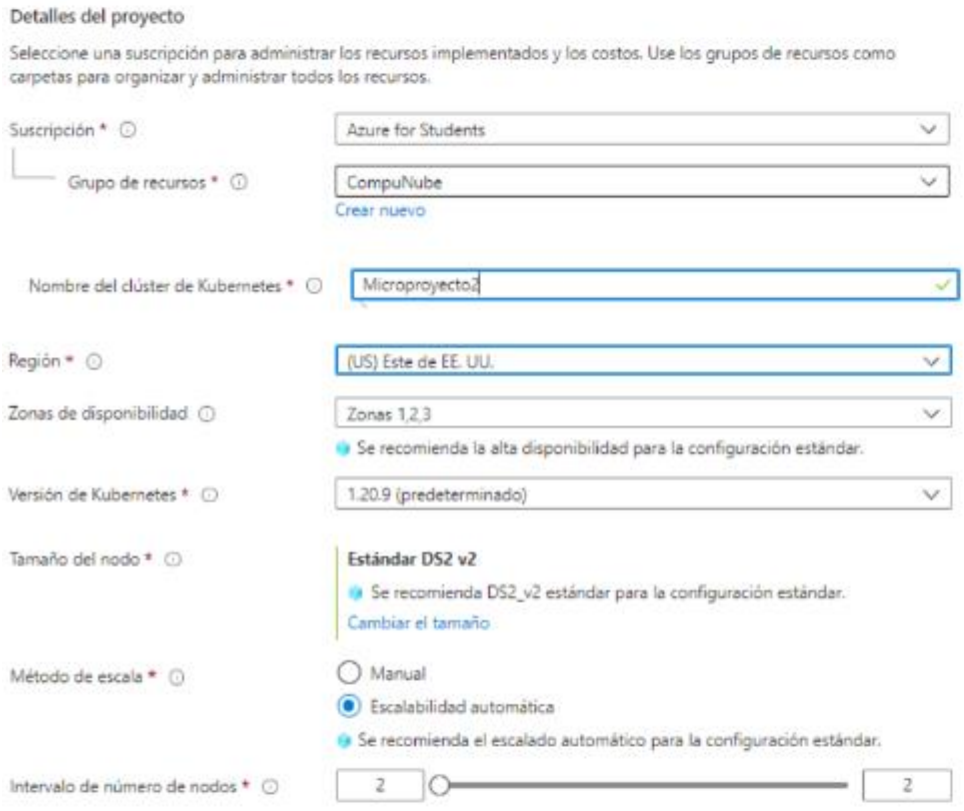
AKS desde portal Azure

Diego Iván Perea Montealegre



Detalles del proyecto

Seleccione una suscripción para administrar los recursos implementados y los costos. Use los grupos de recursos como carpetas para organizar y administrar todos los recursos.

Suscripción * : Azure for Students

Grupo de recursos * : CompuNube

<- minuscula

Nombre del clúster de Kubernetes * : Microproyecto2

<- minuscula

Región * : (US) Este de EE. UU.

Zonas de disponibilidad : Zonas 1,2,3

Se recomienda la alta disponibilidad para la configuración estándar.

Versión de Kubernetes * : 1.20.9 (predeterminado)

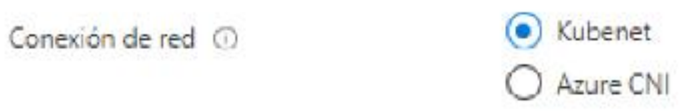
Tamaño del nodo * : **Estándar DS2 v2**

Se recomienda DS2_v2 estándar para la configuración estándar.

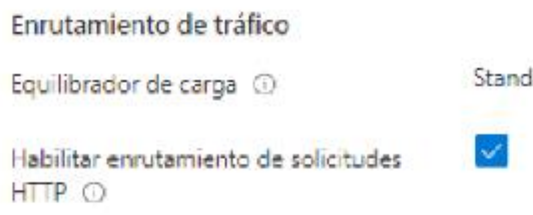
Método de escala * : ○ Manual ● Escalabilidad automática

Se recomienda el escalado automático para la configuración estándar.

Intervalo de número de nodos * : 2 – 2



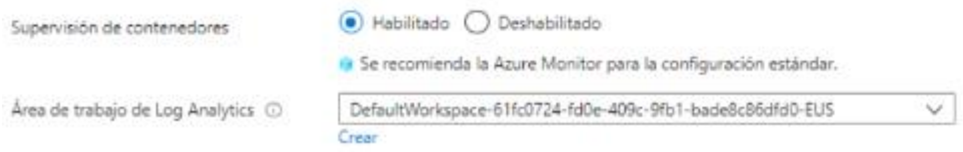
Conexión de red : ● Kubenet ○ Azure CNI



Enrutamiento de tráfico

Equilibrador de carga : Standard

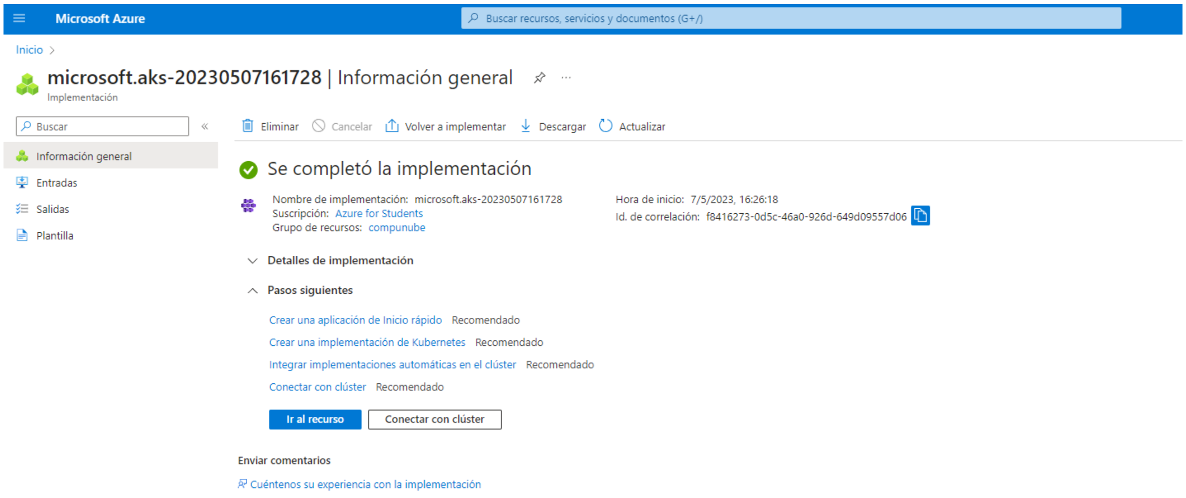
Habilitar enrutamiento de solicitudes HTTP : ☑



Supervisión de contenedores : ● Habilitado ○ Deshabilitado

Se recomienda la Azure Monitor para la configuración estándar.

Área de trabajo de Log Analytics : DefaultWorkspace-61fc0724-fd0e-409c-9fb1-bade8c86dfd0-EUS



microsoft.aks-20230507161728 | Información general

Implementación

Se completó la implementación

Nombre de implementación: microsoft.aks-20230507161728
Suscripción: Azure for Students
Grupo de recursos: compunube

Hora de inicio: 7/5/2023, 16:26:18
Id. de correlación: f8416273-0d5c-46a0-926d-649d09557d06

Detalles de implementación

Pasos siguientes

Crear una aplicación de Inicio rápido Recomendado
Crear una implementación de Kubernetes Recomendado
Integrar implementaciones automáticas en el clúster Recomendado
Conectar con clúster Recomendado

Enviar comentarios

Cuéntenos su experiencia con la implementación

Ingreso de azure por medio del CLI

Aclarar que ya se debe tener instalado minikube , docker y kubernetes en maquina de vagrant (https://github.com/diegoperea20/SpecializationAI/tree/main/CloudComputing/week5/kubernetes-p)

Agrega la clave GPG del repositorio de Azure:

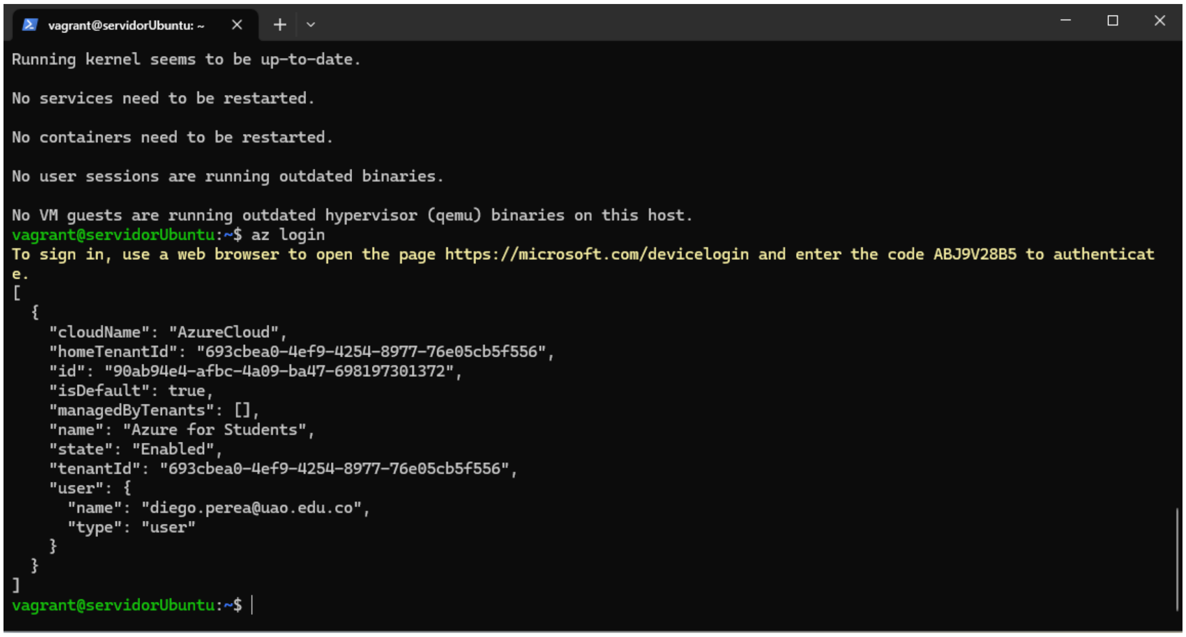
```
curl -sL https://packages.microsoft.com/keys/microsoft.asc | \

gpg --dearmor | \

sudo tee /etc/apt/trusted.gpg.d/microsoft.asc.gpg > /dev/null
```

Agrega el repositorio de Azure CLI a la lista de repositorios de apt-get:

```
AZ_REPO=$(lsb_release -cs)

echo "deb [arch=amd64] https://packages.microsoft.com/repos/azure-cli/ $AZ_REPO main" | \

sudo tee /etc/apt/sources.list.d/azure-cli.list
```

Actualiza la lista de paquetes de apt-get e instala la CLI de Azure:

```
sudo apt-get update

sudo apt-get install azure-cli
```



```
Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
vagrant@servidorUbuntu:~$ az login
To sign in, use a web browser to open the page https://microsoft.com/devicelogin and enter the code ABJ9V28B5 to authenticat
e.
[
  {
    "cloudName": "AzureCloud",
    "homeTenantId": "693cbea0-4ef9-4254-8977-76e05cb5f556",
    "id": "90ab94e4-afbc-4a09-ba47-698197301372",
    "isDefault": true,
    "managedByTenants": [],
    "name": "Azure for Students",
    "state": "Enabled",
    "tenantId": "693cbea0-4ef9-4254-8977-76e05cb5f556",
    "user": {
      "name": "diego.perea@uao.edu.co",
      "type": "user"
    }
  }
]
vagrant@servidorUbuntu:~$
```

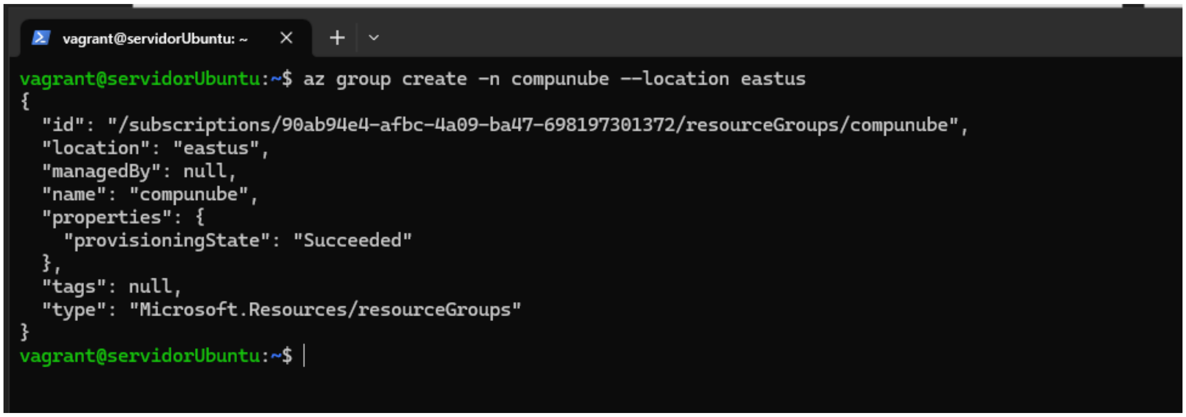
az provider register -n Microsoft.Compute

az provider register -n Microsoft.ContainerService

az provider register -n Microsoft.Network

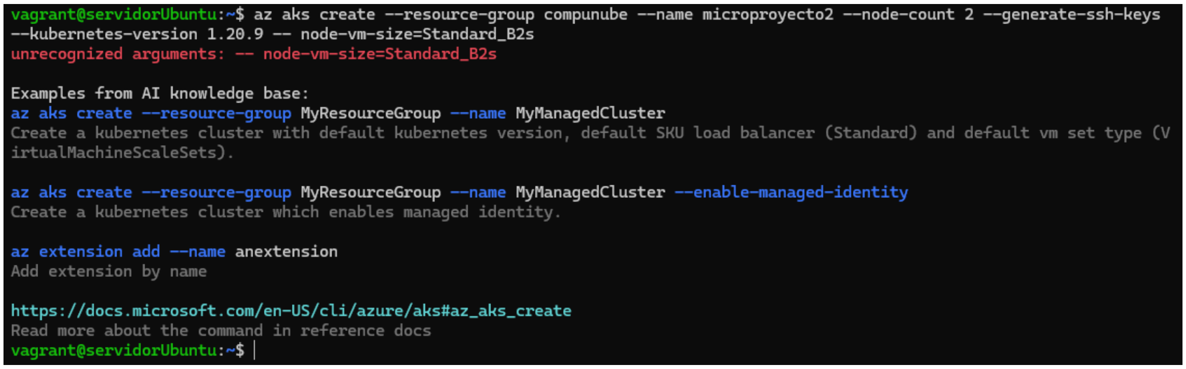
az provider register -n Microsoft.Storage

$ az group create -n compunube --location eastus



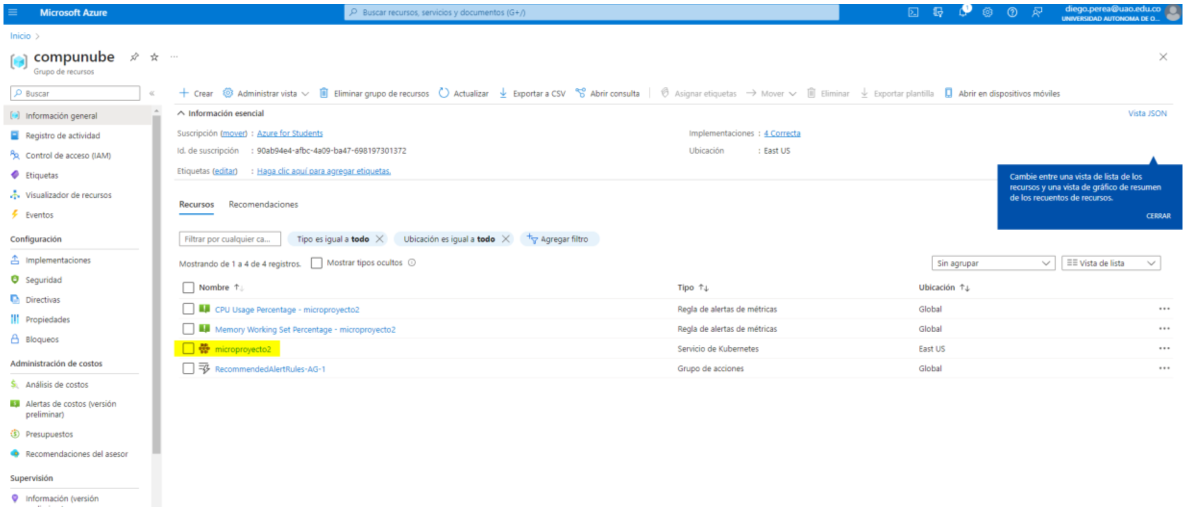
```
vagrant@servidorUbuntu:~$ az group create -n compunube --location eastus
{
  "id": "/subscriptions/90ab94e4-afbc-4a09-ba47-698197301372/resourceGroups/compunube",
  "location": "eastus",
  "managedBy": null,
  "name": "compunube",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
vagrant@servidorUbuntu:~$
```

~$ az aks create --resource-group compunube --name microproyecto2 --node-count 2 --generate-ssh-keys --kubernetes-version 1.20.9 -- node-vm-size=Standard_B2s



```
vagrant@servidorUbuntu:~$ az aks create --resource-group compunube --name microproyecto2 --node-count 2 --generate-ssh-keys
--kubernetes-version 1.20.9 -- node-vm-size=Standard_B2s
unrecognized arguments: -- node-vm-size=Standard_B2s

Examples from AI knowledge base:
az aks create --resource-group MyResourceGroup --name MyManagedCluster
Create a kubernetes cluster with default kubernetes version, default SKU load balancer (Standard) and default vm set type (V
irtualMachineScaleSets).

az aks create --resource-group MyResourceGroup --name MyManagedCluster --enable-managed-identity
Create a kubernetes cluster which enables managed identity.

az extension add --name anextension
Add extension by name

https://docs.microsoft.com/en-US/cli/azure/aks#az_aks_create
Read more about the command in reference docs
vagrant@servidorUbuntu:~$
```

Comprobacion de creación cluster AKS



Suscripción (mover) : Azure for Students

Id. de suscripción : 90ab94e4-afbc-4a09-ba47-698197301372

Etiquetas (editar) : Haga clic aquí para agregar etiquetas.

Implementaciones : 4 Correcta

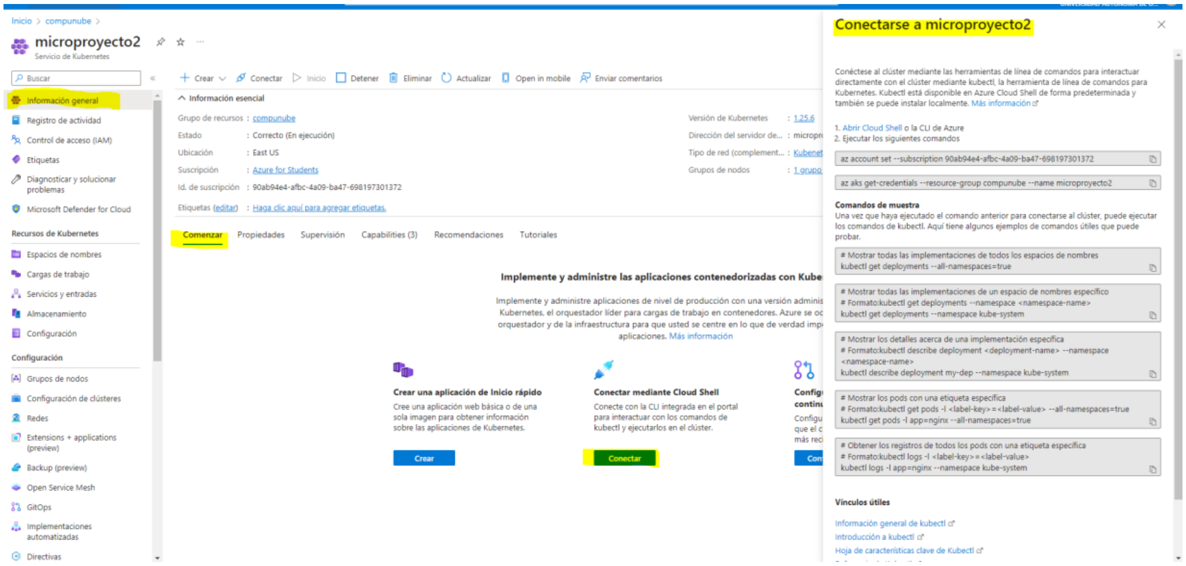
Ubicación : East US

Cambie entre una vista de lista de los recursos y una vista de gráfico de resumen de los recuentos de recursos.

Mostrando de 1 a 4 de 4 registros. □ Mostrar tipos ocultos

□ Nombre ↑↓	Tipo ↑↓	Ubicación ↑↓	
□ CPU Usage Percentage - microproyecto2	Regla de alertas de métricas	Global	···
□ Memory Working Set Percentage - microproyecto2	Regla de alertas de métricas	Global	···
□ microproyecto2	Servicio de Kubernetes	East US	···
□ RecommendedAlertRules-AG-1	Grupo de acciones	Global	···

Acceso a cluster AKS mediante Cloud Shell en Portal Azure



Inicio > compunube >

microproyecto2
Servicio de Kubernetes

Información esencial

Grupo de recursos : compunube
Estado : Correcto (En ejecución)
Ubicación : East US
Suscripción : Azure for Students
Id. de suscripción : 90ab94e4-afbc-4a09-ba47-698197301372
Etiquetas (editar) : Haga clic aquí para agregar etiquetas.

Versión de Kubernetes : 1.25.6
Dirección del servidor de... : micropr
Tipo de red (complement... : Kubenet
Grupos de nodos : 1 grupo

Implemente y administre las aplicaciones contenedorizadas con Kube

Implemente y administre aplicaciones de nivel de producción con una versión adminis Kubernetes, el orquestador líder para cargas de trabajo en contenedores. Azure se oc orquestador y de la infraestructura para que usted se centre en lo que de verdad imp aplicaciones. Más información

Crear una aplicación de Inicio rápido
Cree una aplicación web básica o de una sola imagen para obtener información sobre las aplicaciones de Kubernetes.
Crear

Conectar mediante Cloud Shell
Conecte con la CLI integrada en el portal para interactuar con los comandos de kubectl y ejecutarlos en el clúster.
Conectar

Config contin
Configu que el c más reci
Con

Conectarse a microproyecto2

Conéctese al clúster mediante las herramientas de línea de comandos para interactuar directamente con el clúster mediante kubectl, la herramienta de línea de comandos para Kubernetes. Kubectl está disponible en Azure Cloud Shell de forma predeterminada y también se puede instalar localmente. Más información

1. Abrir Cloud Shell o la CLI de Azure
2. Ejecutar los siguientes comandos

```
az account set --subscription 90ab94e4-afbc-4a09-ba47-698197301372
```

```
az aks get-credentials --resource-group compunube --name microproyecto2
```

Comandos de muestra
Una vez que haya ejecutado el comando anterior para conectarse al clúster, puede ejecutar los comandos de kubectl. Aquí tiene algunos ejemplos de comandos útiles que puede probar.

```
# Mostrar todas las implementaciones de todos los espacios de nombres
kubectl get deployments --all-namespaces=true
```

```
# Mostrar todas las implementaciones de un espacio de nombres específico
# Formato:kubectl get deployments --namespace <namespace-name>
kubectl get deployments --namespace kube-system
```

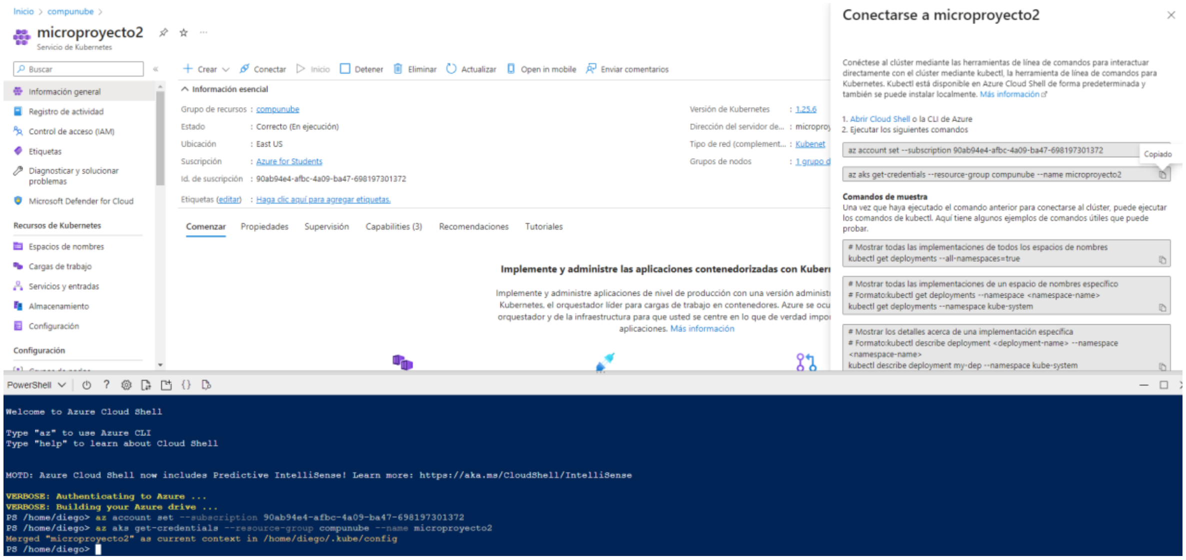
```
# Mostrar los detalles acerca de una implementación específica
# Formato:kubectl describe deployment <deployment-name> --namespace
<namespace-name>
kubectl describe deployment my-dep --namespace kube-system
```

```
# Mostrar los pods con una etiqueta específica
# Formato:kubectl get pods -l <label-key>=<label-value> --all-namespaces=true
kubectl get pods -l app=nginx --all-namespaces=true
```

```
# Obtener los registros de todos los pods con una etiqueta específica
# Formato:kubectl logs -l <label-key>=<label-value>
kubectl logs -l app=nginx --namespace kube-system
```

Vínculos útiles

Información general de kubectl
Introducción a kubectl
Hoja de características clave de Kubectl



Inicio > compunube >

microproyecto2
Servicio de Kubernetes

Grupo de recursos : compunube
Estado : Correcto (En ejecución)
Ubicación : East US
Suscripción : Azure for Students
Id. de suscripción : 90ab94e4-afbc-4a09-ba47-698197301372
Etiquetas (editar) : Haga clic aquí para agregar etiquetas.

Conectarse a microproyecto2

1. Abrir Cloud Shell o la CLI de Azure
2. Ejecutar los siguientes comandos

az account set --subscription 90ab94e4-afbc-4a09-ba47-698197301372
az aks get-credentials --resource-group compunube --name microproyecto2

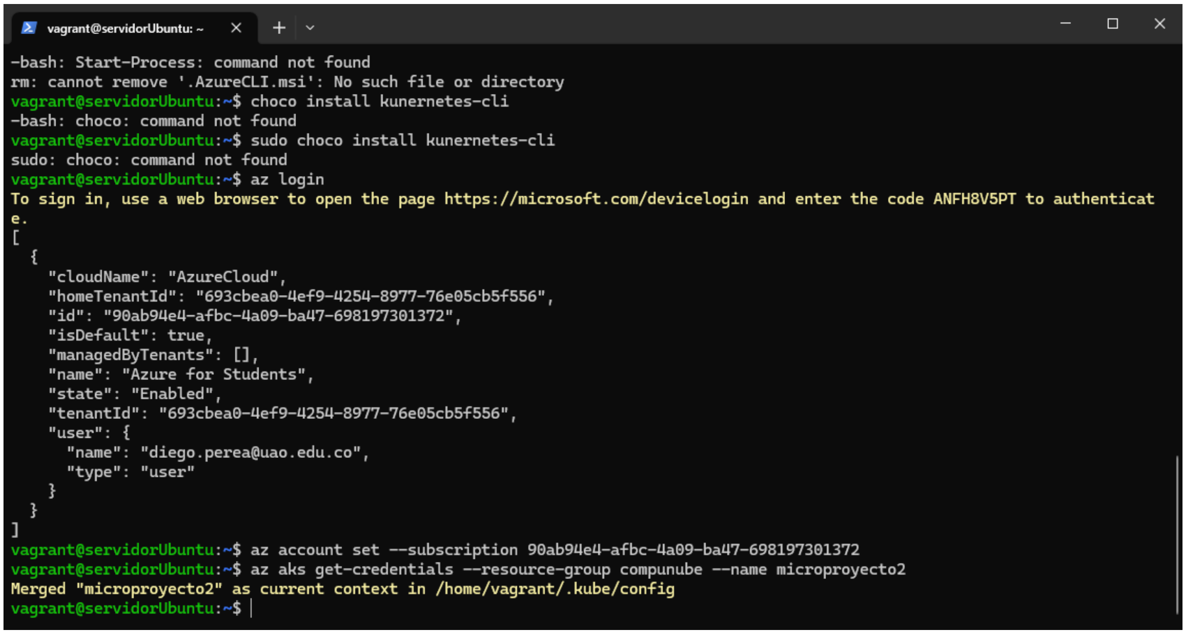
Copiado

```
Welcome to Azure Cloud Shell

Type "az" to use Azure CLI
Type "help" to learn about Cloud Shell


MOTD: Azure Cloud Shell now includes Predictive IntelliSense! Learn more: https://aka.ms/CloudShell/IntelliSense

VERBOSE: Authenticating to Azure ...
VERBOSE: Building your Azure drive ...
PS /home/diego> az account set --subscription 90ab94e4-afbc-4a09-ba47-698197301372
PS /home/diego> az aks get-credentials --resource-group compunube --name microproyecto2
Merged "microproyecto2" as current context in /home/diego/.kube/config
PS /home/diego>
```

Acceso a cluster AKS mediante CLI Azure desde maquina local

az login

se compia y pega lo mismo que se hizo en powershell



```
-bash: Start-Process: command not found
rm: cannot remove '.AzureCLI.msi': No such file or directory
vagrant@servidorUbuntu:~$ choco install kunernetes-cli
-bash: choco: command not found
vagrant@servidorUbuntu:~$ sudo choco install kunernetes-cli
sudo: choco: command not found
vagrant@servidorUbuntu:~$ az login
To sign in, use a web browser to open the page https://microsoft.com/devicelogin and enter the code ANFH8V5PT to authenticat
e.
[
  {
    "cloudName": "AzureCloud",
    "homeTenantId": "693cbea0-4ef9-4254-8977-76e05cb5f556",
    "id": "90ab94e4-afbc-4a09-ba47-698197301372",
    "isDefault": true,
    "managedByTenants": [],
    "name": "Azure for Students",
    "state": "Enabled",
    "tenantId": "693cbea0-4ef9-4254-8977-76e05cb5f556",
    "user": {
      "name": "diego.perea@uao.edu.co",
      "type": "user"
    }
  }
]
vagrant@servidorUbuntu:~$ az account set --subscription 90ab94e4-afbc-4a09-ba47-698197301372
vagrant@servidorUbuntu:~$ az aks get-credentials --resource-group compunube --name microproyecto2
Merged "microproyecto2" as current context in /home/vagrant/.kube/config
vagrant@servidorUbuntu:~$
```

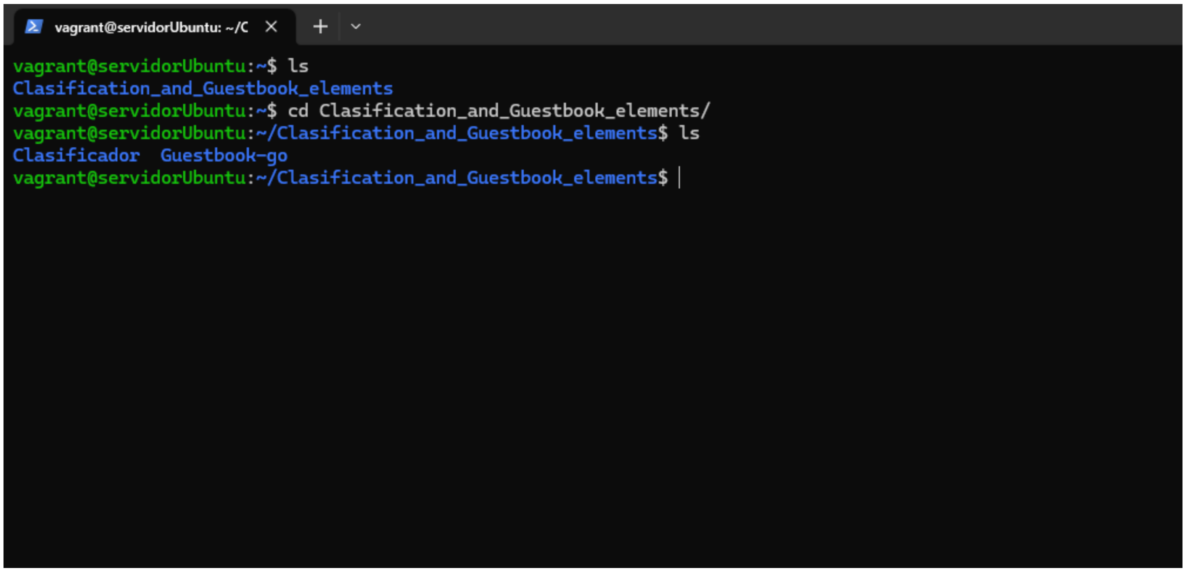
2.CLASIFICADOR DE IMÁGENES CLI AZURE

Se hace git clone al repo https://github.com/diegoperea20/SpecializationAI.git

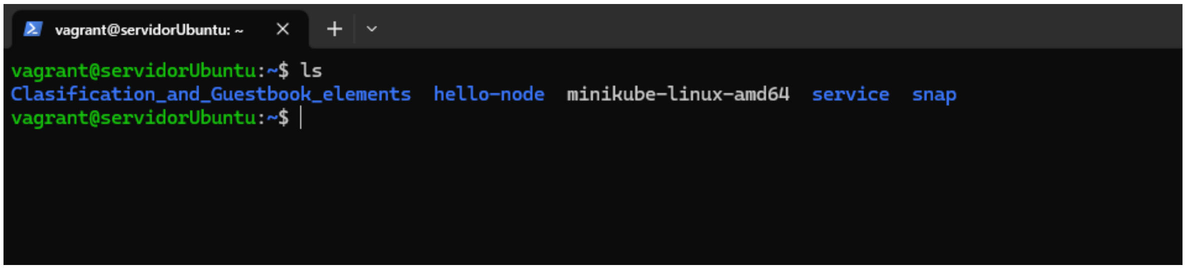
y se mueve "Clasification_and_Guestbook_elements"hacia root

https://github.com/diegoperea20/SpecializationAI/tree/main/CloudComputing/week6/miniproject2/Clasification_and_Guestbook_elements

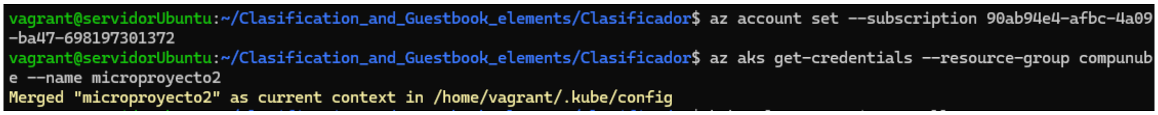
y se elimina lo demás



```
vagrant@servidorUbuntu:~$ ls
Clasification_and_Guestbook_elements
vagrant@servidorUbuntu:~$ cd Clasification_and_Guestbook_elements/
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements$ ls
Clasificador  Guestbook-go
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements$
```

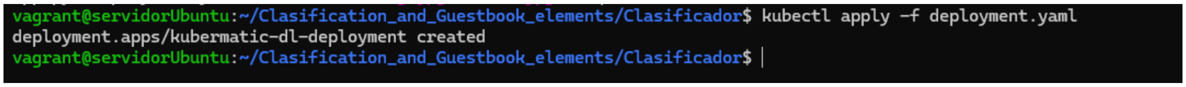


```
vagrant@servidorUbuntu:~$ ls
Clasification_and_Guestbook_elements  hello-node  minikube-linux-amd64  service  snap
vagrant@servidorUbuntu:~$
```



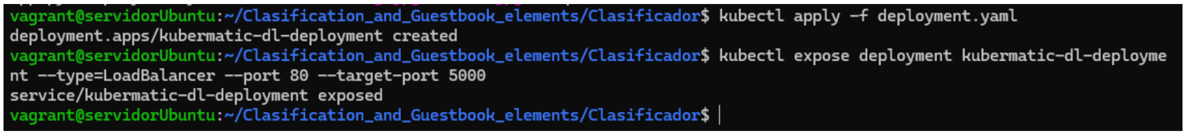
```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$ az account set --subscription 90ab94e4-afbc-4a09
-ba47-698197301372
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$ az aks get-credentials --resource-group compunub
e --name microproyecto2
Merged "microproyecto2" as current context in /home/vagrant/.kube/config
```

kubectl apply -f deployment.yaml



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$ kubectl apply -f deployment.yaml
deployment.apps/kubermatic-dl-deployment created
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$
```

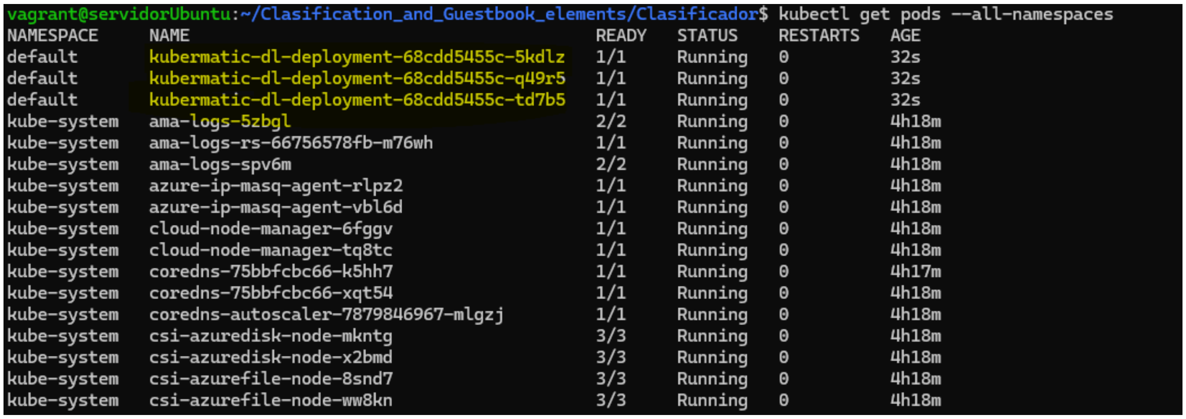
$ kubectl expose deployment kubermatic-dl-deployment --type=LoadBalancer --port 80 --target-port 5000



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$ kubectl apply -f deployment.yaml
deployment.apps/kubermatic-dl-deployment created
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$ kubectl expose deployment kubermatic-dl-deployme
nt --type=LoadBalancer --port 80 --target-port 5000
service/kubermatic-dl-deployment exposed
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$
```

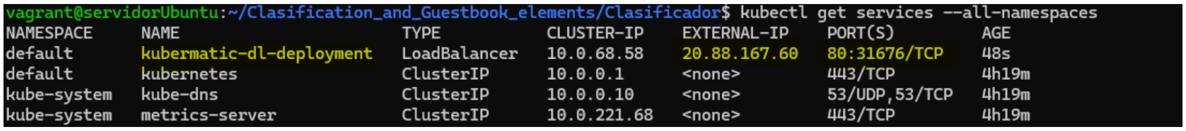
Verificar con kubectl get pods ,services y deployment esten configurados

kubectl get pods --all-namespaces



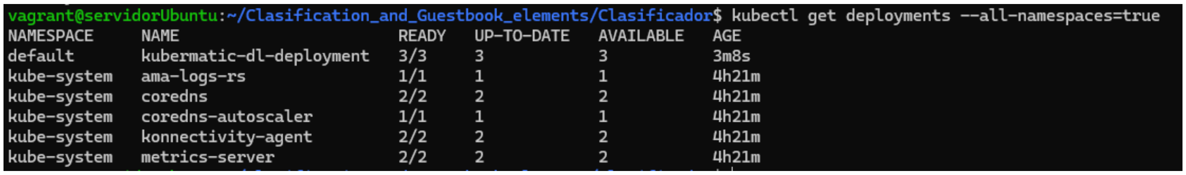
```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$ kubectl get pods --all-namespaces
NAMESPACE     NAME                                       READY   STATUS    RESTARTS   AGE
default       kubermatic-dl-deployment-68cdd5455c-5kdlz  1/1     Running   0          32s
default       kubermatic-dl-deployment-68cdd5455c-q49r5  1/1     Running   0          32s
default       kubermatic-dl-deployment-68cdd5455c-td7b5  1/1     Running   0          32s
kube-system   ama-logs-5zbgl                             2/2     Running   0          4h18m
kube-system   ama-logs-rs-66756578fb-m76wh               1/1     Running   0          4h18m
kube-system   ama-logs-spv6m                             2/2     Running   0          4h18m
kube-system   azure-ip-masq-agent-rlpz2                  1/1     Running   0          4h18m
kube-system   azure-ip-masq-agent-vbl6d                  1/1     Running   0          4h18m
kube-system   cloud-node-manager-6fggv                   1/1     Running   0          4h18m
kube-system   cloud-node-manager-tq8tc                   1/1     Running   0          4h18m
kube-system   coredns-75bbfcbc66-k5hh7                   1/1     Running   0          4h17m
kube-system   coredns-75bbfcbc66-xqt54                   1/1     Running   0          4h18m
kube-system   coredns-autoscaler-7879846967-mlgzj        1/1     Running   0          4h18m
kube-system   csi-azuredisk-node-mkntg                   3/3     Running   0          4h18m
kube-system   csi-azuredisk-node-x2bmd                   3/3     Running   0          4h18m
kube-system   csi-azurefile-node-8snd7                   3/3     Running   0          4h18m
kube-system   csi-azurefile-node-ww8kn                   3/3     Running   0          4h18m
```

kubectl get services --all-namespaces



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$ kubectl get services --all-namespaces
NAMESPACE     NAME                       TYPE           CLUSTER-IP     EXTERNAL-IP     PORT(S)         AGE
default       kubermatic-dl-deployment   LoadBalancer   10.0.68.58     20.88.167.60    80:31676/TCP    48s
default       kubernetes                 ClusterIP      10.0.0.1       <none>          443/TCP         4h19m
kube-system   kube-dns                   ClusterIP      10.0.0.10      <none>          53/UDP,53/TCP   4h19m
kube-system   metrics-server             ClusterIP      10.0.221.68    <none>          443/TCP         4h19m
```

kubectl get deployments --all-namespaces=true



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$ kubectl get deployments --all-namespaces=true
NAMESPACE     NAME                       READY   UP-TO-DATE   AVAILABLE   AGE
default       kubermatic-dl-deployment   3/3     3            3           3m8s
kube-system   ama-logs-rs                1/1     1            1           4h21m
kube-system   coredns                    2/2     2            2           4h21m
kube-system   coredns-autoscaler         1/1     1            1           4h21m
kube-system   konnectivity-agent         2/2     2            2           4h21m
kube-system   metrics-server             2/2     2            2           4h21m
```

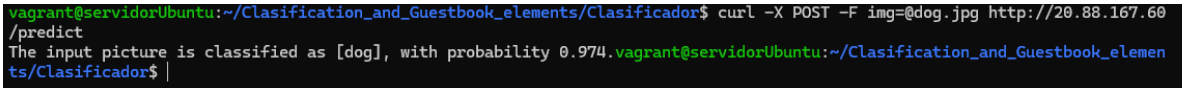
Finalmente hacemos dos pruebas con curl pasándole primero una imagen local de un perro para que la aplicación realice la predicción. La segunda prueba es otro curl, pero esta vez le pasamos una imagen de un caballo como vemos a continuación:

Colocar la external-ip de kubectl get services --all-namespaces

En este caso es EXTERNAL-IP =20.88.167.60

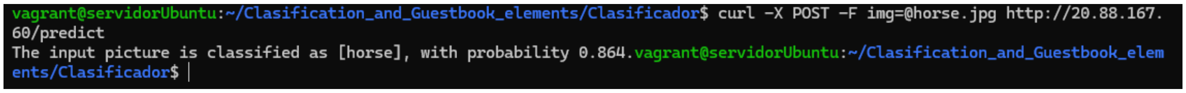
$ curl -X POST -F img=@dog.jpg http: //EXTERNAL-IP/predict

$ curl -X POST -F img=@dog.jpg http://20.88.167.60/predict



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$ curl -X POST -F img=@dog.jpg http://20.88.167.60
/predict
The input picture is classified as [dog], with probability 0.974.vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elemen
ts/Clasificador$ |
```

$ curl -X POST -F img=@horse.jpg http://20.88.167.60/predict

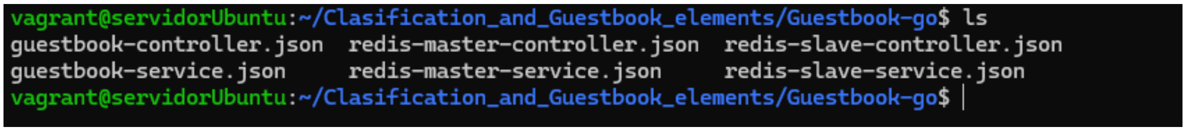


```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Clasificador$ curl -X POST -F img=@horse.jpg http://20.88.167.
60/predict
The input picture is classified as [horse], with probability 0.864.vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elem
ents/Clasificador$ |
```

3.Aplicacion de interés Implementación del servicio web Guestbook Cli Azure

Aplicación guestbook-go: https://github.com/kubernetes/examples/tree/master/guestbook-go

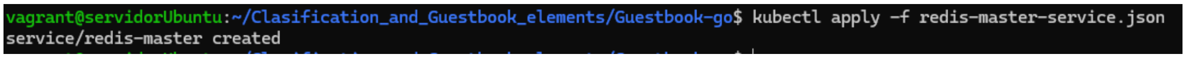
Ir a la carpeta Guestbook-go



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Guestbook-go$ ls
guestbook-controller.json  redis-master-controller.json  redis-slave-controller.json
guestbook-service.json     redis-master-service.json     redis-slave-service.json
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Guestbook-go$ |
```

Creacion de servicio Redis , (Servicio kubernetes como balanceador)

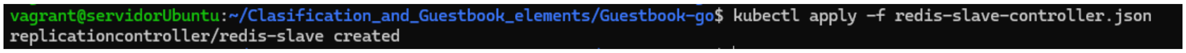
$ kubectl apply -f redis-master-service.json



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Guestbook-go$ kubectl apply -f redis-master-service.json
service/redis-master created
```

Creación los pods esclavos de redis , controlador de responsable de manejar varias instacias de un pod duplicado

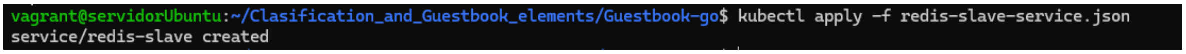
§ kubectl apply -f redis-slave-controller.json



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Guestbook-go$ kubectl apply -f redis-slave-controller.json
replicationcontroller/redis-slave created
```

Creacion servicio esclavo de redis , (balancea carga de clientes)

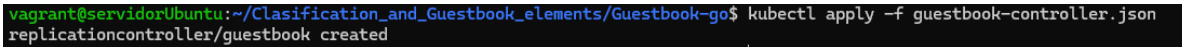
kubectl apply -f redis-slave-service.json



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Guestbook-go$ kubectl apply -f redis-slave-service.json
service/redis-slave created
```

Creamos los pods del guestbook, los servicios maestro o esclavo dependiendo de si la solicitud es de lectura o escritura.

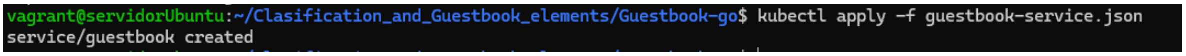
$ kubectl apply -f guestbook-controller.json



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Guestbook-go$ kubectl apply -f guestbook-controller.json
replicationcontroller/guestbook created
```

agrupar los pods del guestbook para que sea visible externamente para que sea tipo baleanceador

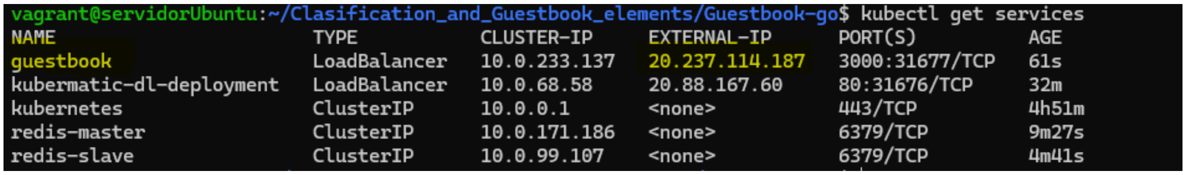
$ kubectl apply -f guestbook-service.json



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Guestbook-go$ kubectl apply -f guestbook-service.json
service/guestbook created
```

Visualizar los pods,servicios y controladores

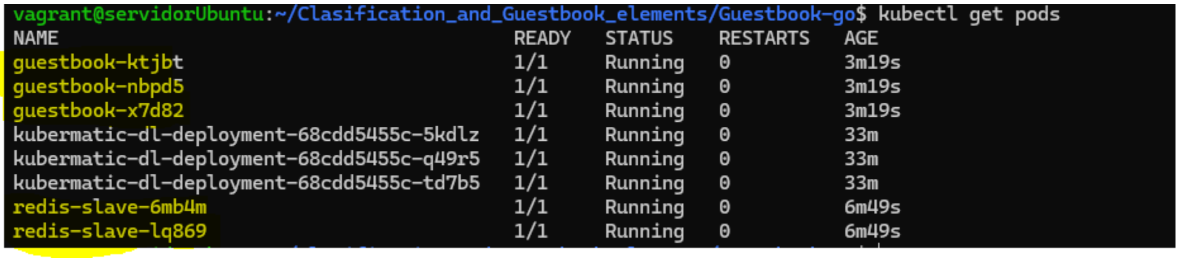
$ kubectl get services



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Guestbook-go$ kubectl get services
NAME                       TYPE           CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
guestbook                  LoadBalancer   10.0.233.137    20.237.114.187   3000:31677/TCP   61s
kubermatic-dl-deployment   LoadBalancer   10.0.68.58      20.88.167.60     80:31676/TCP     32m
kubernetes                 ClusterIP      10.0.0.1        <none>           443/TCP          4h51m
redis-master               ClusterIP      10.0.171.186    <none>           6379/TCP         9m27s
redis-slave                ClusterIP      10.0.99.107     <none>           6379/TCP         4m41s
```

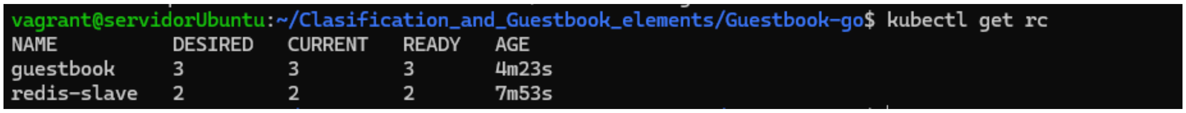
Ip externa: 20.237.114.187   en el puerto 3000

$ kubectl get pods



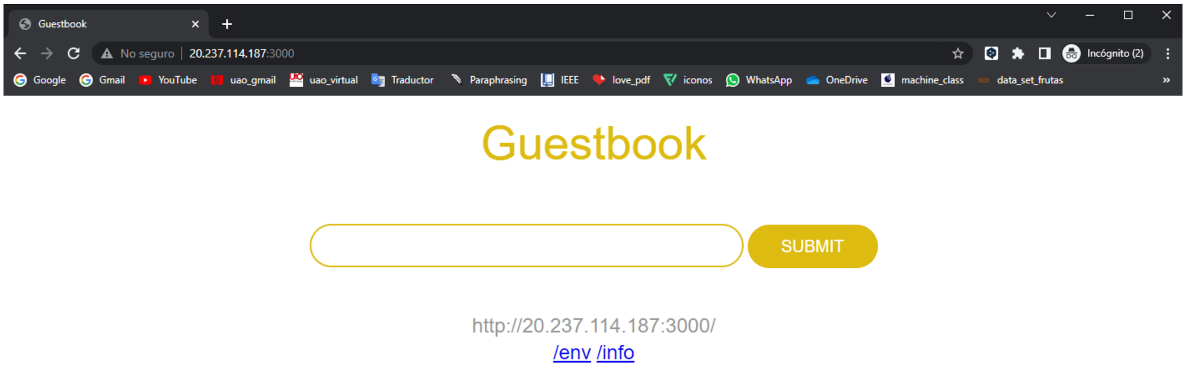
```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Guestbook-go$ kubectl get pods
NAME                                        READY   STATUS    RESTARTS   AGE
guestbook-ktjbt                             1/1     Running   0          3m19s
guestbook-nbpd5                             1/1     Running   0          3m19s
guestbook-x7d82                             1/1     Running   0          3m19s
kubermatic-dl-deployment-68cdd5455c-5kdlz   1/1     Running   0          33m
kubermatic-dl-deployment-68cdd5455c-q49r5   1/1     Running   0          33m
kubermatic-dl-deployment-68cdd5455c-td7b5   1/1     Running   0          33m
redis-slave-6mb4m                           1/1     Running   0          6m49s
redis-slave-lq869                           1/1     Running   0          6m49s
```

$ kubectl get rc



```
vagrant@servidorUbuntu:~/Clasification_and_Guestbook_elements/Guestbook-go$ kubectl get rc
NAME          DESIRED   CURRENT   READY   AGE
guestbook     3         3         3       4m23s
redis-slave   2         2         2       7m53s
```

Se pude ver la aplicación en el navegador web con la Ip externa: 20.237.114.187   en el puerto 3000

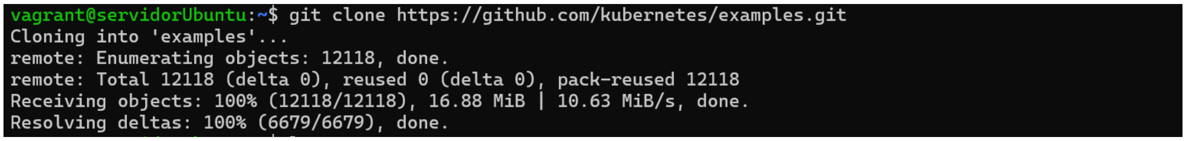


http://20.237.114.187:3000/
/env /info

Sí no sirve hacer el siguiente

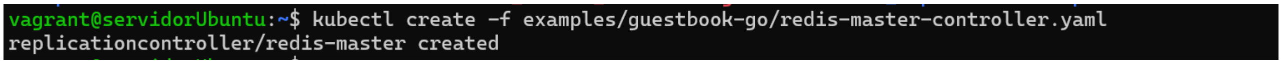
CON EL ACTUALIZADO

git clone https://github.com/kubernetes/examples.git



```
vagrant@servidorUbuntu:~$ git clone https://github.com/kubernetes/examples.git
Cloning into 'examples'...
remote: Enumerating objects: 12118, done.
remote: Total 12118 (delta 0), reused 0 (delta 0), pack-reused 12118
Receiving objects: 100% (12118/12118), 16.88 MiB | 10.63 MiB/s, done.
Resolving deltas: 100% (6679/6679), done.
```

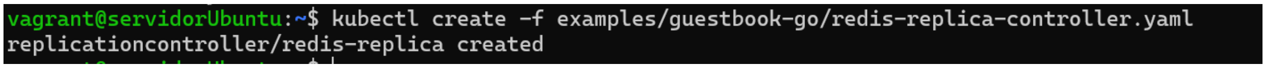
kubectl create -f examples/guestbook-go/redis-master-controller.yaml



```
vagrant@servidorUbuntu:~$ kubectl create -f examples/guestbook-go/redis-master-controller.yaml
replicationcontroller/redis-master created
```

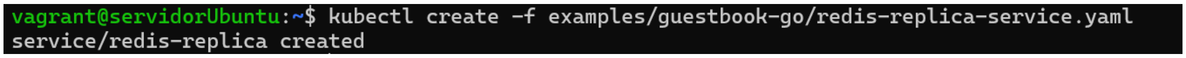
kubectl create -f examples/guestbook-go/redis-master-service.yaml

kubectl create -f examples/guestbook-go/redis-replica-controller.yaml



```
vagrant@servidorUbuntu:~$ kubectl create -f examples/guestbook-go/redis-replica-controller.yaml
replicationcontroller/redis-replica created
```

kubectl create -f examples/guestbook-go/redis-replica-service.yaml

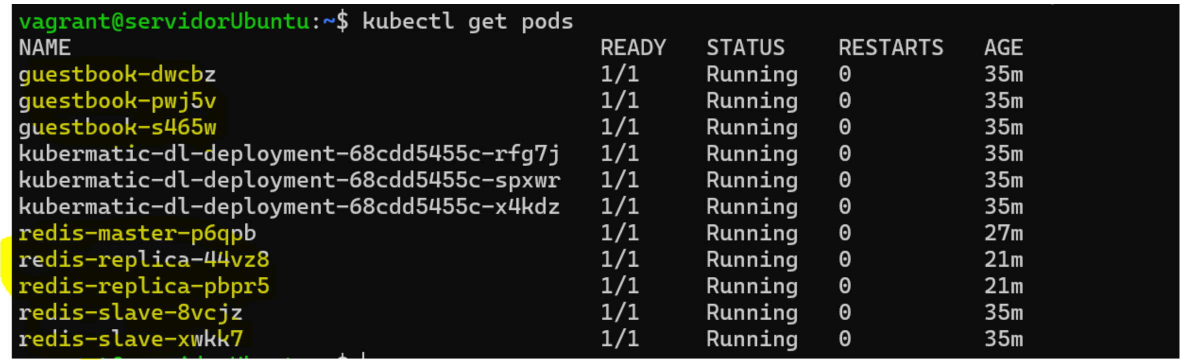


```
vagrant@servidorUbuntu:~$ kubectl create -f examples/guestbook-go/redis-replica-service.yaml
service/redis-replica created
```

kubectl create -f examples/guestbook-go/guestbook-controller.yaml

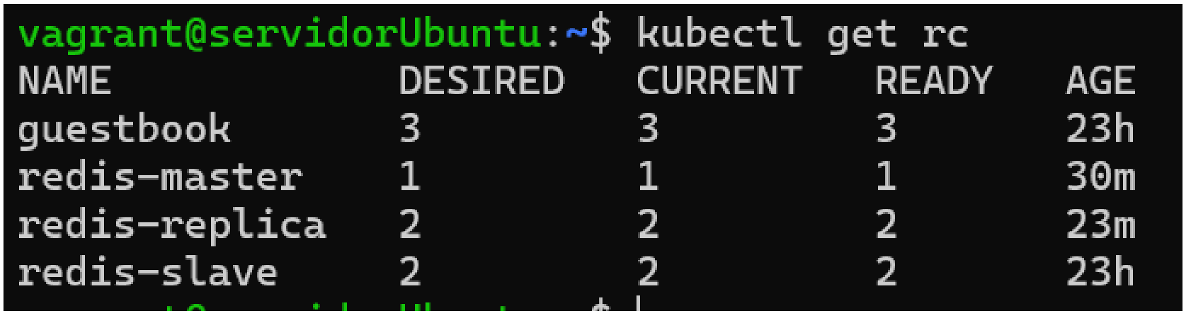
kubectl create -f examples/guestbook-go/guestbook-service.yaml

kubectl get pods



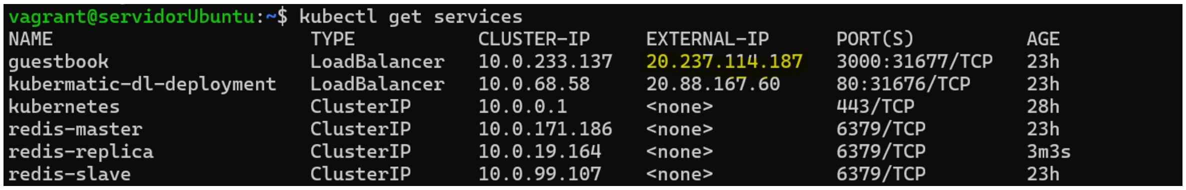
```
vagrant@servidorUbuntu:~$ kubectl get pods
NAME                                      READY   STATUS    RESTARTS   AGE
guestbook-dwcbz                           1/1     Running   0          35m
guestbook-pwj5v                           1/1     Running   0          35m
guestbook-s465w                           1/1     Running   0          35m
kubermatic-dl-deployment-68cdd5455c-rfg7j 1/1     Running   0          35m
kubermatic-dl-deployment-68cdd5455c-spxwr 1/1     Running   0          35m
kubermatic-dl-deployment-68cdd5455c-x4kdz 1/1     Running   0          35m
redis-master-p6qpb                        1/1     Running   0          27m
redis-replica-44vz8                       1/1     Running   0          21m
redis-replica-pbpr5                       1/1     Running   0          21m
redis-slave-8vcjz                         1/1     Running   0          35m
redis-slave-xwkk7                         1/1     Running   0          35m
```

kubectl get rc



```
vagrant@servidorUbuntu:~$ kubectl get rc
NAME            DESIRED   CURRENT   READY   AGE
guestbook       3         3         3       23h
redis-master    1         1         1       30m
redis-replica   2         2         2       23m
redis-slave     2         2         2       23h
```

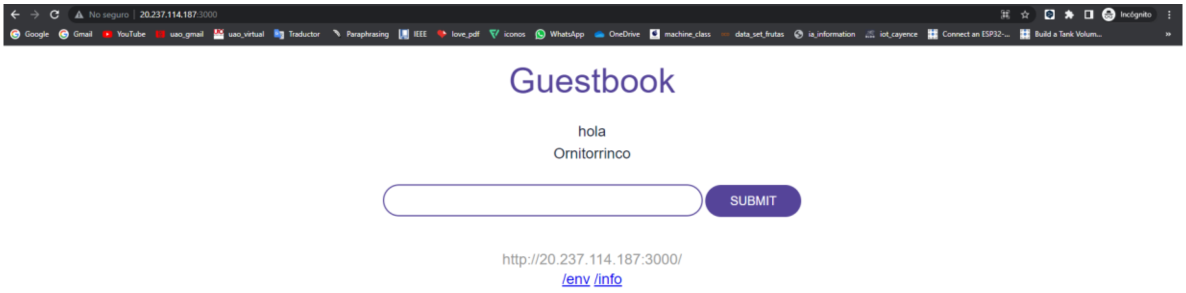
kubectl get services



```
vagrant@servidorUbuntu:~$ kubectl get services
NAME                       TYPE           CLUSTER-IP     EXTERNAL-IP      PORT(S)          AGE
guestbook                  LoadBalancer   10.0.233.137   20.237.114.187   3000:31677/TCP   23h
kubermatic-dl-deployment   LoadBalancer   10.0.68.58     20.88.167.60     80:31676/TCP     23h
kubernetes                 ClusterIP      10.0.0.1       <none>           443/TCP          28h
redis-master               ClusterIP      10.0.171.186   <none>           6379/TCP         23h
redis-replica              ClusterIP      10.0.19.164    <none>           6379/TCP         3m3s
redis-slave                ClusterIP      10.0.99.107    <none>           6379/TCP         23h
```

CON LA IP EXTERNA EN este caso 20.237.114.187

En navegador 20.237.114.187:3000

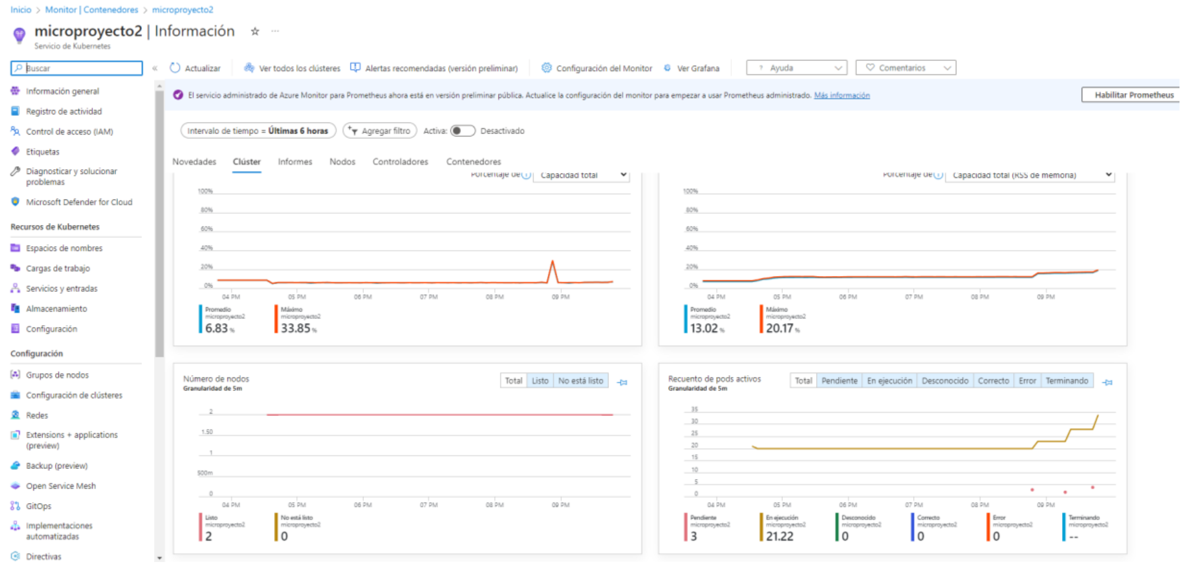


Guestbook

hola
Ornitorrinco

http://20.237.114.187:3000/
/env /info

4.Supervision y monitoreo



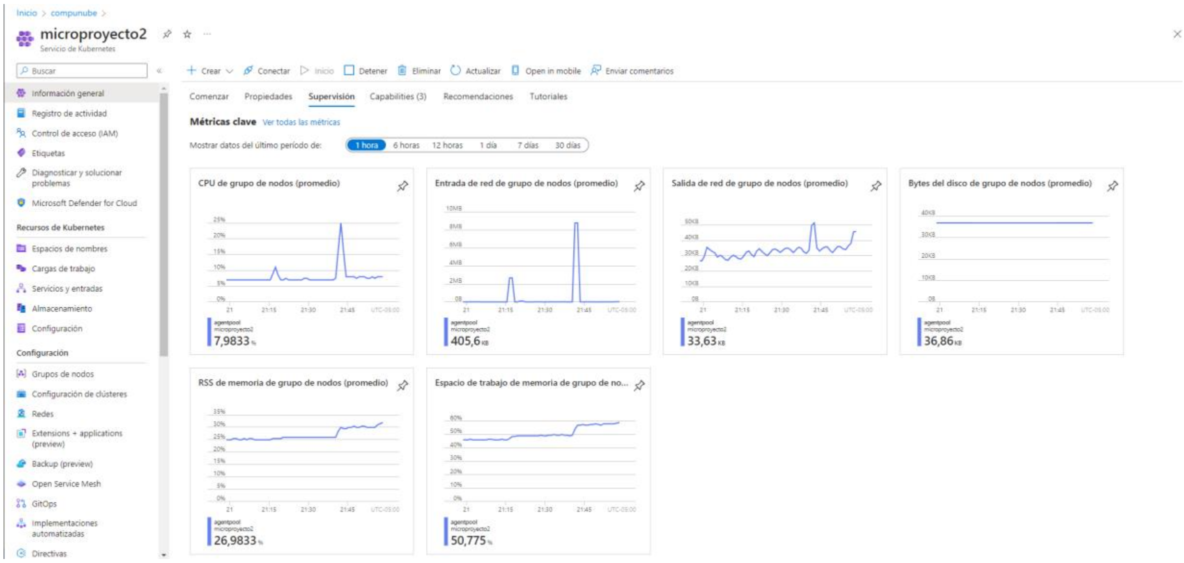
El servicio administrado de Azure Monitor para Prometheus ahora está en versión preliminar pública. Actualice la configuración del monitor para empezar a usar Prometheus administrado. Más información

Intervalo de tiempo = **Últimas 6 horas**

Activa: Desactivado

Porcentaje de: Capacidad total

Porcentaje de: Capacidad total (RSS de memoria)



Métricas clave Ver todas las métricas

Mostrar datos del último período de: 1 hora

CPU de grupo de nodos (promedio)
agentpool microproyecto2
7,9833 %

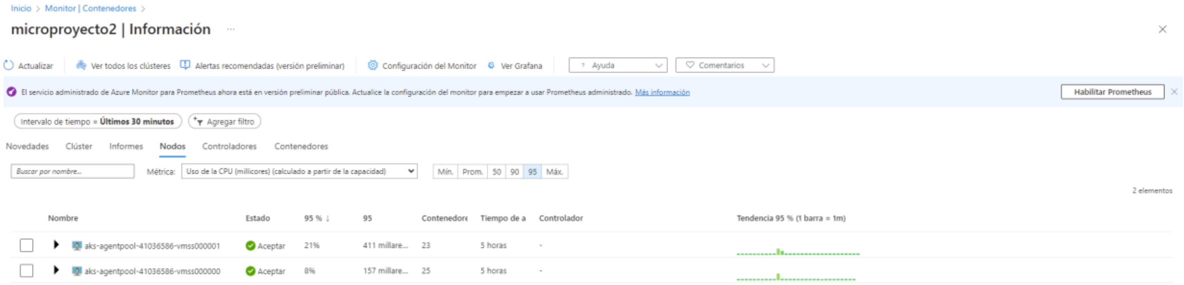
Entrada de red de grupo de nodos (promedio)
agentpool microproyecto2
405,6 KB

Salida de red de grupo de nodos (promedio)
agentpool microproyecto2
33,63 KB

Bytes del disco de grupo de nodos (promedio)
agentpool microproyecto2
36,86 KB

RSS de memoria de grupo de nodos (promedio)
agentpool microproyecto2
26,9833 %

Espacio de trabajo de memoria de grupo de no...
agentpool microproyecto2
50,775 %



microproyecto2 | Información

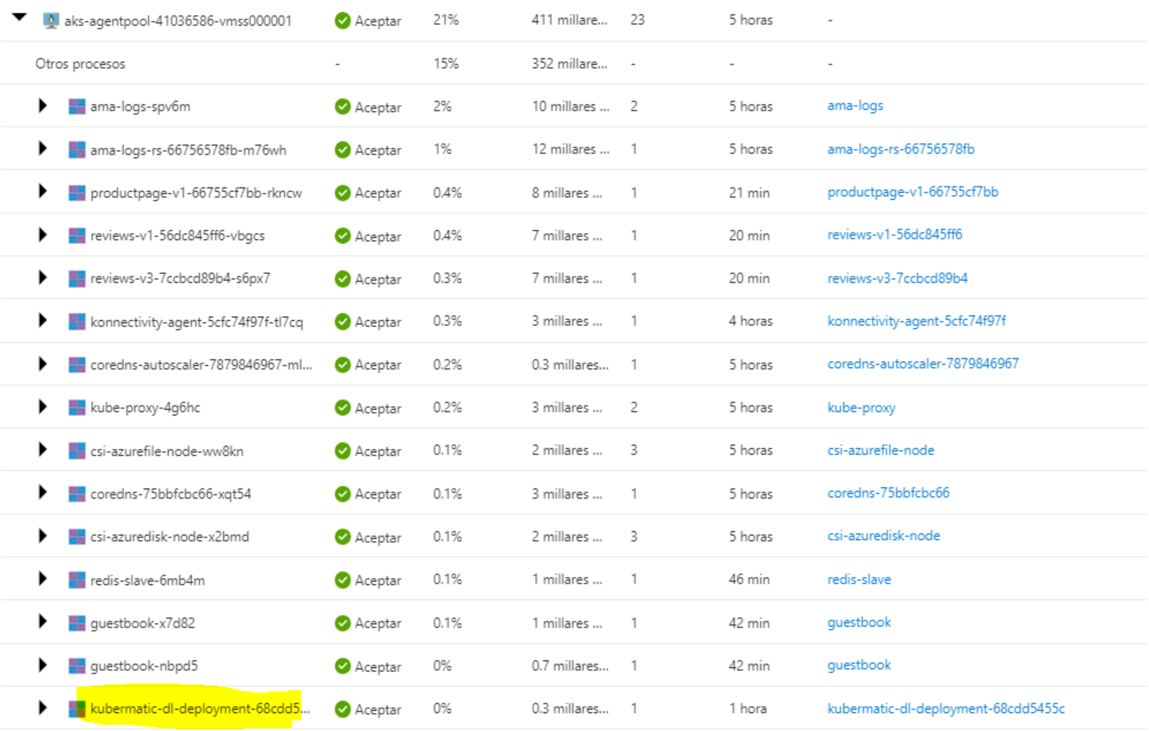
El servicio administrado de Azure Monitor para Prometheus ahora está en versión preliminar pública. Actualice la configuración del monitor para empezar a usar Prometheus administrado. Más información

Intervalo de tiempo = **Últimos 30 minutos**

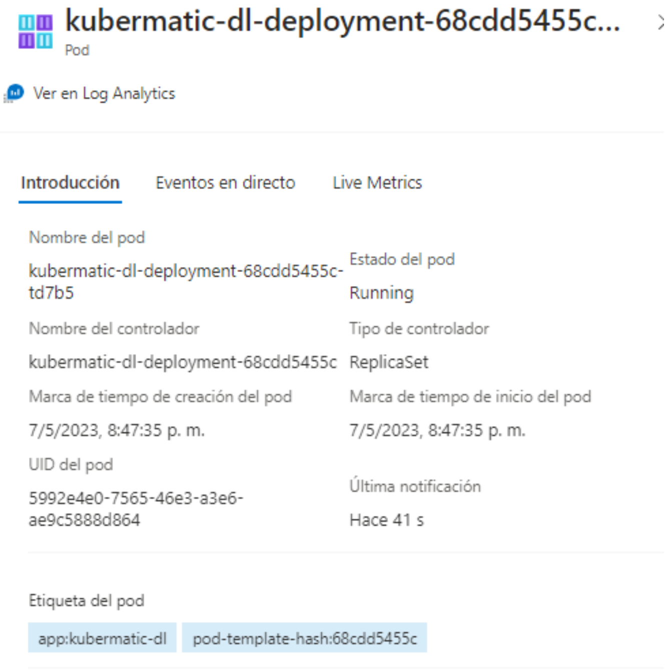
Métrica: Uso de la CPU (millicores) (calculado a partir de la capacidad)

2 elementos

Nombre	Estado	95 % ↓	95	Contenedore	Tiempo de a	Controlador	Tendencia 95 % (1 barra = 1m)
aks-agentpool-41036586-vmss000001	Aceptar	21%	411 millare...	23	5 horas	-	
aks-agentpool-41036586-vmss000000	Aceptar	8%	157 millare...	25	5 horas	-	



aks-agentpool-41036586-vmss000001	✓ Aceptar	21%	411 millare...	23	5 horas	-
Otros procesos	-	15%	352 millare...	-	-	-
ama-logs-spv6m	✓ Aceptar	2%	10 millares ...	2	5 horas	ama-logs
ama-logs-rs-66756578fb-m76wh	✓ Aceptar	1%	12 millares ...	1	5 horas	ama-logs-rs-66756578fb
productpage-v1-66755cf7bb-rkncw	✓ Aceptar	0.4%	8 millares ...	1	21 min	productpage-v1-66755cf7bb
reviews-v1-56dc845ff6-vbgcs	✓ Aceptar	0.4%	7 millares ...	1	20 min	reviews-v1-56dc845ff6
reviews-v3-7ccbcd89b4-s6px7	✓ Aceptar	0.3%	7 millares ...	1	20 min	reviews-v3-7ccbcd89b4
konnectivity-agent-5cfc74f97f-tl7cq	✓ Aceptar	0.3%	3 millares ...	1	4 horas	konnectivity-agent-5cfc74f97f
coredns-autoscaler-7879846967-ml...	✓ Aceptar	0.2%	0.3 millares...	1	5 horas	coredns-autoscaler-7879846967
kube-proxy-4g6hc	✓ Aceptar	0.2%	3 millares ...	2	5 horas	kube-proxy
csi-azurefile-node-ww8kn	✓ Aceptar	0.1%	2 millares ...	3	5 horas	csi-azurefile-node
coredns-75bbfcbc66-xqt54	✓ Aceptar	0.1%	3 millares ...	1	5 horas	coredns-75bbfcbc66
csi-azuredisk-node-x2bmd	✓ Aceptar	0.1%	2 millares ...	3	5 horas	csi-azuredisk-node
redis-slave-6mb4m	✓ Aceptar	0.1%	1 millares ...	1	46 min	redis-slave
guestbook-x7d82	✓ Aceptar	0.1%	1 millares ...	1	42 min	guestbook
guestbook-nbpd5	✓ Aceptar	0%	0.7 millares...	1	42 min	guestbook
kubermatic-dl-deployment-68cdd5...	✓ Aceptar	0%	0.3 millares...	1	1 hora	kubermatic-dl-deployment-68cdd5455c



kubermatic-dl-deployment-68cdd5455c...

Pod

Nombre del pod	Estado del pod
kubermatic-dl-deployment-68cdd5455c-td7b5	Running
Nombre del controlador	Tipo de controlador
kubermatic-dl-deployment-68cdd5455c	ReplicaSet
Marca de tiempo de creación del pod	Marca de tiempo de inicio del pod
7/5/2023, 8:47:35 p. m.	7/5/2023, 8:47:35 p. m.
UID del pod	Última notificación
5992e4e0-7565-46e3-a3e6-ae9c5888d864	Hace 41 s

Etiqueta del pod

app:kubermatic-dl pod-template-hash:68cdd5455c