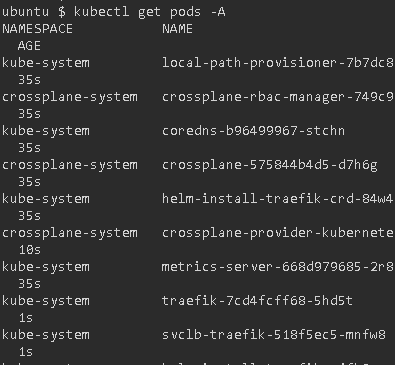
|  |  |
| --- | --- |
| **DOCUMENT RULES:** | |
| **Task Number / Name:** | **Crossplane** |
| **Task name & column name should be written:** | **Bold (CTRL+B)** |
| **Commands should be written in the after # sign:** | *Italic (CTRL+I) #hostname* |
| **Output photo should be cropped or compressed:**  **Photo could be more than one:**  **If you need extra lines, add the line next after it:** | ***Description photo should be with title bar (CTRL + I + B)*** |
| **All other text should be written:** | Standard |
| **Font name and text size:** | Calibri and 9 |
| **Group name:** | Dev\_ops\_ |
| **Student name and surname:** | Murad Abbaszade |
| **E-mail:** | [muradabbaszade6@gmail.com](mailto:muradabbaszade6@gmail.com) |
| **WhatsApp number:** | **+994703664205** |

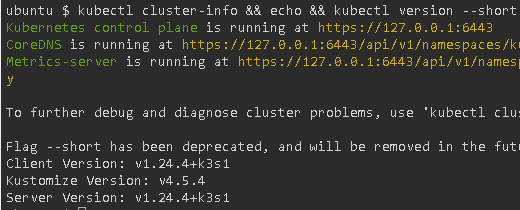
# Crossplane

# Interactive Crossplane Workshop

Check the cluster setup kubectl get pods –A



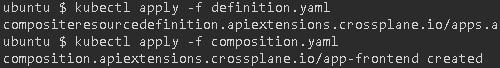
Check Kuberentes cluster info and version kubectl cluster-info && echo && kubectl version –short



## Composed App in Action

kubectl apply -f definition.yaml

kubectl apply -f composition.yaml



kubectl get xrd



kubectl get compositions



kubectl create ns devops-team

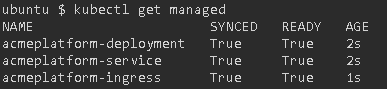


kubectl apply -f app-claim.yaml

kubectl wait deployment.apps/acmeplatform --namespace devops-team --for condition=AVAILABLE=True --timeout 1m



kubectl get managed



kubectl port-forward deployment/acmeplatform -n devops-team --address 0.0.0.0 8080:80



## Application Lifecycle

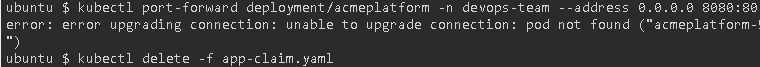
kubectl apply -f app-claim-blue.yaml

kubectl wait deployment.apps/acmeplatform --namespace devops-team --for condition=AVAILABLE=True --timeout 1m



kubectl port-forward deployment/acmeplatform -n devops-team --address 0.0.0.0 8080:80

kubectl delete -f app-claim.yaml

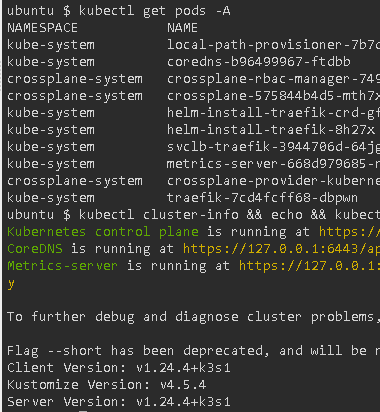


# Validate Crossplane compositions with Datree

## Setting up the Environment

kubectl get pods –A

kubectl cluster-info && echo && kubectl version –short



## Introduction to Datree

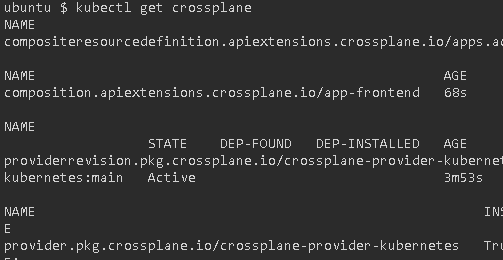
Datree prevents Kubernetes misconfigurations from reaching production.

## Prepare the resources

* kubectl apply -f definition.yaml
* kubectl apply -f composition.yaml
* kubectl create ns devops-team
* kubectl apply -f app-claim.yaml



kubectl get crossplane



kubectl describe service acmeplatform -n devops-team | grep ^Labels -A2

kubectl describe deployment acmeplatform -n devops-team | grep ^Labels -A2

kubectl describe hpa acmeplatform -n devops-team | grep ^Labels -A2

