Desafio #10

Objetivo:

El objetivo de este desafío es guiar la instalación y configuración de ArgoCD en un entorno de Kubernetes. ArgoCD es una herramienta de GitOps utilizada para gestionar despliegues en Kubernetes de manera declarativa. A través de este ejercicio, se pondrán en práctica los pasos necesarios para implementar una infraestructura automatizada de despliegue continuo.

Escenario:

Partiendo de un entorno local de Kubernetes utilizando algunas de las soluciones vistas en clase como plataforma de desarrollo (Killercoda, Kc0crl, Minikube, K3s, etc). El equipo de trabajo ha solicitado la implementación de ArgoCD para gestionar el flujo de despliegues de aplicaciones de manera automatizada. Se utilizarán manifiestos de Kubernetes y se integrará la plataforma con un repositorio Git para la gestión de aplicaciones.

Requisitos:

1. Instalar Kubectl
2. Instalar ArgoCD
3. Conectar un repositorio de Github

Para este desafio utilizo el mismo entorno que utilice para los anteriores, donde Kubectl y Minikube ya están instalados, dejo documentación en caso de querer instalar;

https://kubernetes.io/docs/tasks/tools/

Una vez instalado minikube, iniciamos;

$ minikube start

```
Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> cd ../..
PS C:\> cd .\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10\
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> minikube start
* minikube v1.35.0 en Microsoft Windows 10 Pro 10.0.19045.5371 Build 19045.5371
* Using the hyperv driver based on existing profile
* Starting "minikube" primary control-plane node in "minikube" cluster
* Restarting existing hyperv VM for "minikube" ...
! Failing to connect to https://registry.k8s.io/ from inside the minikube VM
* To pull new external images, you may need to configure a proxy: https://minikube.sigs.k8s.io/docs/referen
/proxy/
* Preparando Kubernetes v1.32.0 en Docker 27.4.0...
* Configurando CNI bridge CNI ...
* Verifying Kubernetes components...
  - Using image gcr.io/k8s-minikube/storage-provisioner:v5
* Complementos habilitados: storage-provisioner, default-storageclass
* Done! kubectl is now configured to use "minikube" cluster and "default" namespace by default
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10>
```

creamos un namespace para Argocd

```
S C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> kubectl create ns argocd
amespace/argocd created
S C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10>
```

Utilizamos el siguiente comando para instalar Argocd

$ kubectl apply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-cd/stable/manifests/install.yaml

```
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> kubectl ap
ply -n argocd -f https://raw.githubusercontent.com/argoproj/argo-cd/stable/manifests/install
.yaml
customresourcedefinition.apiextensions.k8s.io/applications.argoproj.io created
customresourcedefinition.apiextensions.k8s.io/applicationsets.argoproj.io created
customresourcedefinition.apiextensions.k8s.io/appprojects.argoproj.io created
serviceaccount/argocd-application-controller created
serviceaccount/argocd-applicationset-controller created
serviceaccount/argocd-dex-server created
serviceaccount/argocd-notifications-controller created
serviceaccount/argocd-redis created
serviceaccount/argocd-repo-server created
serviceaccount/argocd-server created
role.rbac.authorization.k8s.io/argocd-application-controller created
role.rbac.authorization.k8s.io/argocd-applicationset-controller created
role.rbac.authorization.k8s.io/argocd-dex-server created
role.rbac.authorization.k8s.io/argocd-notifications-controller created
role.rbac.authorization.k8s.io/argocd-redis created
role.rbac.authorization.k8s.io/argocd-server created
clusterrole.rbac.authorization.k8s.io/argocd-application-controller created
clusterrole.rbac.authorization.k8s.io/argocd-applicationset-controller created
clusterrole.rbac.authorization.k8s.io/argocd-server created
rolebinding.rbac.authorization.k8s.io/argocd-application-controller created
rolebinding.rbac.authorization.k8s.io/argocd-applicationset-controller created
rolebinding.rbac.authorization.k8s.io/argocd-dex-server created
rolebinding.rbac.authorization.k8s.io/argocd-notifications-controller created
rolebinding.rbac.authorization.k8s.io/argocd-redis created
rolebinding.rbac.authorization.k8s.io/argocd-server created
clusterrolebinding.rbac.authorization.k8s.io/argocd-application-controller created
clusterrolebinding.rbac.authorization.k8s.io/argocd-applicationset-controller created
```

Ejecutamos;

$ kubectl get all -n argocd

Para comprobar que esté corriendo;

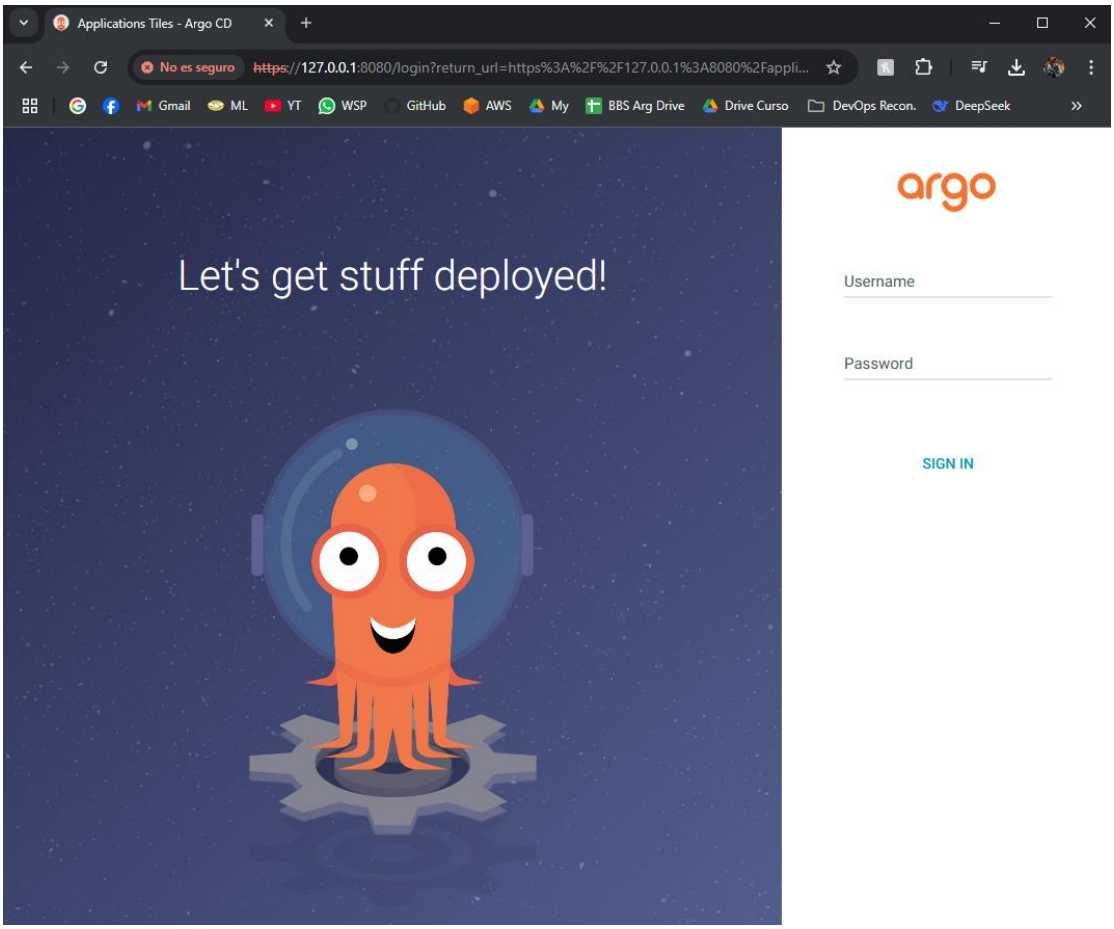
```
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> kubectl get all -n argocd
NAME                                                READY   STATUS    RESTARTS   AGE
pod/argocd-application-controller-0                 1/1     Running   0          98s
pod/argocd-applicationset-controller-597675595-kbpbz   1/1     Running   0          99s
pod/argocd-dex-server-5674dc45f9-bdwqb              1/1     Running   0          98s
pod/argocd-notifications-controller-7696b54558-glxbd   1/1     Running   0          98s
pod/argocd-redis-cd975cdbd-n2spx                    1/1     Running   0          98s
pod/argocd-repo-server-674756d699-4rh58             1/1     Running   0          98s
pod/argocd-server-65f5d8cc59-tr4bq                  1/1     Running   0          98s

NAME                                               TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)                      AGE
service/argocd-applicationset-controller           ClusterIP   10.111.80.215    <none>        7000/TCP,8080/TCP            99s
service/argocd-dex-server                          ClusterIP   10.103.187.207   <none>        5556/TCP,5557/TCP,5558/TCP   99s
service/argocd-metrics                             ClusterIP   10.110.83.197    <none>        8082/TCP                     99s
service/argocd-notifications-controller-metrics    ClusterIP   10.99.50.252     <none>        9001/TCP                     99s
service/argocd-redis                               ClusterIP   10.104.78.214    <none>        6379/TCP                     99s
service/argocd-repo-server                         ClusterIP   10.97.10.125     <none>        8081/TCP,8084/TCP            99s
service/argocd-server                              ClusterIP   10.111.163.165   <none>        80/TCP,443/TCP               99s
service/argocd-server-metrics                      ClusterIP   10.106.210.37    <none>        8083/TCP                     99s

NAME                                            READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/argocd-applicationset-controller   1/1     1            1           99s
deployment.apps/argocd-dex-server               1/1     1            1           99s
deployment.apps/argocd-notifications-controller    1/1     1            1           98s
deployment.apps/argocd-redis                    1/1     1            1           98s
deployment.apps/argocd-repo-server              1/1     1            1           98s
deployment.apps/argocd-server                   1/1     1            1           98s

NAME                                                      DESIRED   CURRENT   READY   AGE
replicaset.apps/argocd-applicationset-controller-597675595   1         1         1       99s
replicaset.apps/argocd-dex-server-5674dc45f9              1         1         1       99s
replicaset.apps/argocd-notifications-controller-7696b54558   1         1         1       98s
replicaset.apps/argocd-redis-cd975cdbd                    1         1         1       98s
replicaset.apps/argocd-repo-server-674756d699             1         1         1       98s
replicaset.apps/argocd-server-65f5d8cc59                  1         1         1       98s

NAME                                               READY   AGE
statefulset.apps/argocd-application-controller   1/1     98s
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10>
```

Realizamos un port-forward;

$ kubectl port-forward svc/argocd-server -n argocd 8080:443

```
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> kubectl port-forward svc/argocd-server -n argocd 8080:443
Forwarding from 127.0.0.1:8080 -> 8080
Forwarding from [::1]:8080 -> 8080
Handling connection for 8080
```

Testeamos que la aplicación esté levantada en localhost:8080



Ejecuto comando;

$ kubectl get secret argocd-initial-admin-secret -n argocd -o jsonpath="{.data.password}"

| base64 --decode

El mismo me da error con la librería base64, busque alguna alternativa y encontre la siguiente;

$ $pass = kubectl get secret argocd-initial-admin-secret -n argocd -o jsonpath="{.data.password}"

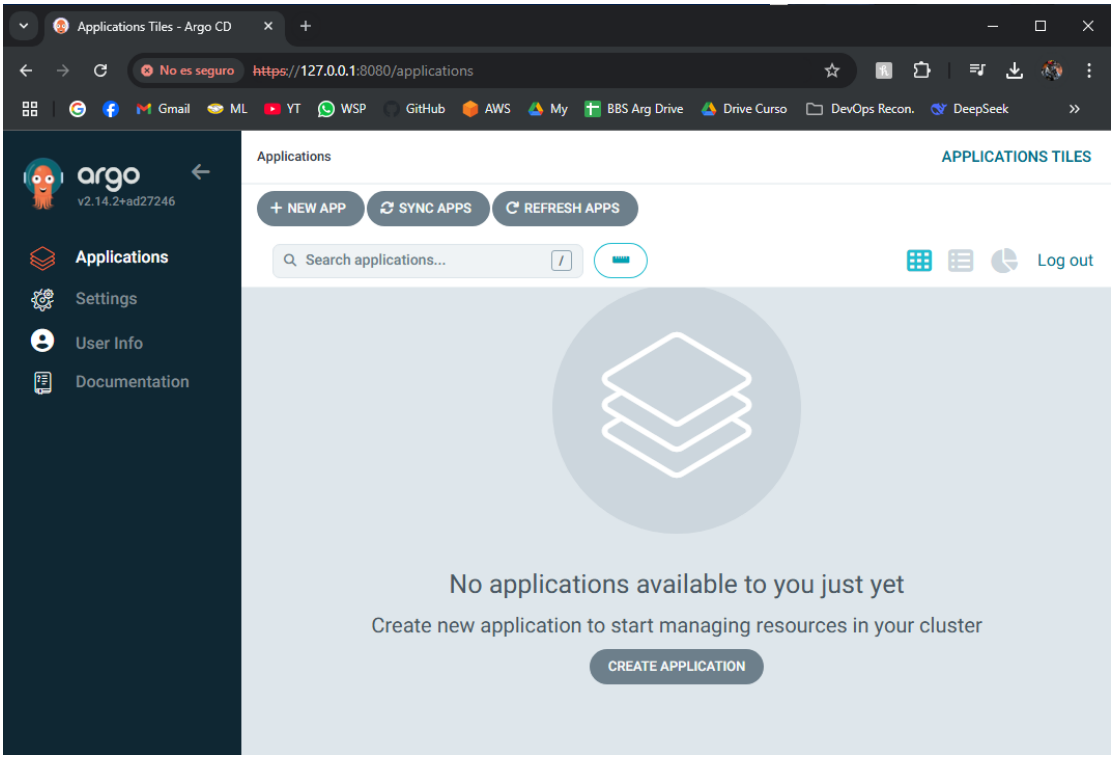
[System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($pass))

```
PS C:\Windows\system32> $pass = kubectl get secret argocd-initial-admin-secret -n argocd -o jsonpath="{.data.password}"
>> [System.Text.Encoding]::UTF8.GetString([System.Convert]::FromBase64String($pass))
IQBc4O-6c2N-Gj3C
PS C:\Windows\system32>
```

Utilizamos las credenciales;

$ user; admin
$ password; IQBc4O-6c2N-Gj3C (Brindadas por el paso anterior)

Ingresamos;



En este caso opté por realizar la aplicación por manifiesto, otra opción es por interfaz,

Creo el archivo .yaml para el manifiesto

$ nano 01-guestbook-app.yaml

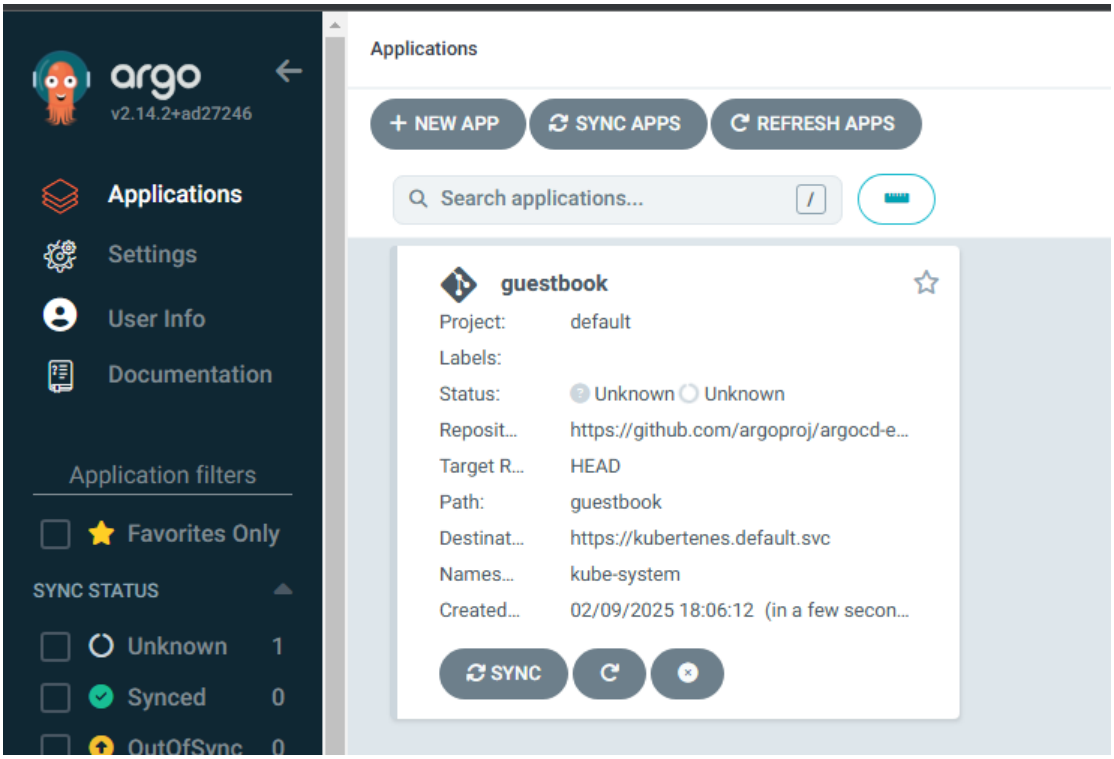
Seteamos la siguiente configuración;

```
  GNU nano for Windows 64 bits, v7.2-22.1 2023.04.15
apiVersion: argoproj.io/v1alpha1
kind: Application
metadata:
  name: guestbook
  namespace: argocd
spec:
  source:
    path: guestbook
    repoURL: https://github.com/argoproj/argocd-example-apps.git
    targetRevision: HEAD
  destination:
    server: 'https://kubertenes.default.svc'
    namespace: kube-system
  project: default
```

Desplegamos la aplicación;

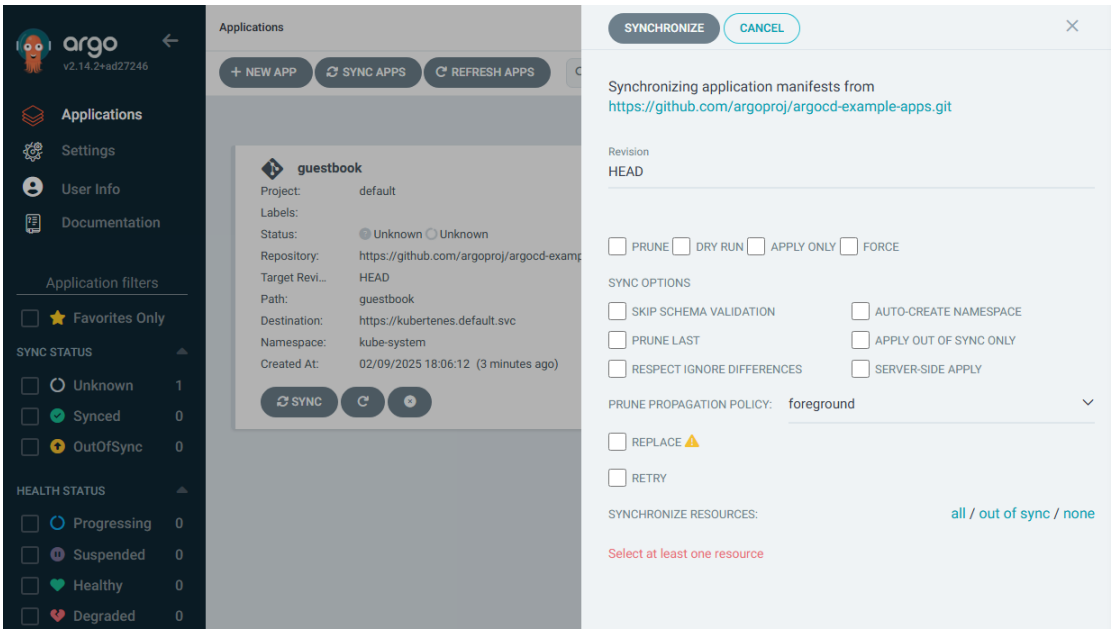
```
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> nano 01-guestbook-app.yaml
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> kubectl apply -f .\01-guestbook-app.yaml
application.argoproj.io/guestbook created
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10>
```

Verificamos en ArgoCD que se pueda visualizar;

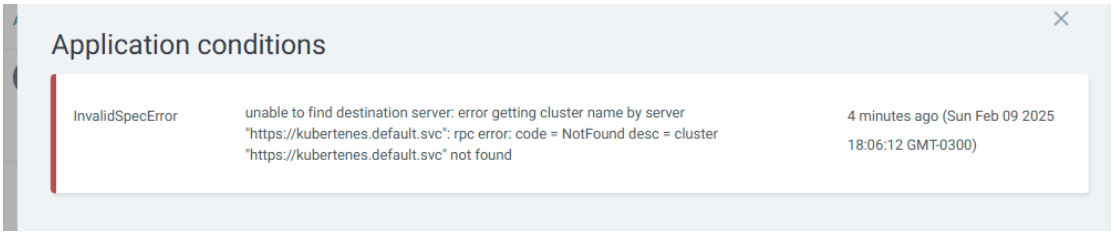


Le damos a Sync para configurar la Sincronización;

*Error; Syncrhonize Resources no detecta elementos para sincronizar.



Investigamos la app;



*Error en manifiesto; el contenido de 'server:' no es el correcto, ubicamos el correcto de mi equipo con el comando;

$ cat ~/.kube/config

Se encuentra en;

clusters:
- cluster:
    certificate-authority-data: ***
    server: https://kubernetes.docker.internal:6443

```
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> cat ~/.kube/config
apiVersion: v1
clusters:
- cluster:
    certificate-authority-data: LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURCVENDQWUyZ0F3SUJBZ01JVFlXMmx6YWQ3bWt3RFFZSktvWklodmNOQVFF
dGVEVUTUJFR0ExVUUKQXhNS2EzVmlaWEp1WlhSbGN6QWVGdzB5T1RBeU1EY3hPRFV3T1RaYUZ3MHpOVEF5TURVeE9EVXdOVFphTUJVeApFekFSQmdOVkJBTVRDbXQxbW1WeW
WE13Z2dFaU1BMEdDU3FHU01iM0RRRUJBUVVBQTRJQkR3QXdnZ0VLCkFvSUJBUUN2Mm1XYWRiRjQrMlQwQnJUVTYvUDVSVFVEZ2Y3V1lpVnlxMFd2Q2R1YTMzR1RwS1UzVGlH
4KZVJBOFRmN3ZMT0VsaXJzRXBIZnRQWGZjZ31rQmFVbHYxWHRZYnZLVEdyQ3NUZX1pSTRVMk5JekZFOXVQdkFDagp4dzRGVHYrSjBRWVNpU3RDQW9NaTQzdy9ZLzlRK3Y5cz
VENMa3Y5VFBVeHdDdG5uRTZQa1NvbjdDamlFCm1mcF1BbmZORnVTT2ZINXRHTEM2MFd6ZTZjR2NpQk1UaHNxUFdaWkVUcU1PM0hSbkE5ODRxUDQ0ZUYycXNSeG8KMGxkSExC
ZILz1wdzJJckJ5VWV5aU1HOWt2dUQxNXNRZ2ttRC9zZj1ydF1xbVBPZERacGl2d1hEcjhZawpUTWp0bHVGSGZ6MXBiRTJOdGFPTFVPcmsxenp6QWdNQkFBR2pXVEJYTUE0R0
d0VCL3dRRUF3SUNwREFQCkJnTlZIUk1CQWY4RUJUQURBUgvTUIwR0ExVWREZ1FXQkJUZV1IdlREWUJDcHdOOXNBMllTU1kWm1WcVZ6QVYKQmdOVkhSRUVEakFNZ2dwcmRX
VsZEdWek1BMEdDU3FHU01iM0RRRUJDd1VBQTRJQkFRQURkRHZvSGtvZApzWGU3d2J2SDhqZFEvK3p4eHoxRUhSV1pQL0JQZ2NhR2JDN0dlZ1J1UXk2VEFFZ1hRNmx6dEdHU3
Q3hLC1ZxVXUyYX1zbStWcDgxOD1Ha0tDUmpwMjlDYXBHaUw4UV1QMWozN2JPWWptVG1oaXpqK1NXM0p5M1hxU2QxSWkKcEZJbn1RODZwYVZ2SGJuVTAvQ1dTd1FwTkJ3Ti9x
V2L3NtNFB1QmtHY1ZyV1ZXQ3FxU2NMUGNPYWR6VApTRkFuNnoyMWU4c1czdFBQRVp0WnFmWUJCTWNRaFJpSHdwWjZpcVNEb1BRUzRjS202RjRCZn1EUm80aHVuSHZ3CmpiUT
QlpWRWR4amIvRjVvaFNkaHZ5R1BvalIrc2VCcDNnRjNzaGJheUVFbXkyRnIrMnppQUVIMzRPMDkKWTdrZ1F0T2FJZ0tpCi0tLS0tRU5EIENFUlRJRklDQVRFLS0tLS0K
    server: https://kubernetes.docker.internal:6443
  name: docker-desktop
- cluster:
    certificate-authority: C:\Users\brian\.minikube\ca.crt
    extensions:
    - extension:
        last-update: Sun, 09 Feb 2025 17:39:44 -03
        provider: minikube.sigs.k8s.io
        version: v1.35.0
      name: cluster_info
    server: https://172.21.39.43:8443
  name: minikube
contexts:
- context:
    cluster: docker-desktop
    user: docker-desktop
  name: docker-desktop
- context:
    cluster: minikube
    extensions:
    - extension:
        last-update: Sun, 09 Feb 2025 17:39:44 -03
        provider: minikube.sigs.k8s.io
        version: v1.35.0
      name: context_info
    namespace: default
    user: minikube
  name: minikube
current-context: minikube
kind: Config
preferences: {}
```

Modificamos;

```
  GNU nano for Windows 64 bits, v7.2-22.1 2023.04.15
apiVersion: argoproj.io/v1alpha1
kind: Application
metadata:
  name: guestbook
  namespace: argocd
spec:
  source:
    path: guestbook
    repoURL: https://github.com/argoproj/argocd-example-apps.git
    targetRevision: HEAD
  destination:
    server: 'https://kubernetes.docker.internal:6443'
    namespace: kube-system
  project: default
```

Aplicamos nuevamente;

```
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> kubectl apply -f .\01-guestbook-app.yaml
application.argoproj.io/guestbook configured
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10>
```

*Error; Continuamos teniendo error, verificamos cuál puede ser la causa, ejecutamos comando;

$ argocd cluster list

**Atención con este comando, se debe iniciar sesión por argocd-cli*

```
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> argocd cluster list
SERVER                          NAME        VERSION  STATUS   MESSAGE                                                  PROJECT
https://kubernetes.default.svc  in-cluster           Unknown  Cluster has no applications and is not being monitored.
```

Verificamos los nodos de Kubectl e información de los Cluster

$ kubectl get nodes
$ kubectl cluster-info

```
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> kubectl get nodes
NAME       STATUS   ROLES    AGE   VERSION
minikube   Ready    <none>   44m   v1.32.0
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> kubectl cluster-info
Kubernetes control plane is running at https://172.21.39.43:8443
CoreDNS is running at https://172.21.39.43:8443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
```

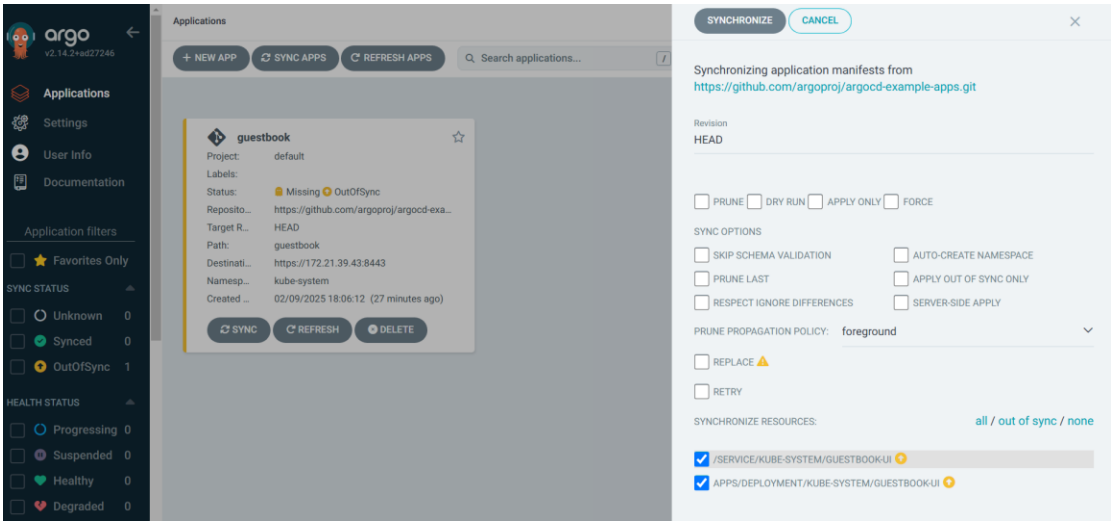
*Error; kubernetes está siendo ejecutado desde minikube, la cuál corre desde otra IP, y no está agregada a la cluster list de Argo CD

Agregamos;

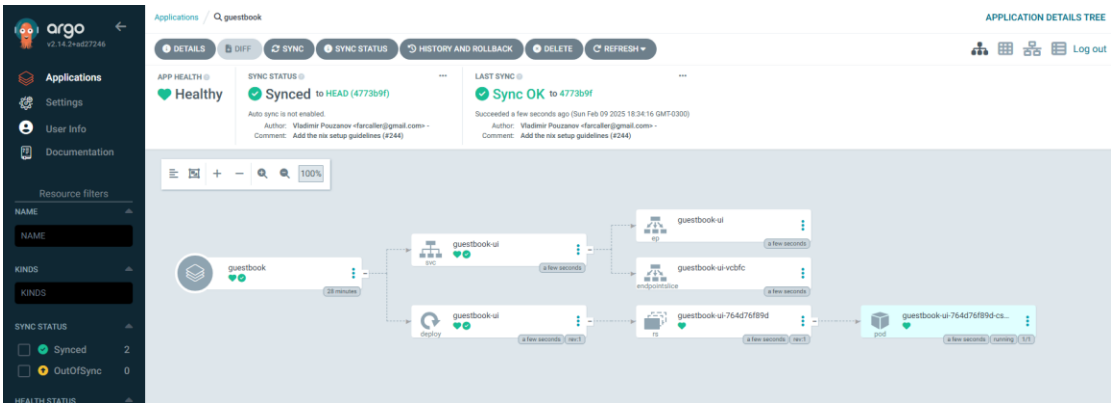
$ argocd cluster add minikube

```
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> argocd cluster add minikube
WARNING: This will create a service account `argocd-manager` on the cluster referenced by context `minikube` with full cluster level privileges
Do you want to continue [y/N]? y
time="2025-02-09T18:28:03-03:00" level=info msg="ServiceAccount \"argocd-manager\" created in namespace \"kube-system\""
time="2025-02-09T18:28:03-03:00" level=info msg="ClusterRole \"argocd-manager-role\" created"
time="2025-02-09T18:28:03-03:00" level=info msg="ClusterRoleBinding \"argocd-manager-role-binding\" created"
time="2025-02-09T18:28:03-03:00" level=info msg="Created bearer token secret for ServiceAccount \"argocd-manager\""
Cluster 'https://172.21.39.43:8443' added
PS C:\Users\brian\OneDrive\Escritorio\DevOps_Engineer\Github_bootcamp\Desafio_10> argocd cluster list
SERVER                          NAME        VERSION  STATUS      MESSAGE                                                  PROJECT
https://172.21.39.43:8443       minikube    1.32     Successful
https://kubernetes.default.svc  in-cluster           Unknown     Cluster has no applications and is not being monitored.
```

Verificamos en la aplicación;



Sincronizamos..



Aplicación Sincronizada