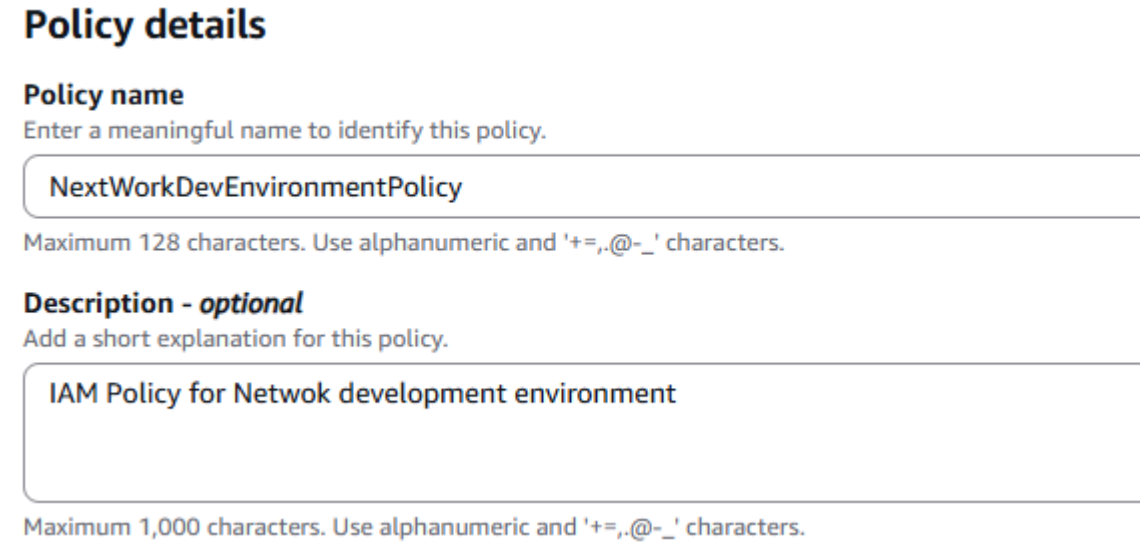


Replace de the content by this :

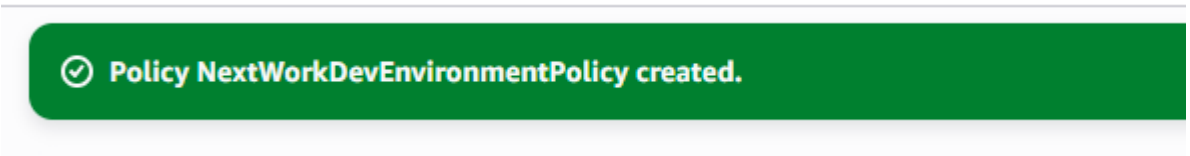
Select next

Fill in your policy's details:

- Name: NextWorkDevEnvironmentPolicy
- Description: IAM Policy for NextWork's development environment



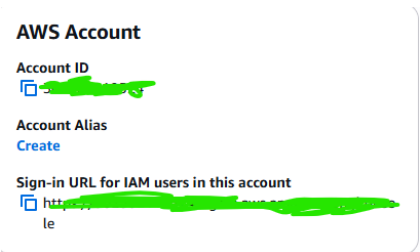
ended click on **Create Policy**



Step #3: CREATE AN AWS ACCOUNT ALIAS

In this step, get ready to simplify user login to your AWS Account using an Account Alias.

In IAM click on dashboard in left-hand side, then choose **create alias** under Account Alias



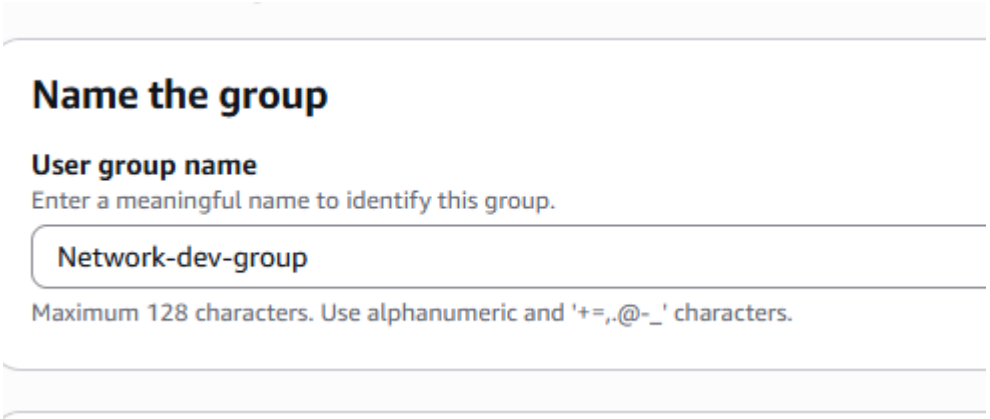
Step #4 CREATE IAM USERS AND USERS GROUP

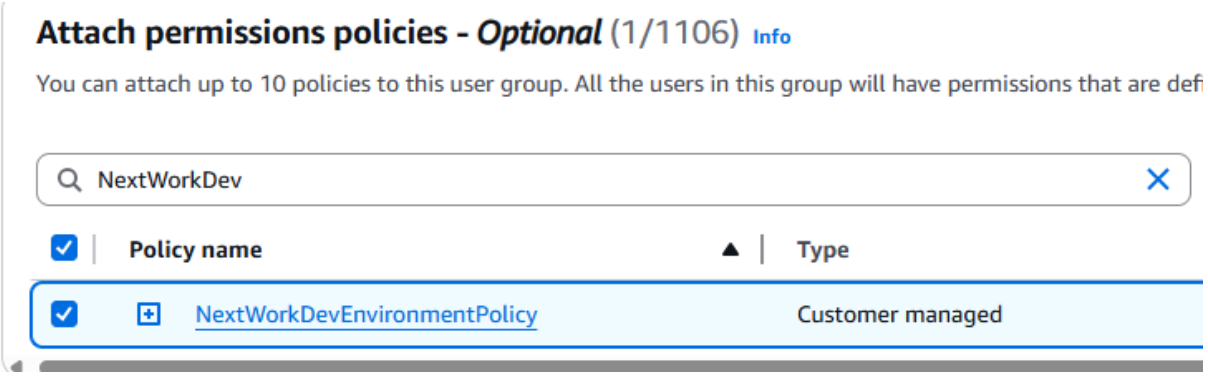
In this step, get ready to:

- Step up a dedicated IAM group for all NextWork interns, so you can manage all intern's permission from one place.
- Step up a dedicated IAM user for your new intern, so they have a way to log in.

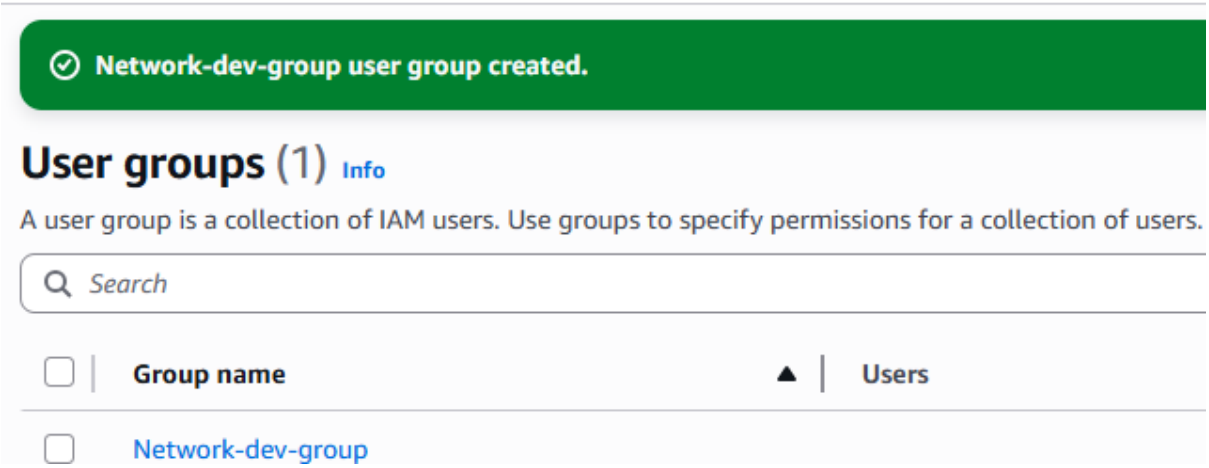
So Here we go!

- choose **users group** in your left-hand navigation panel
- **Create Group**



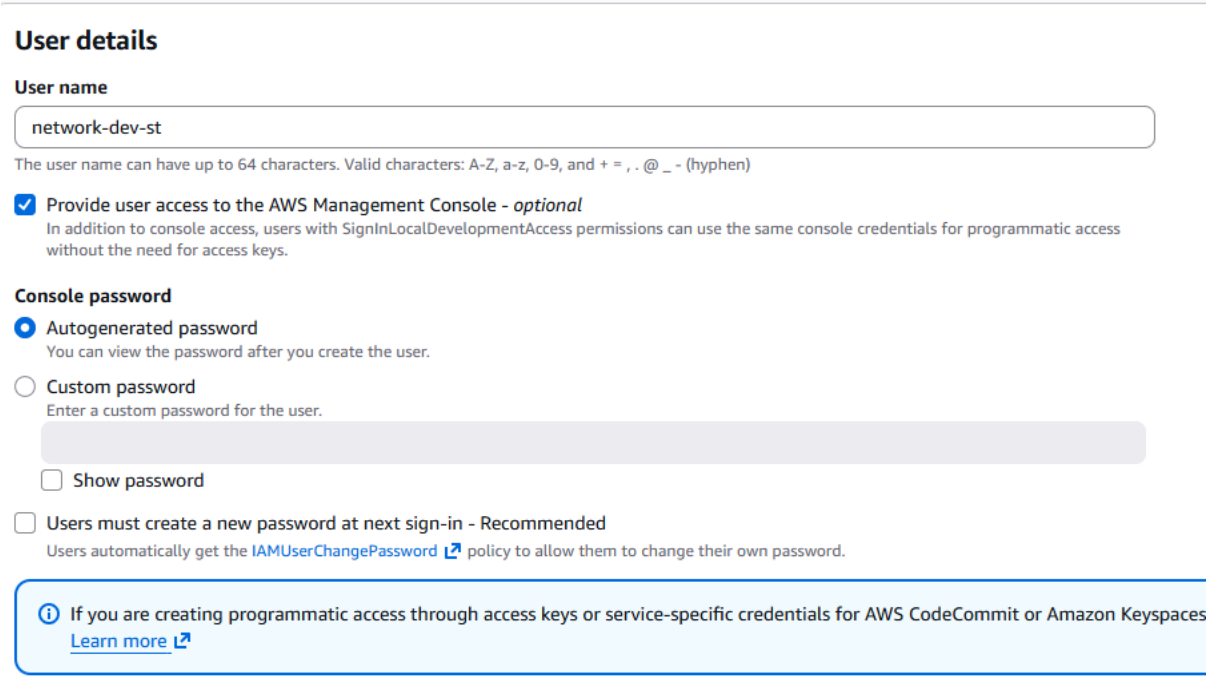


You can show confirmation message

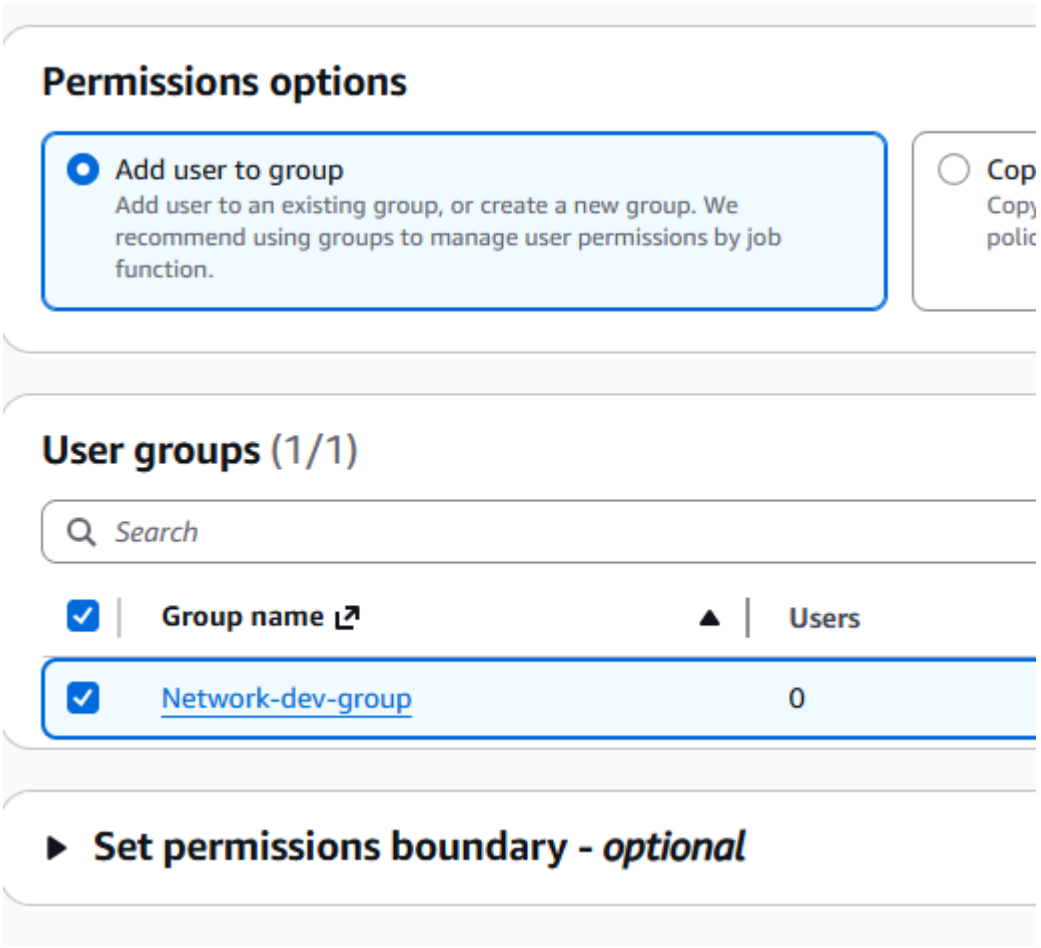


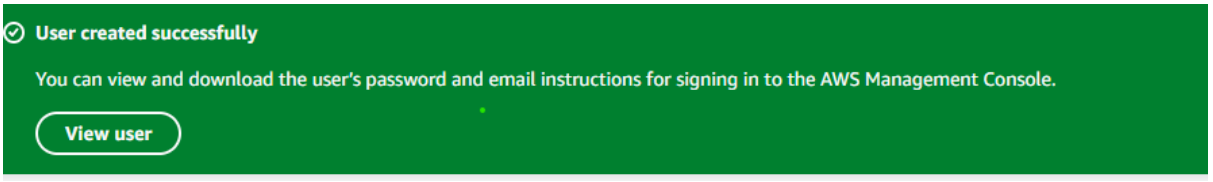
After the user group has been created you need to create a user, then add them inside the user group.

- In left-hand of panel navigation IAM choose **user,** then **Create User**



Click on **Next, T**o set permissions for your user, we'll simply add it to the user group you've created. Select the checkbox **next to nextwork-dev-group.** - Select Next. - Select **Create user!**

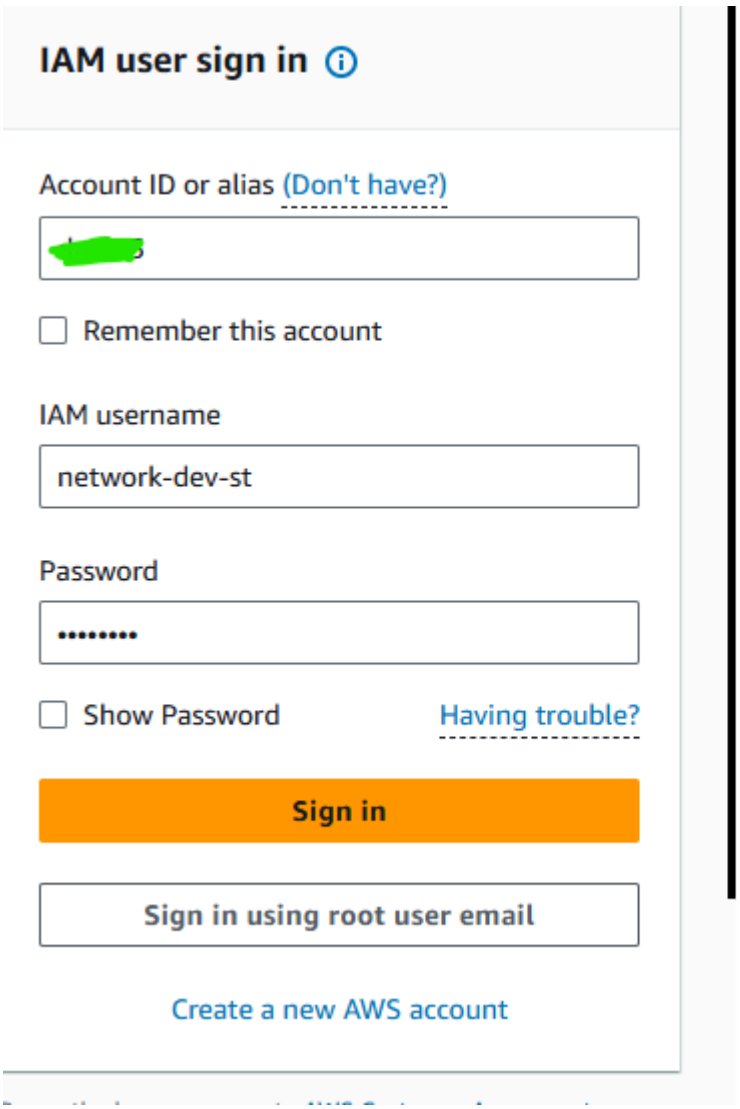




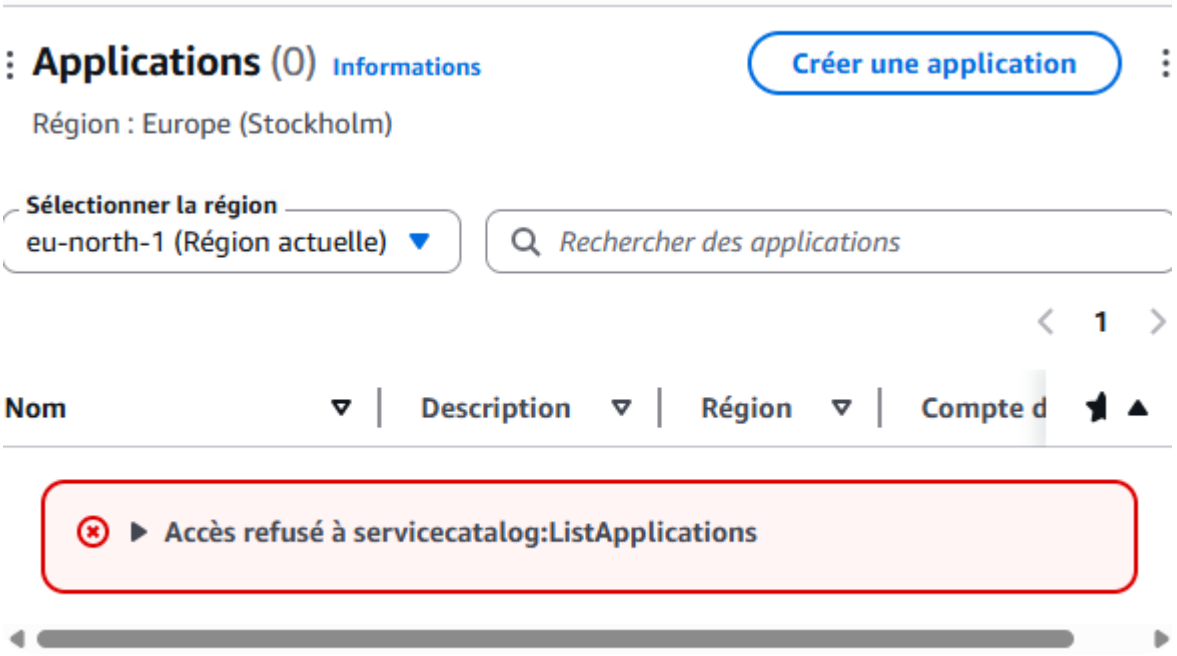
TEST YOUR INTERN'S ACCESS

In this stel we will loging to our own AWS account as the intern and test: access to production and development, because we want to make shore our intern's doesn't have aavaibility to do anything that affect our production environment.

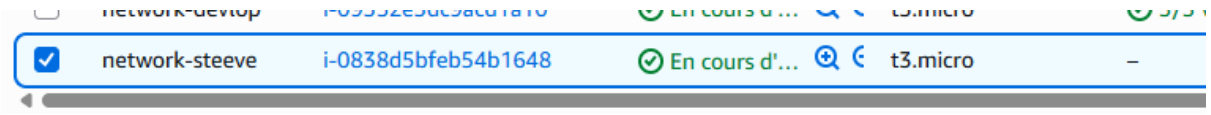
- So copy your **console sign-in** url and paste in the new window of your navigator



After login you can see that the user gets just permission to something in the development instances and anything else.



Example : When we try to stop instance named **network-steeve**, we get message error because we don't get permission to perform that action



That's end of our project.