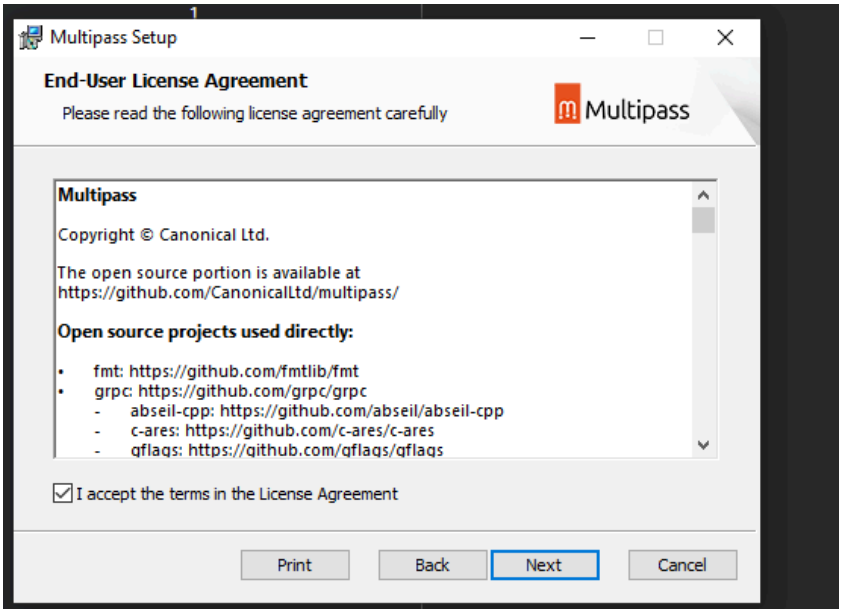
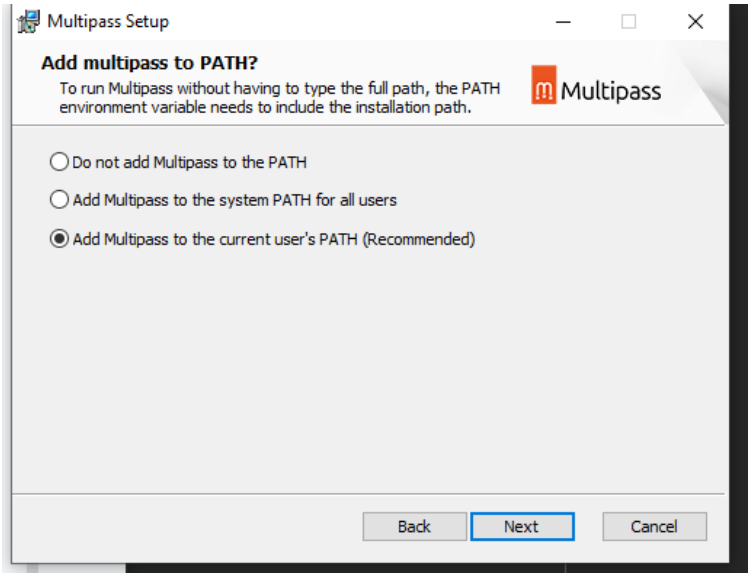
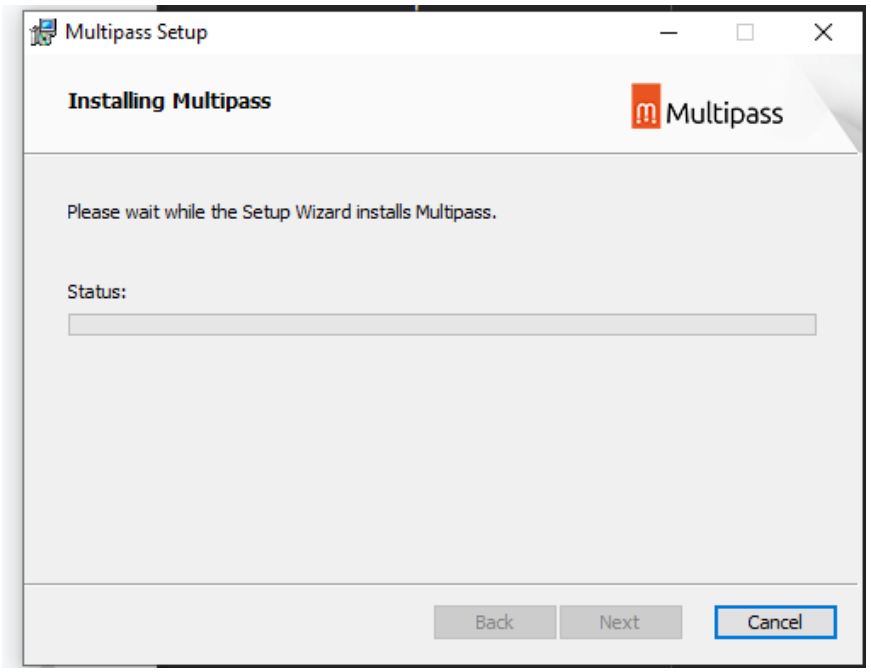
Master DevSecOps en Lite Thinking
Reto 2. Clúster de Kubernetes Local
Consultor: Jorge Valente Fragoso Mora
Alumno: Jorge Alberto Rodriguez Mendez
Correo: jarome.developer@gmail.com

1. Instalación de MicroK8s en Windows (30 puntos)
    - Instale Multipass
        1. Se va al a pagina oficial y se descarga el aplicativo

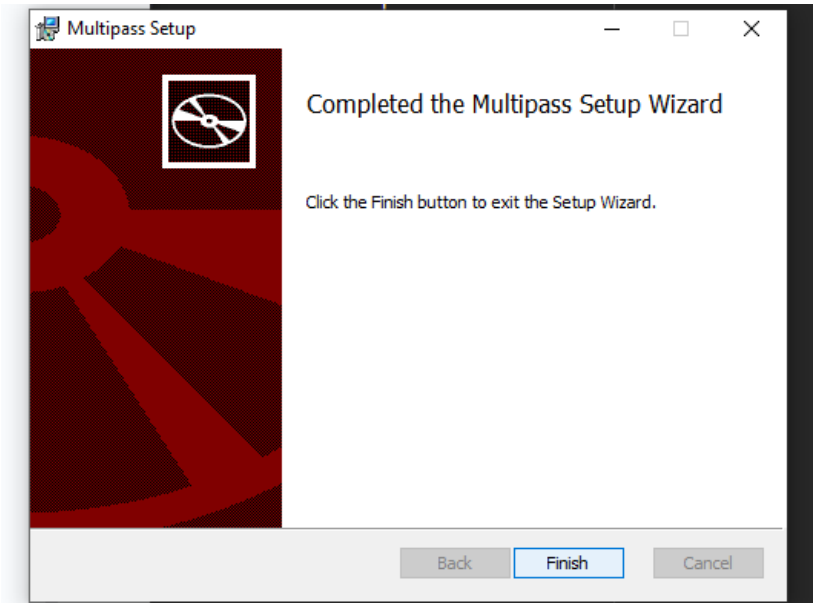


Se le da aceptar a los terminos y condiciones de uso

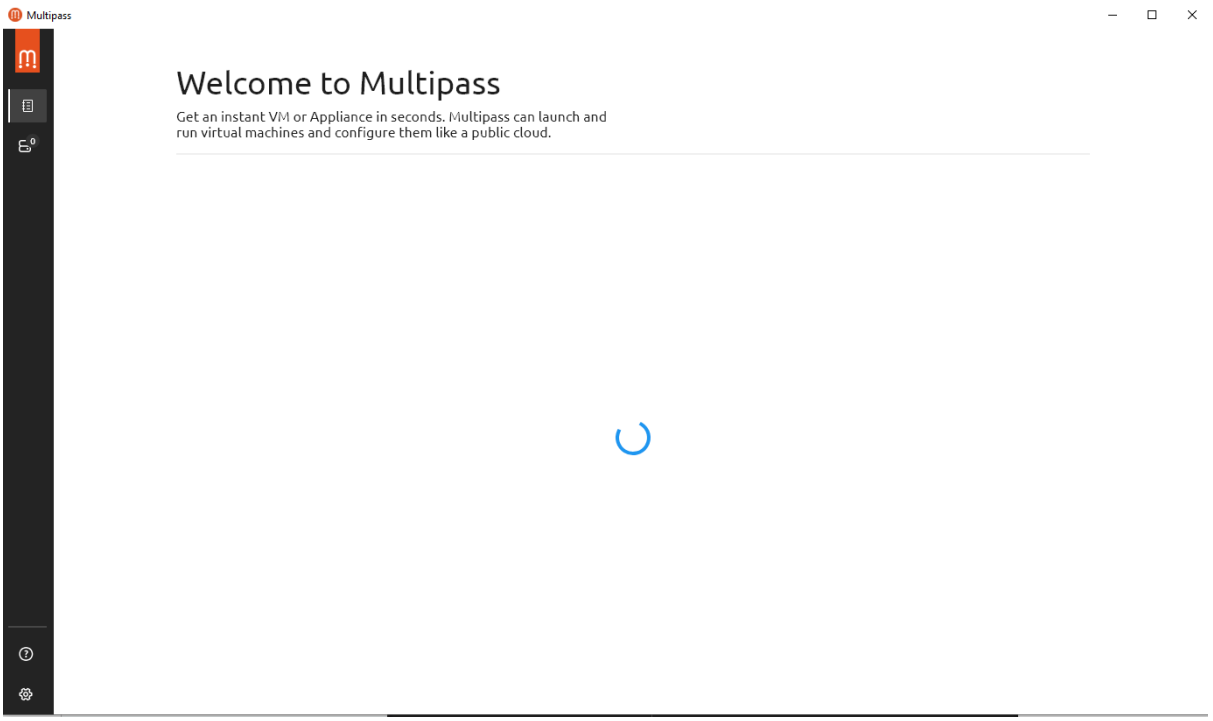




Se le da click en siguiente

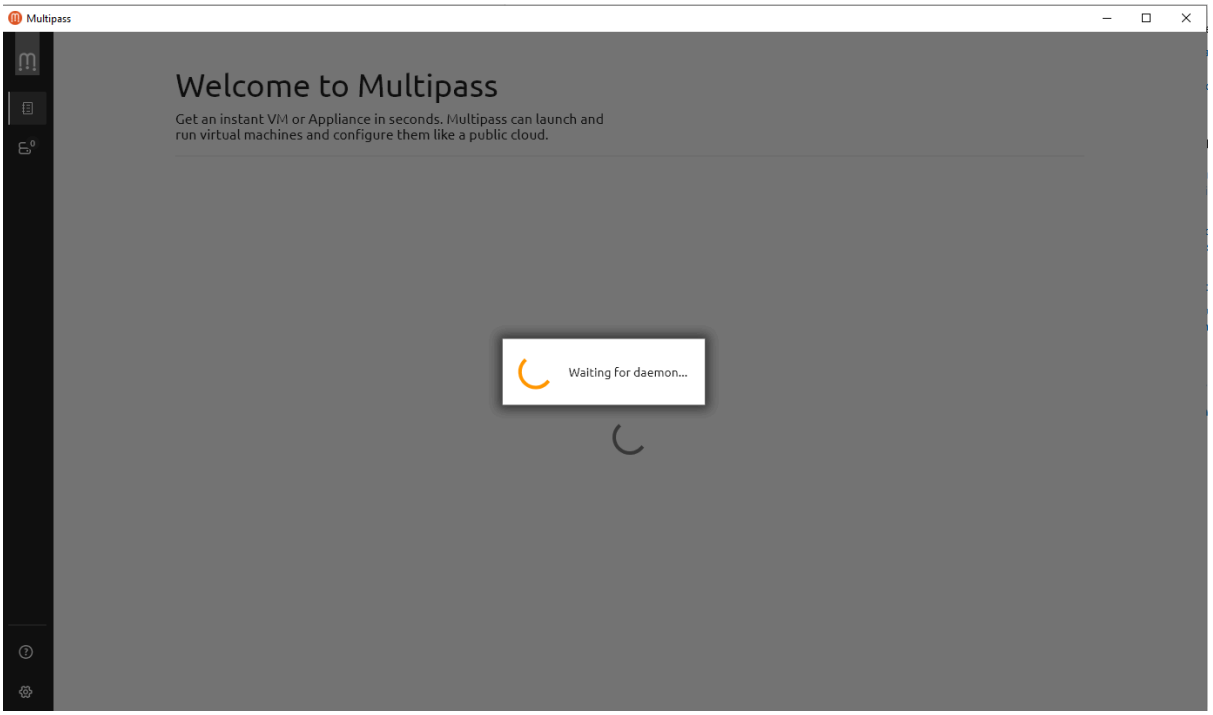


Se selecciona la opción Finish

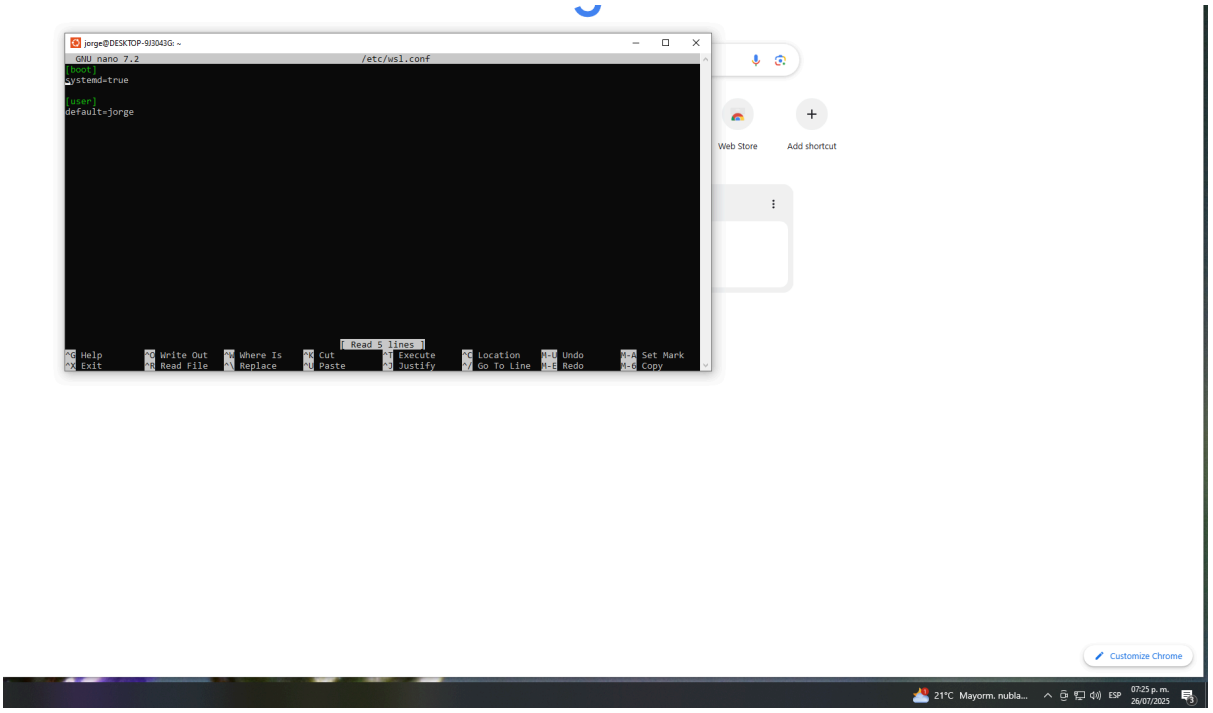


Se ejecuta el aplicativo

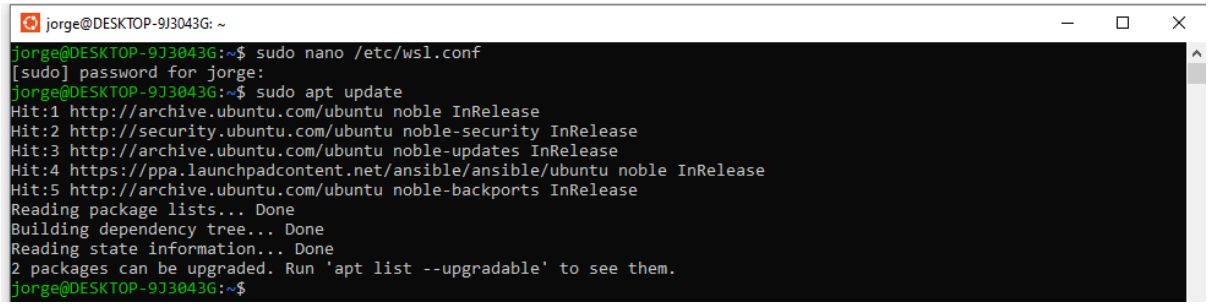
Se quedo pasmado



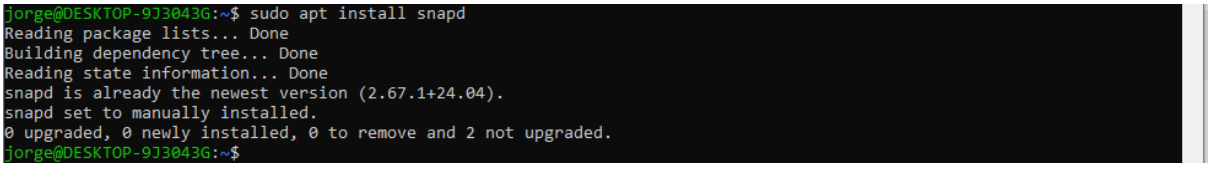
Instale Microk8S en WSL de Windows



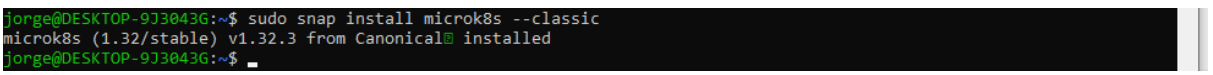
El primer comando dentro de la WSL es verificar si el [boot], la variable systemd=true



Actualizamos los paquetes



Se instala el gestor de paquetes snap para poder descargar microK8s



se ejecuta el comando sudo snap install microk8s

```
jorge@DESKTOP-9J3043G:~$ sudo usermod -a -G microk8s $USER
jorge@DESKTOP-9J3043G:~$ sudo chown -f -R $USER ~/.kube
jorge@DESKTOP-9J3043G:~$ ~~
```

Se añade con el comando sudo usermod -a -G microk8 $USER para la ejecución de MicroK8s

```
jorge@DESKTOP-9J3043G:~$ sudo apt install iptables
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libip4tc2 libip6tc2 libnetfilter-conntrack3 libnfnetlink0 libnftables1 libnftnl11 nftables
Suggested packages:
  firewalld
The following NEW packages will be installed:
  iptables libip4tc2 libip6tc2 libnetfilter-conntrack3 libnfnetlink0 libnftables1 libnftnl11 nftables
0 upgraded, 8 newly installed, 0 to remove and 2 not upgraded.
Need to get 983 kB of archives.
After this operation, 4274 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://archive.ubuntu.com/ubuntu noble/main amd64 libip4tc2 amd64 1.8.10-3ubuntu2 [23.3 kB]
Get:2 http://archive.ubuntu.com/ubuntu noble/main amd64 libip6tc2 amd64 1.8.10-3ubuntu2 [23.7 kB]
Get:3 http://archive.ubuntu.com/ubuntu noble/main amd64 libnfnetlink0 amd64 1.0.2-2build1 [14.8 kB]
Get:4 http://archive.ubuntu.com/ubuntu noble/main amd64 libnetfilter-conntrack3 amd64 1.0.9-6build1 [45.2 kB]
Get:5 http://archive.ubuntu.com/ubuntu noble/main amd64 libnftnl11 amd64 1.2.6-2build1 [66.0 kB]
Get:6 http://archive.ubuntu.com/ubuntu noble/main amd64 iptables amd64 1.8.10-3ubuntu2 [381 kB]
27% [6 iptables 1441 B/381 kB 0%]
```

Se instala IP TABLES

2. Exploración del cluster de Kubernetes (30 puntos)

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl get nodes
NAME              STATUS   ROLES    AGE   VERSION
desktop-9j3043g   Ready    <none>   15m   v1.32.3
jorge@DESKTOP-9J3043G:~$
```

- Lista los nodos que se están ejecutando en el cluster

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl get services --all-namespaces
NAMESPACE     NAME         TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)                  AGE
default       kubernetes   ClusterIP   10.152.183.1     <none>        443/TCP                  22m
kube-system   kube-dns     ClusterIP   10.152.183.10    <none>        53/UDP,53/TCP,9153/TCP   22m
jorge@DESKTOP-9J3043G:~$
```

- Lista los servicios de todos los espacios de nombre

3. Instalación de NGINX en el cluster (20 puntos)

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl create deployment nginx --image=nginx
deployment.apps/nginx created
jorge@DESKTOP-9J3043G:~$
```

- Crea un deployment para implementar una imagen de Nginx

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl get deployments
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
nginx   1/1     1            1           54s
jorge@DESKTOP-9J3043G:~$
```

- Lista los deployments

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl get pods
NAME                     READY   STATUS    RESTARTS   AGE
nginx-5869d7778c-4w8lr   1/1     Running   0          88s
jorge@DESKTOP-9J3043G:~$
```

• Lista los Pods

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl expose deployment nginx --port=80 --type=NodePort
service/nginx exposed
jorge@DESKTOP-9J3043G:~$
```

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl get nodes -o wide
NAME              STATUS   ROLES    AGE   VERSION   INTERNAL-IP     EXTERNAL-IP   OS-IMAGE           KERNEL-VERSION                    CONTAINER-RUNTIME
desktop-9j3043g   Ready    <none>   33m   v1.32.3   172.18.183.17   <none>        Ubuntu 24.04.2 LTS   6.6.87.2-microsoft-standard-WSL2   containerd://1.6.36
jorge@DESKTOP-9J3043G:~$ microk8s kubectl get service nginx
NAME    TYPE       CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
nginx   NodePort   10.152.183.213   <none>        80:31077/TCP   2m26s
jorge@DESKTOP-9J3043G:~$
```

• Obten la IP del nodo y accede al sitio web desde linea de comando con wget

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl port-forward service/nginx 8080:80
Forwarding from 127.0.0.1:8080 -> 80
Forwarding from [::1]:8080 -> 80
Handling connection for 8080
```

TERMINAL 1

```
jorge@DESKTOP-9J3043G:~$ wget http://localhost:8080
--2025-07-26 20:10:04--  http://localhost:8080/
Resolving localhost (localhost)... 127.0.0.1
Connecting to localhost (localhost)|127.0.0.1|:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 615 [text/html]
Saving to: 'index.html'

index.html                  100%[=============================================>]     615  --.-KB/s    in 0s

2025-07-26 20:10:04 (86.5 MB/s) - 'index.html' saved [615/615]

jorge@DESKTOP-9J3043G:~$
```

Se utiliza un proxy para redireccionar por el puerto 8080 el trafico de nginx

4. Escalado de Instancias (20 puntos) o Escala a 5 replicas el pod de nginx.

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl scale deployment nginx --replicas=5
deployment.apps/nginx scaled
jorge@DESKTOP-9J3043G:~$
```

Se escala a 5 replicas

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl get pods
NAME                     READY   STATUS    RESTARTS   AGE
nginx-5869d7778c-2gb96   1/1     Running   0          36s
nginx-5869d7778c-4w8lr   1/1     Running   0          14m
nginx-5869d7778c-bpfvs   1/1     Running   0          36s
nginx-5869d7778c-hffx6   1/1     Running   0          36s
nginx-5869d7778c-llvp2   1/1     Running   0          36s
jorge@DESKTOP-9J3043G:~$
```

Se listan los 5 pods

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl scale deployment nginx --replicas=2
deployment.apps/nginx scaled
jorge@DESKTOP-9J3043G:~$
```

Se reducen los dos pods

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl get pods
NAME                     READY   STATUS    RESTARTS   AGE
nginx-5869d7778c-2gb96   1/1     Running   0          75s
nginx-5869d7778c-4w8lr   1/1     Running   0          14m
jorge@DESKTOP-9J3043G:~$
jorge@DESKTOP-9J3043G:~$
```

Se listan los pods

EXTRA

Aplicando los conceptos aprendidos he agregado esto extra al reto:

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl scale deployment nginx --replicas=5
deployment.apps/nginx scaled
jorge@DESKTOP-9J3043G:~$
```

Se escala a 5 Replicas

Se ejecutara esto en una terminal `for i in {1..10}; do curl -s localhost:8080 | grep title; done`
para generar trafico.

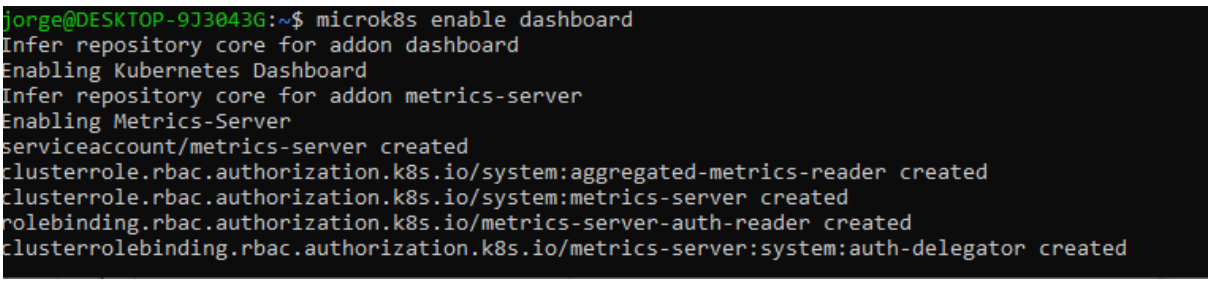
jorge@DESKTOP-9J3043G: ~

```
jorge@DESKTOP-9J3043G:~$ for i in {1..10}; do curl -s localhost:8080 | grep title; done
<title>Welcome to nginx!</title>
<title>Welcome to nginx!</title>
<title>Welcome to nginx!</title>
<title>Welcome to nginx!</title>
<title>Welcome to nginx!</title>
<title>Welcome to nginx!</title>
<title>Welcome to nginx!</title>
<title>Welcome to nginx!</title>
<title>Welcome to nginx!</title>
<title>Welcome to nginx!</title>
jorge@DESKTOP-9J3043G:~$
```

Cambiar la imagen por una más visual como nginxMaster/hello

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl set image deployment/nginx nginx=nginxMaster/hello
deployment.apps/nginx image updated
jorge@DESKTOP-9J3043G:~$
```

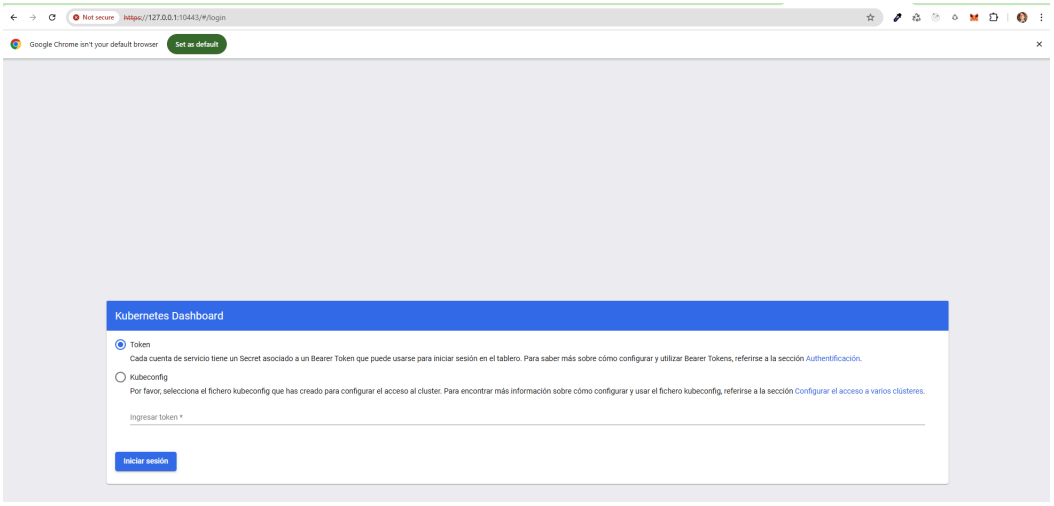
Se genera la imagen nginxMaster



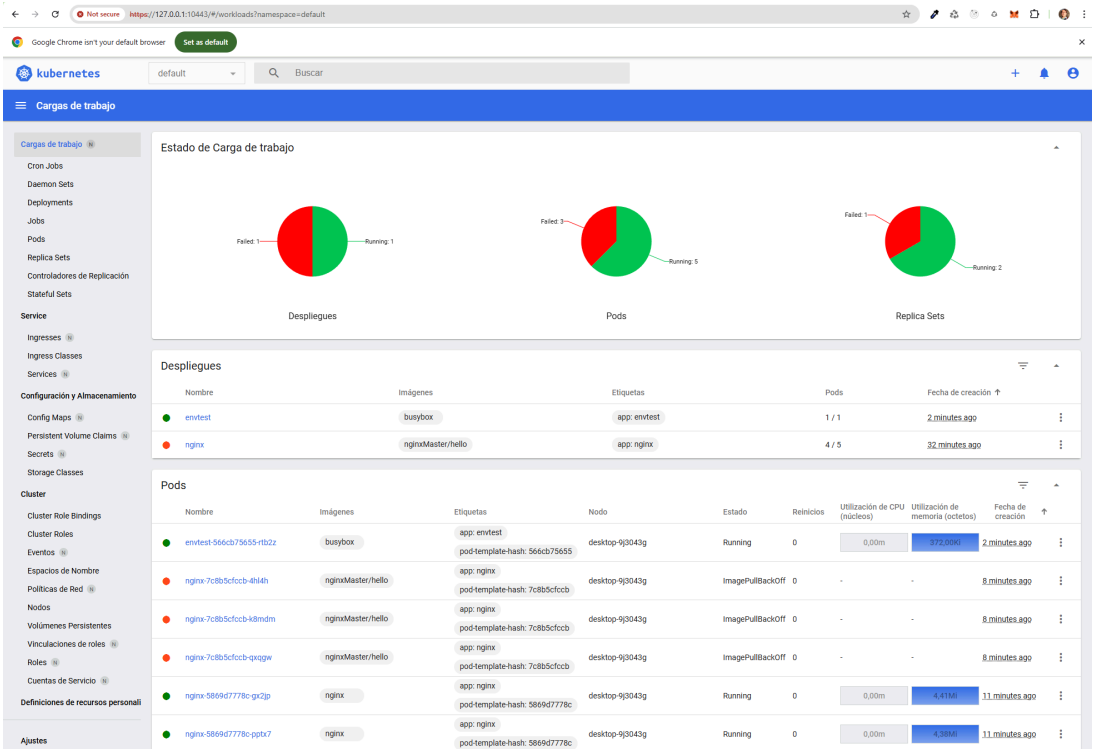
microk8s enable dashboard

Se instala el dashboard

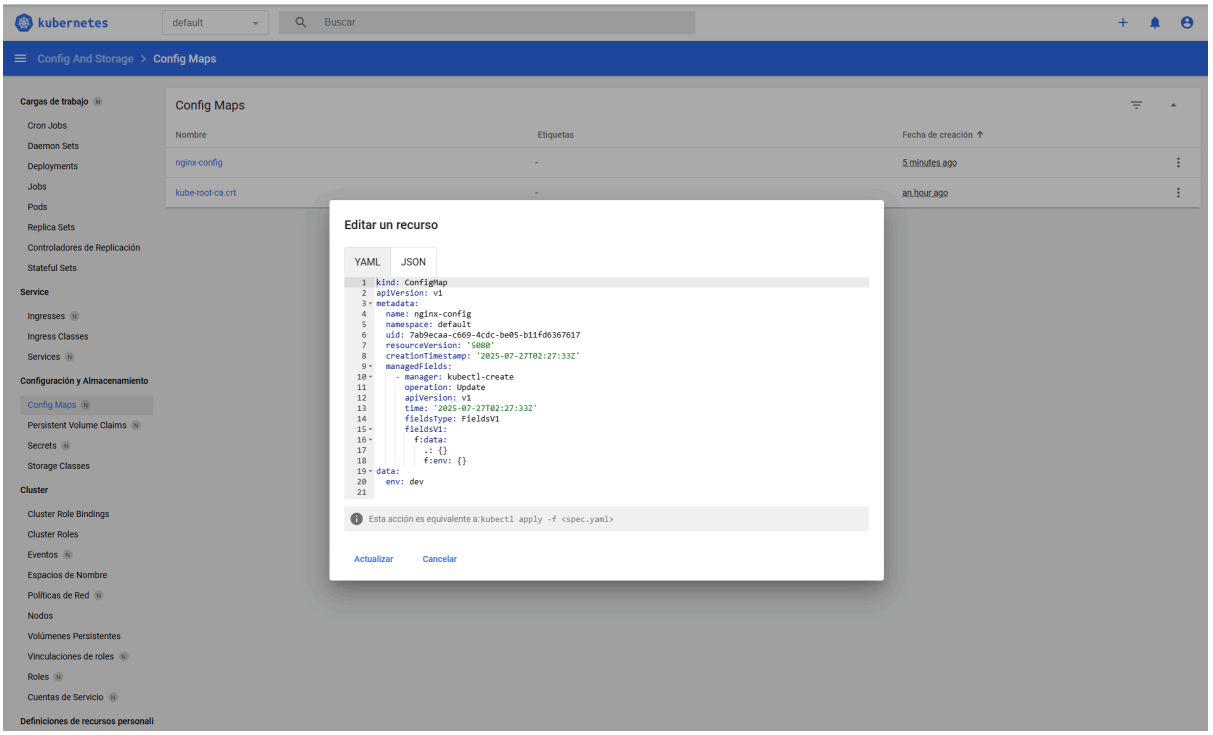
En el proceso de instalación



Se muestra la configuración, se requiere pegar el token para que muestre el tablero de Kubernetes



Se muestra el tablero de Kubernetes



Nos muestra opciones que podemos hacer de manera visual de kubernetes

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl create configmap nginx-config --from-literal=env=dev
```

Se agrega un ConfigMap

```
deployment.apps/nginx image updated
jorge@DESKTOP-9J3043G:~$ microk8s kubectl create configmap nginx-config --from-literal=env=dev
configmap/nginx-config created
jorge@DESKTOP-9J3043G:~$
```

El mensaje de configMap creado

```
jorge@DESKTOP-9J3043G:~$ microk8s kubectl create configmap nginx-config --from-literal=env=dev
configmap/nginx-config created
jorge@DESKTOP-9J3043G:~$ microk8s kubectl create deployment envtest --image=busybox -- sleep 3600
deployment.apps/envtest created
jorge@DESKTOP-9J3043G:~$ microk8s kubectl exec -it $(microk8s kubectl get pods -l app=envtest -o jsonpath='{.items[0].me
tadata.name}') -- printenv
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=envtest-566cb75655-rtb2z
KUBERNETES_PORT=tcp://10.152.183.1:443
KUBERNETES_PORT_443_TCP_PROTO=tcp
NGINX_SERVICE_HOST=10.152.183.213
KUBERNETES_PORT_443_TCP=tcp://10.152.183.1:443
KUBERNETES_PORT_443_TCP_ADDR=10.152.183.1
NGINX_PORT_80_TCP_PORT=80
KUBERNETES_SERVICE_HOST=10.152.183.1
KUBERNETES_SERVICE_PORT_HTTPS=443
NGINX_SERVICE_PORT=80
NGINX_PORT_80_TCP_ADDR=10.152.183.213
KUBERNETES_SERVICE_PORT=443
KUBERNETES_PORT_443_TCP_PORT=443
NGINX_PORT=tcp://10.152.183.213:80
NGINX_PORT_80_TCP=tcp://10.152.183.213:80
NGINX_PORT_80_TCP_PROTO=tcp
TERM=xterm
HOME=/root
jorge@DESKTOP-9J3043G:~$
```

Se muestran conceptos de configuración, entorno y debugging de contenedores

```
jorge@DESKTOP-9J3043G:~$ microk8s enable metrics-server
Infer repository core for addon metrics-server
Addon core/metrics-server is already enabled
jorge@DESKTOP-9J3043G:~$ microk8s kubectl top pods
NAME                        CPU(cores)   MEMORY(bytes)
envtest-566cb75655-rtb2z    0m           0Mi
nginx-5869d7778c-2gb96      0m           4Mi
nginx-5869d7778c-4w8lr      0m           4Mi
nginx-5869d7778c-gx2jp      0m           4Mi
nginx-5869d7778c-pptx7      0m           4Mi
jorge@DESKTOP-9J3043G:~$
```

Se muestra la observabilidad

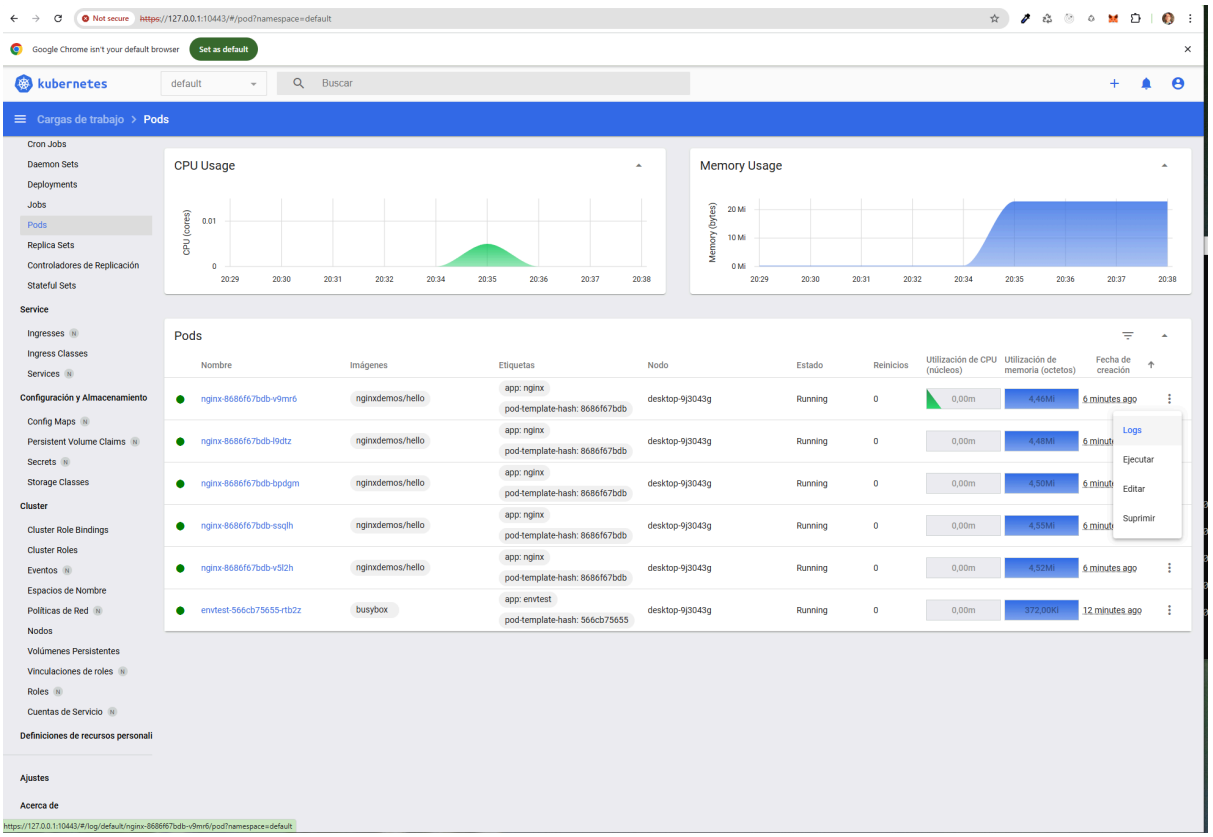
Eventos

Nombre	Motivo	Mensaje	Origen	Sub-objeto	Conteo	Primer visto	último visto ↑
nginx.1855fb10126b1374	ScalingReplicaSet	Scaled down replica set nginx-5869d7778c from 1 to 0	deployment-controller	-	1	a minute ago	a minute ago
nginx.1855fb0f863267d0	ScalingReplicaSet	Scaled up replica set nginx-8686f67bdb from 4 to 5	deployment-controller	-	1	a minute ago	a minute ago
nginx.1855fb0f674eeb40	ScalingReplicaSet	Scaled down replica set nginx-5869d7778c from 4 to 3	deployment-controller	-	1	a minute ago	a minute ago
nginx.1855fb0f701878a4	ScalingReplicaSet	Scaled up replica set nginx-8686f67bdb from 3 to 4	deployment-controller	-	1	a minute ago	a minute ago
nginx.1855fb0f814899f6	ScalingReplicaSet	Scaled down replica set nginx-5869d7778c from 3 to 1	deployment-controller	-	1	a minute ago	a minute ago
nginx.1855fb0d8f99a6f2	ScalingReplicaSet	Scaled down replica set nginx-7c8b5cfccb from 3 to 0	deployment-controller	-	1	a minute ago	a minute ago
nginx.1855fb0d93e70173	ScalingReplicaSet	Scaled up replica set nginx-8686f67bdb from 0 to 3	deployment-controller	-	1	a minute ago	a minute ago

Los eventos en los pods

Para mas pruebas, se instala una herramienta de bechMarking

```
jorge@DESKTOP-9J3043G:~$ sudo snap install hey
[sudo] password for jorge:
hey 0.1.2 from Reda Ahdjoudj (rahdjoudj) installed
jorge@DESKTOP-9J3043G:~$
```

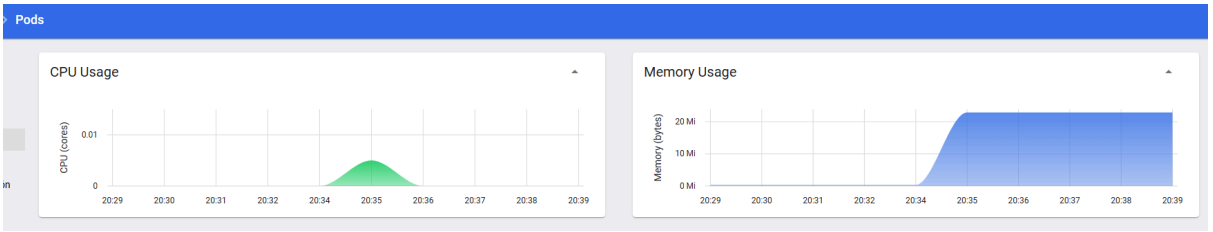


Se realiza la petición a un pod con el siguiente comando:
hey -z 10s -c 5 http://localhost:8080

En logs se puede apreciar la cantidad de peticiones

Logs de nginx en nginx-8686f67...

```
/docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
/docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
/docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
10-listen-on-ipv6-by-default.sh: info: /etc/nginx/conf.d/default.conf is not a file or does not exist
/docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
/docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
/docker-entrypoint.sh: Configuration complete; ready for start up
2025/07/27 02:34:35 [notice] 1#1: using the "epoll" event method
2025/07/27 02:34:35 [notice] 1#1: nginx/1.29.0
2025/07/27 02:34:35 [notice] 1#1: built by gcc 14.2.0 (Alpine 14.2.0)
2025/07/27 02:34:35 [notice] 1#1: OS: Linux 6.6.87.2-microsoft-standard-WSL2
2025/07/27 02:34:35 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 65536:65536
2025/07/27 02:34:35 [notice] 1#1: start worker processes
2025/07/27 02:34:35 [notice] 1#1: start worker process 20
2025/07/27 02:34:35 [notice] 1#1: start worker process 21
2025/07/27 02:34:35 [notice] 1#1: start worker process 22
2025/07/27 02:34:35 [notice] 1#1: start worker process 23
```



Tambien nos muestra la metrica el uso

Muchas gracias, Maestro, por su tiempo y por compartir con nosotros su experiencia. El Master me ha gustado mucho y realmente valoro sus enseñanzas y me inspiro a seguir buscando y aprender de este mundo.