

LAPORAN PRAKTIKUM VIRTUALISASI KOMPUTER

MENJALANKAN KUBERNETES DASHBOARD DAN FILE DEPLOYMENT YAML



Agus Pranata Marpaung

13323033

DIII TEKNOLOGI KOMPUTER

**INSTITUT TEKNOLOGI DEL
FAKULTAS VOKASI**

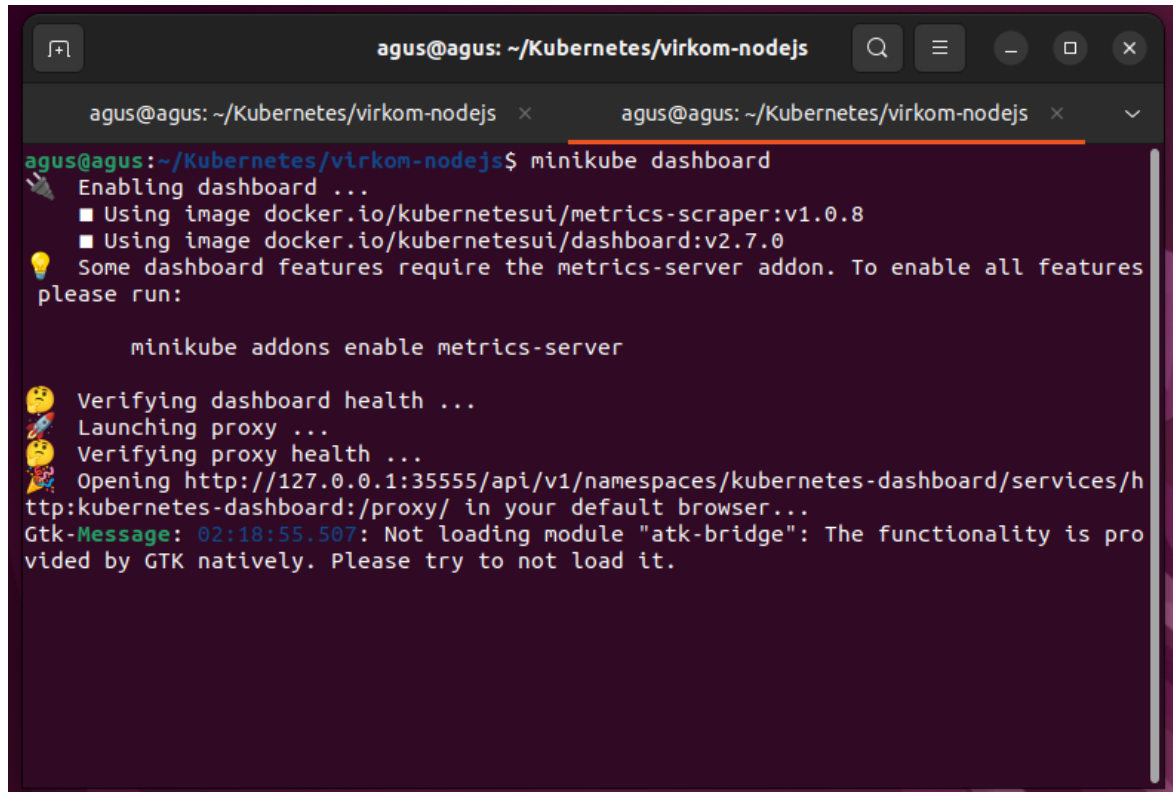
Judul Praktikum

Minggu/Sesi	:	XV/2
Kode Mata Kuliah	:	4332103
Nama Mata Kuliah	:	VIRTUALISASI KOMPUTER
Setoran	:	Jawaban dalam bentuk <i>softcopy</i>
Batas Waktu Setoran	:	9 Desember 2024 jam 21:30
Tujuan	:	1. Mahasiswa mampu membuat konfigurasi file yaml pada deployment dan service.

Petunjuk

Praktikum

1. Pada bagian ini, kita akan menampilkan dashboard di Web dari Minikube yang mana Minikube ini adalah bagian dari Kubernetes. Untuk menampilkan dashboard, kita dapat melakukannya dengan perintah **minikube dashboard**. Dan dia akan otomatis membuka web tersebut di Firefox.

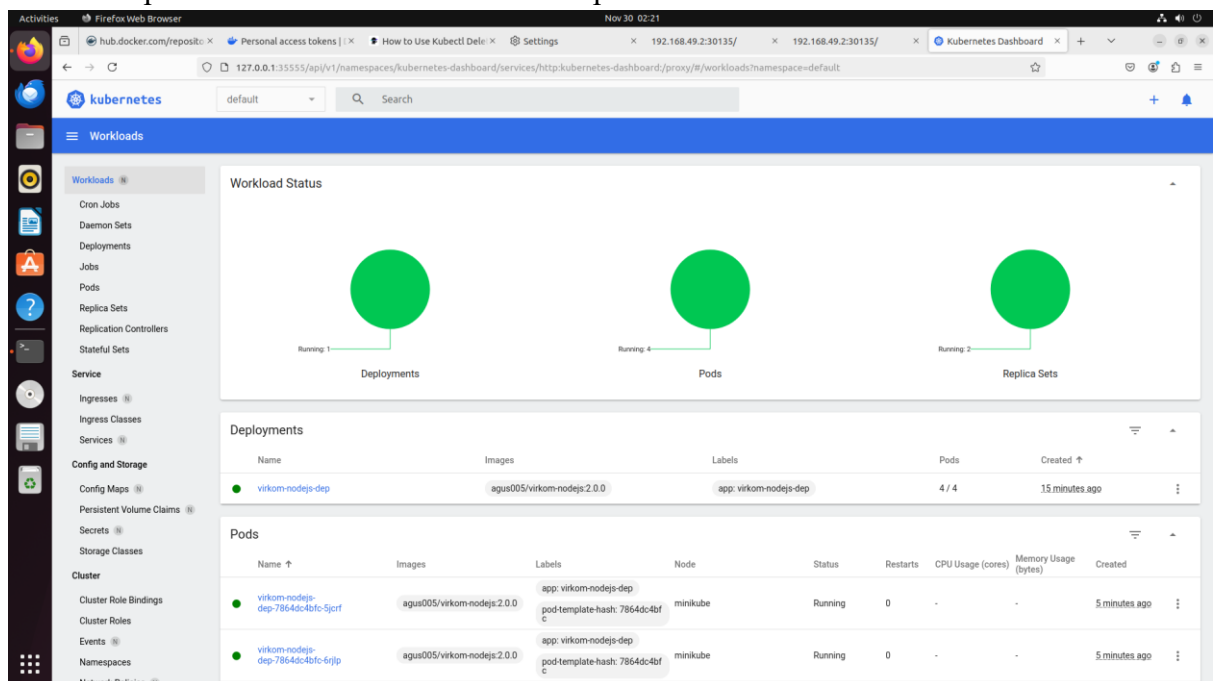


```
agus@agus: ~/Kubernetes/virkom-nodejs
agus@agus: ~/Kubernetes/virkom-nodejs$ minikube dashboard
Enabling dashboard ...
  ■ Using image docker.io/kubernetesui/metrics-scraper:v1.0.8
  ■ Using image docker.io/kubernetesui/dashboard:v2.7.0
Some dashboard features require the metrics-server addon. To enable all features
please run:

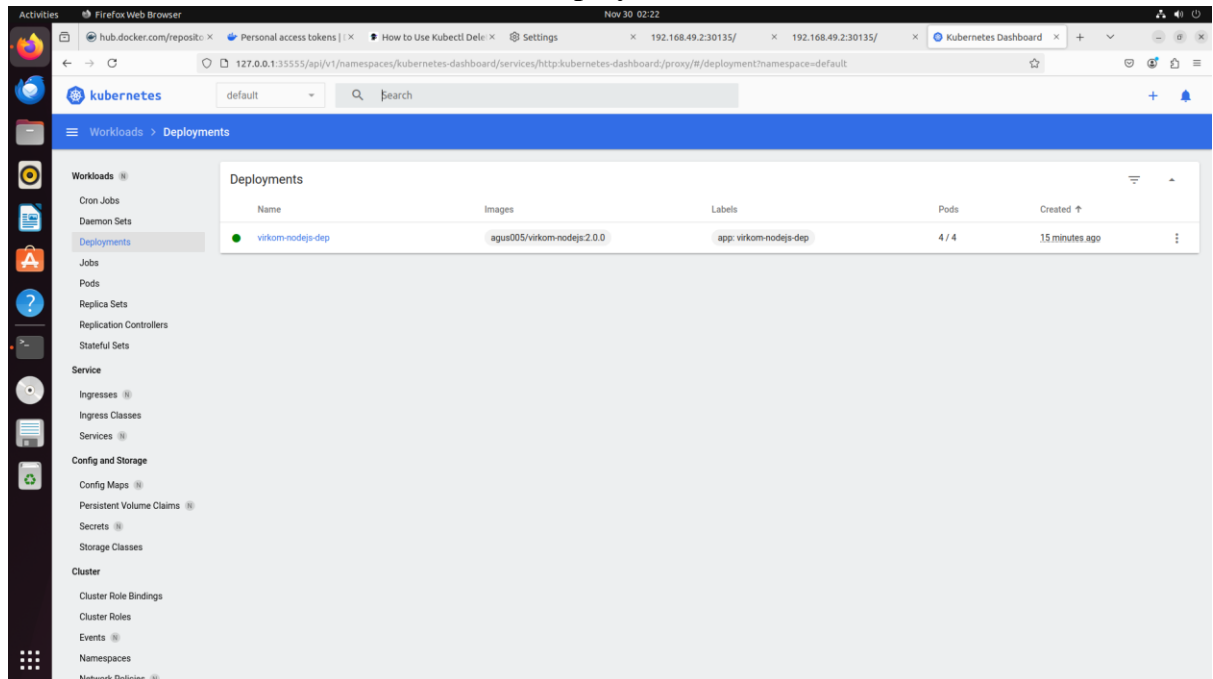
    minikube addons enable metrics-server

Verifying dashboard health ...
Launching proxy ...
Verifying proxy health ...
Opening http://127.0.0.1:35555/api/v1/namespaces/kubernetes-dashboard/services/
http:kubernetes-dashboard:/proxy/ in your default browser...
Gtk-Message: 02:18:55.507: Not loading module "atk-bridge": The functionality is pro
vided by GTK natively. Please try to not load it.
```

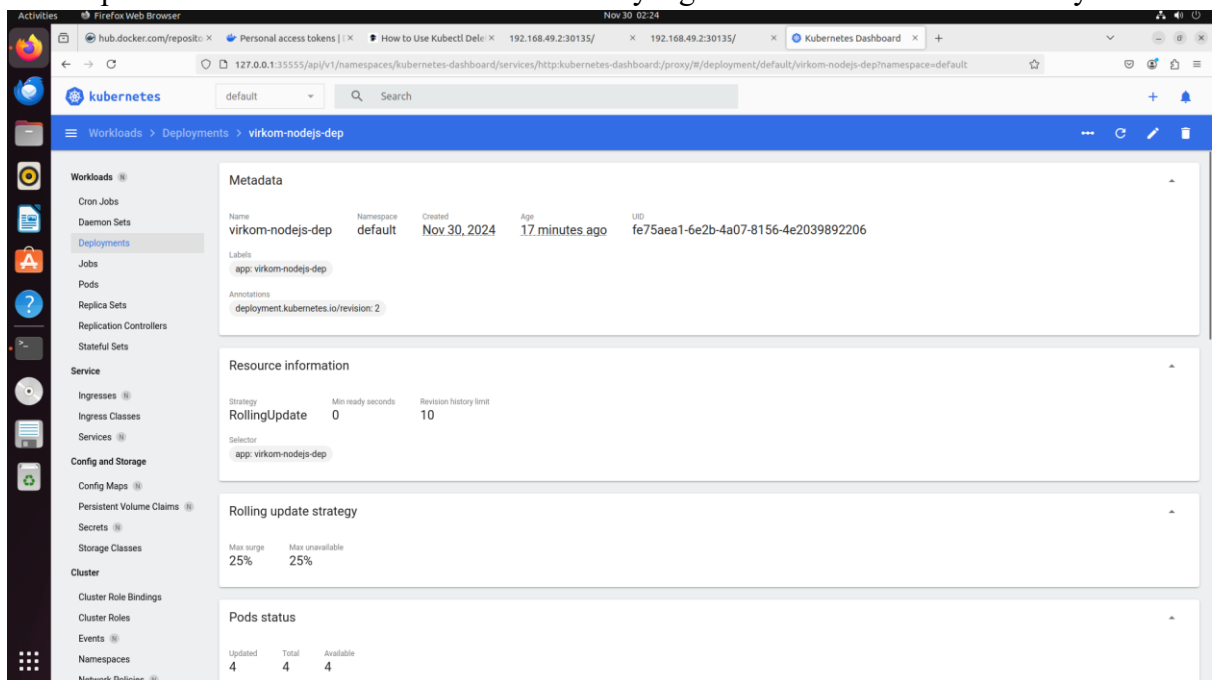
2. Untuk tampilan dashboard dari Kubernetes seperti ini.



3. Kemudian kita arahkan ke **Workloads > Deployments**.



4. Kita dapat melihat semua informasi dari kubectl yang sudah kita lakukan sebelumnya.

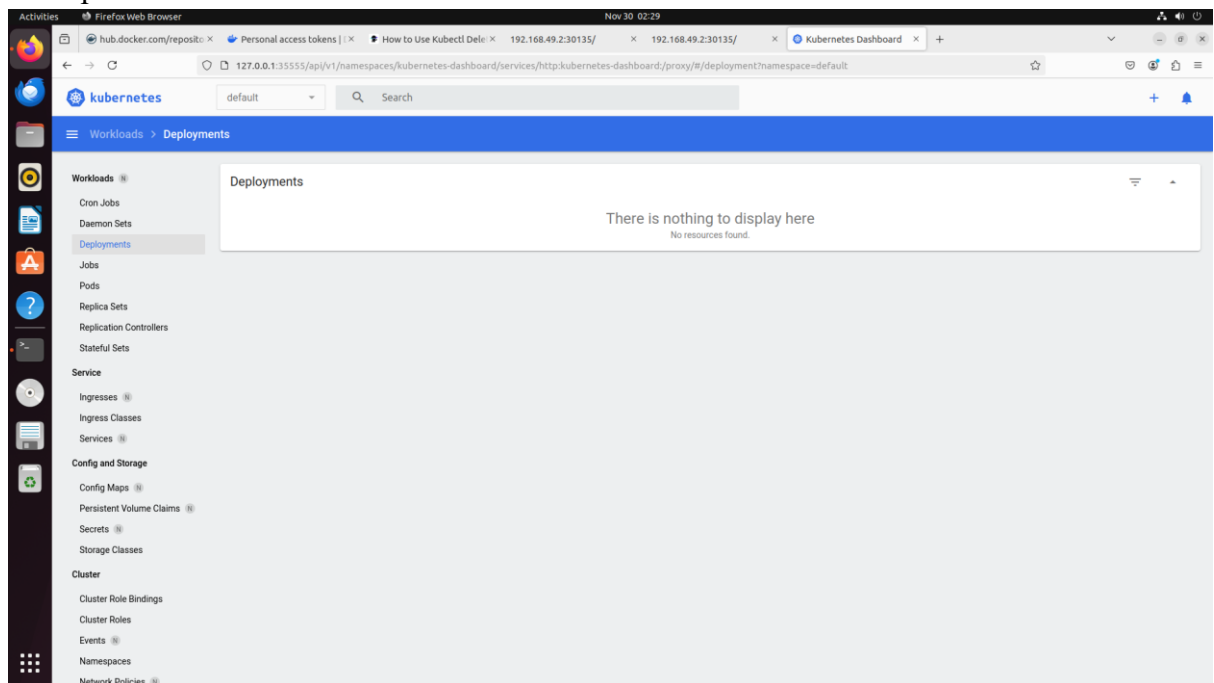


5. Setelah itu, kita dapat menghapus deployment dan juga service yang sudah kita buat sebelumnya dengan menjalankan perintah berikut.

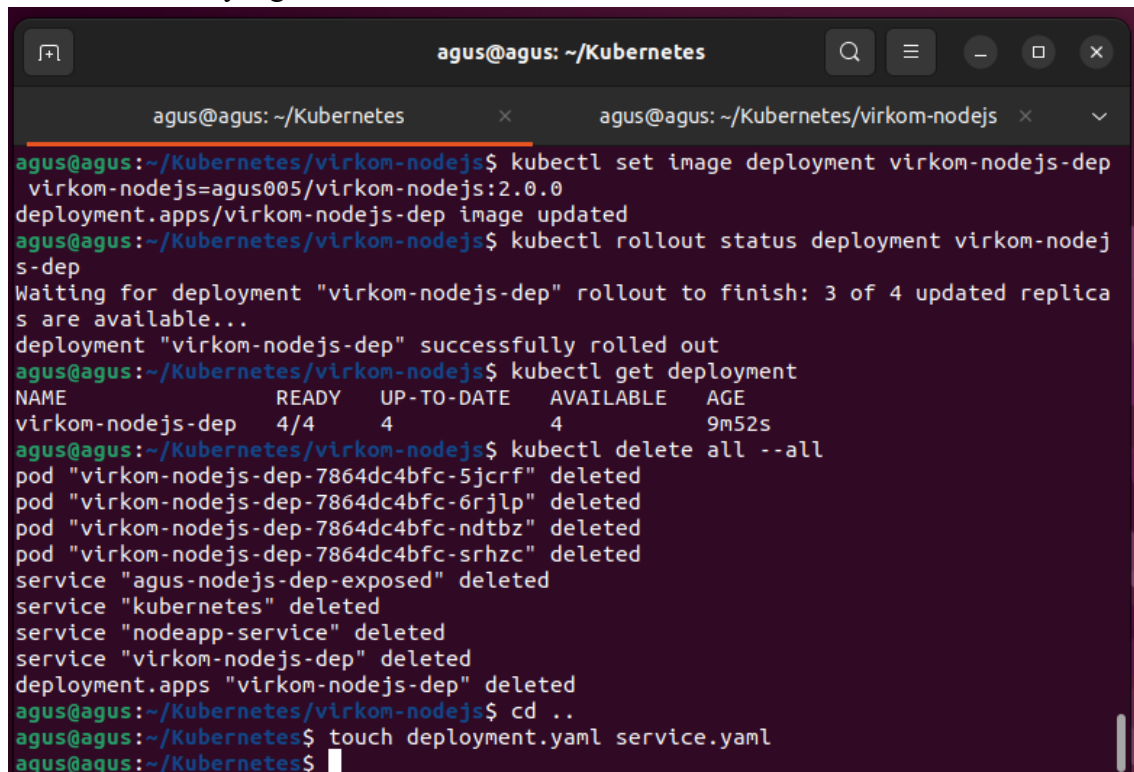
```
kubectl delete all --all
```

```
agus@agus:~/Kubernetes/virkom-nodejs$ kubectl delete all --all
pod "virkom-nodejs-dep-7864dc4bfc-5jcrf" deleted
pod "virkom-nodejs-dep-7864dc4bfc-6rjlp" deleted
pod "virkom-nodejs-dep-7864dc4bfc-ndtbz" deleted
pod "virkom-nodejs-dep-7864dc4bfc-srhzc" deleted
service "agus-nodejs-dep-exposed" deleted
service "kubernetes" deleted
service "nodeapp-service" deleted
service "virkom-nodejs-dep" deleted
deployment.apps "virkom-nodejs-dep" deleted
agus@agus:~/Kubernetes/virkom-nodejs$
```

6. Kemudian, kita cek di dashboard dan terlihat sudah tidak ada lagi Deployment yang ditampilkan.

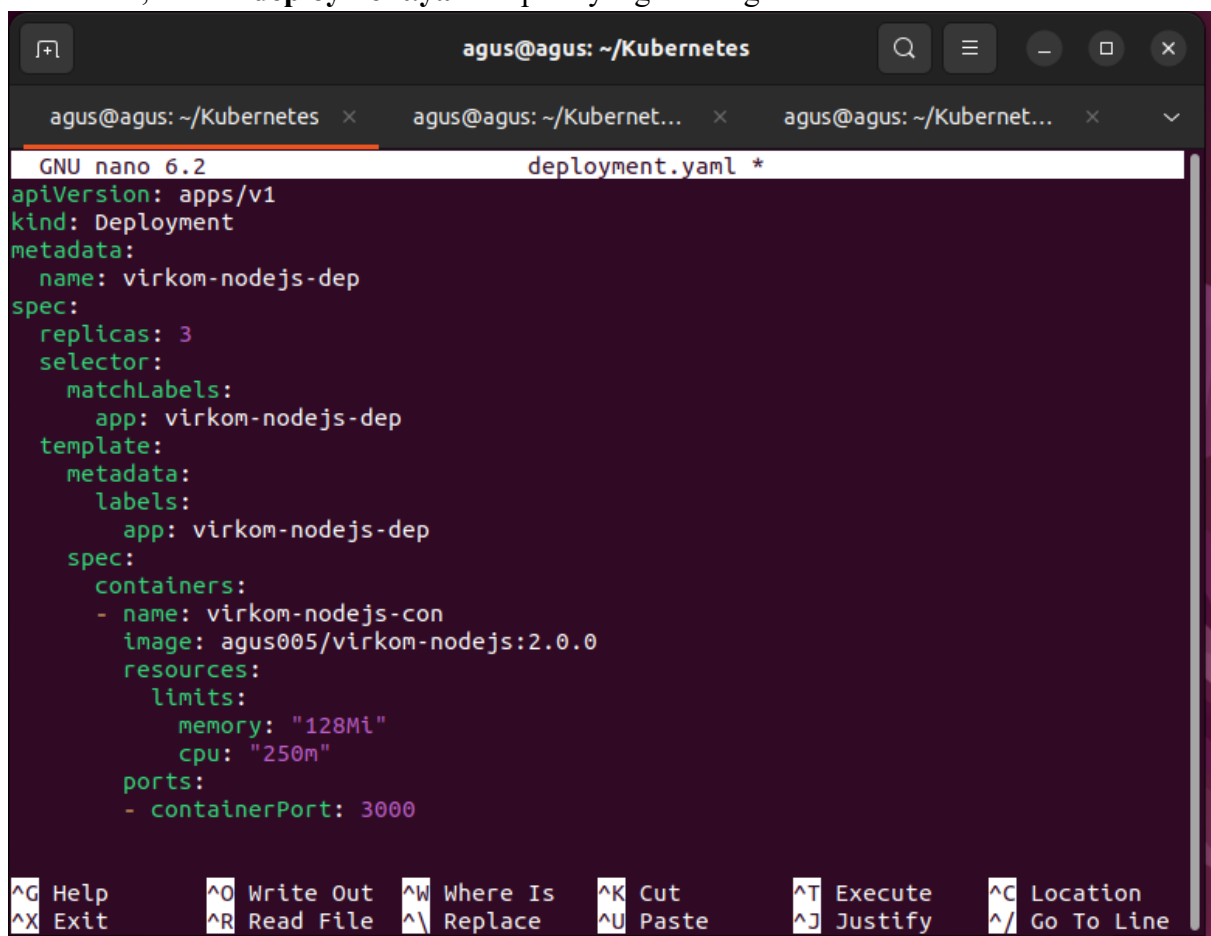


7. Lalu, kita akan membuat 2 file berekstensi **.yaml** dengan nama **deployment** dan juga **service** di folder yang kita buat.



```
agus@agus: ~/Kubernetes
agus@agus: ~/Kubernetes
agus@agus: ~/Kubernetes/virkom-nodejs$ kubectl set image deployment virkom-nodejs-dep virkom-nodejs=agus005/virkom-nodejs:2.0.0
deployment.apps/virkom-nodejs-dep image updated
agus@agus: ~/Kubernetes/virkom-nodejs$ kubectl rollout status deployment virkom-nodejs-dep
Waiting for deployment "virkom-nodejs-dep" rollout to finish: 3 of 4 updated replicas are available...
deployment "virkom-nodejs-dep" successfully rolled out
agus@agus: ~/Kubernetes/virkom-nodejs$ kubectl get deployment
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
virkom-nodejs-dep    4/4     4            4           9m52s
agus@agus: ~/Kubernetes/virkom-nodejs$ kubectl delete all --all
pod "virkom-nodejs-dep-7864dc4bfc-5jcrf" deleted
pod "virkom-nodejs-dep-7864dc4bfc-6rjlp" deleted
pod "virkom-nodejs-dep-7864dc4bfc-ndtbz" deleted
pod "virkom-nodejs-dep-7864dc4bfc-srhzc" deleted
service "agus-nodejs-dep-exposed" deleted
service "kubernetes" deleted
service "nodeapp-service" deleted
service "virkom-nodejs-dep" deleted
deployment.apps "virkom-nodejs-dep" deleted
agus@agus: ~/Kubernetes/virkom-nodejs$ cd ..
agus@agus: ~/Kubernetes$ touch deployment.yaml service.yaml
agus@agus: ~/Kubernetes$
```

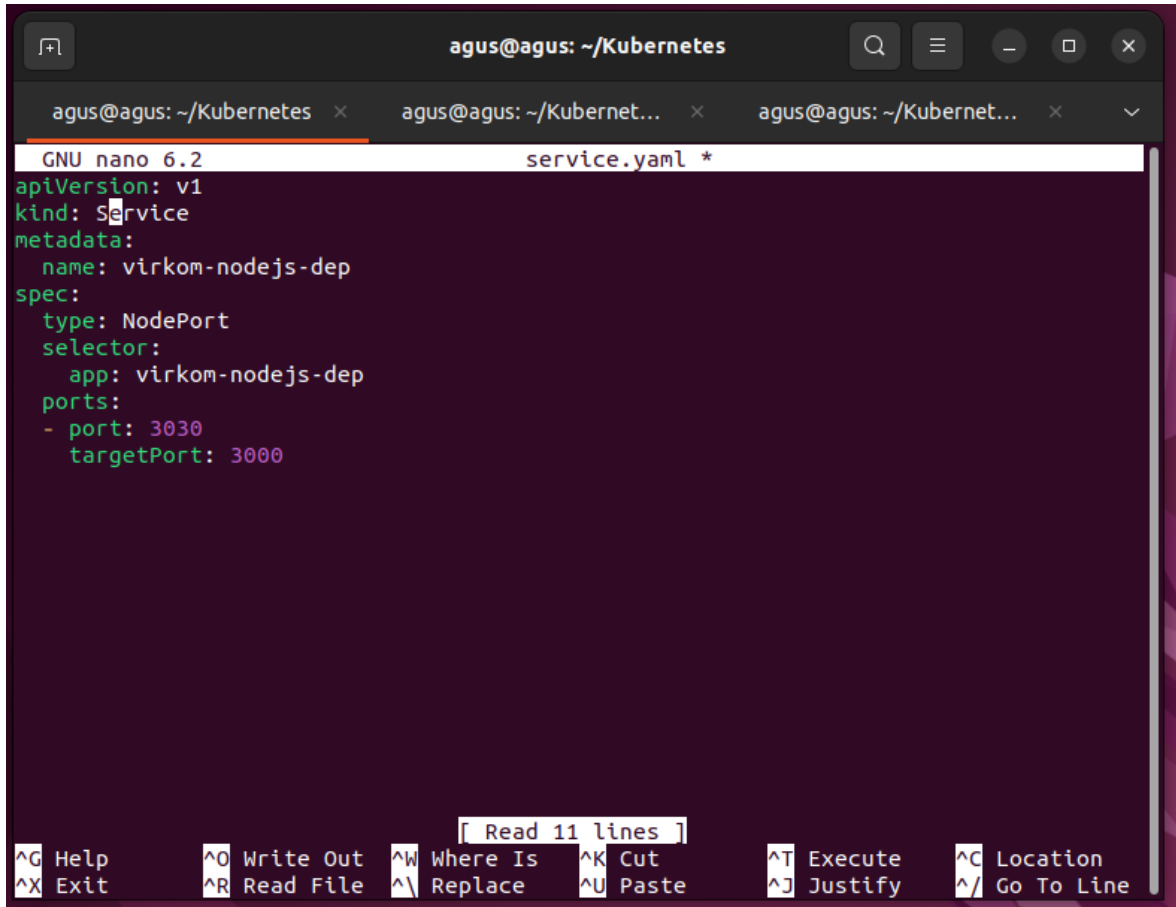
8. Kemudian, kita isi **deployment.yaml** seperti yang ada di gambar berikut.



```
GNU nano 6.2 deployment.yaml *
apiVersion: apps/v1
kind: Deployment
metadata:
  name: virkom-nodejs-dep
spec:
  replicas: 3
  selector:
    matchLabels:
      app: virkom-nodejs-dep
  template:
    metadata:
      labels:
        app: virkom-nodejs-dep
    spec:
      containers:
        - name: virkom-nodejs-con
          image: agus005/virkom-nodejs:2.0.0
          resources:
            limits:
              memory: "128Mi"
              cpu: "250m"
          ports:
            - containerPort: 3000
```

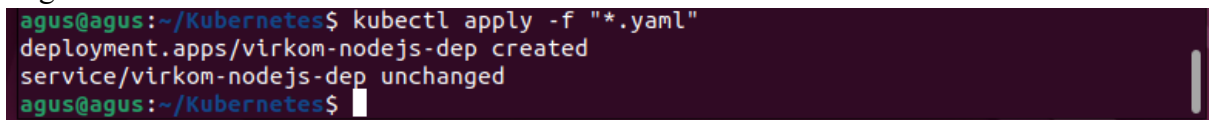
Pada file ini, kita membuat sebuah deployment dengan nama yang kita inginkan. Spesifikasi yang digunakan pada deployment itu membuat sebuah replica dengan jumlah 3 pods dan dilabeli sesuai dengan nama deployment. Kemudian untuk setiap pod yang kita buat memiliki spesifikasi yang berisi 1 container dengan nama yang kita inginkan dan menggunakan image yang telah kita buat sebelumnya. Kemudian untuk resourcenya dibuat Batasan maksimum pada memori yang digunakan adalah 128 Megabyte dan CPU adalah 250 milicpu, yang dimana hitungan untuk 1 CPU=1000 milicpu. Lalu kita mengekspos port yang ada di dalam container, yaitu port 3000.

9. Kemudian, kita isi file dari service.yaml seperti pada gambar di bawah ini.



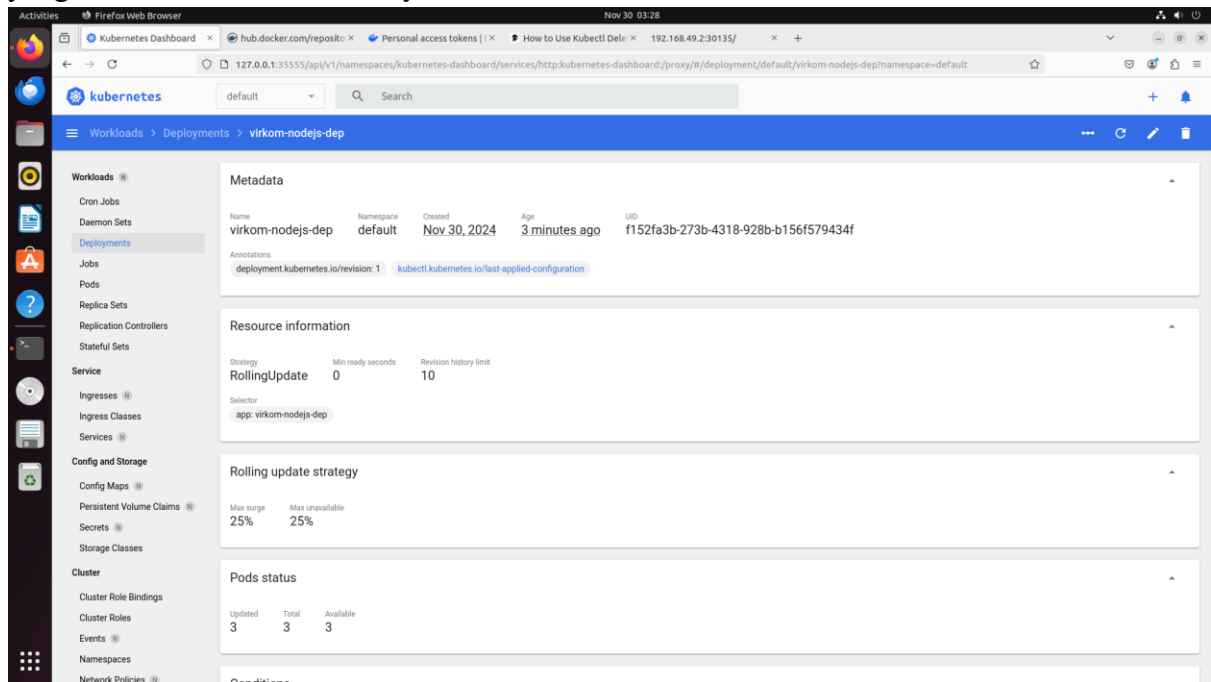
```
GNU nano 6.2 service.yaml *
apiVersion: v1
kind: Service
metadata:
  name: virkom-nodejs-dep
spec:
  type: NodePort
  selector:
    app: virkom-nodejs-dep
  ports:
  - port: 3030
    targetPort: 3000
```

10. Setelah kedua file sudah diisi, maka kita tinggal menerapkannya dengan perintah yang ada di gambar ini.

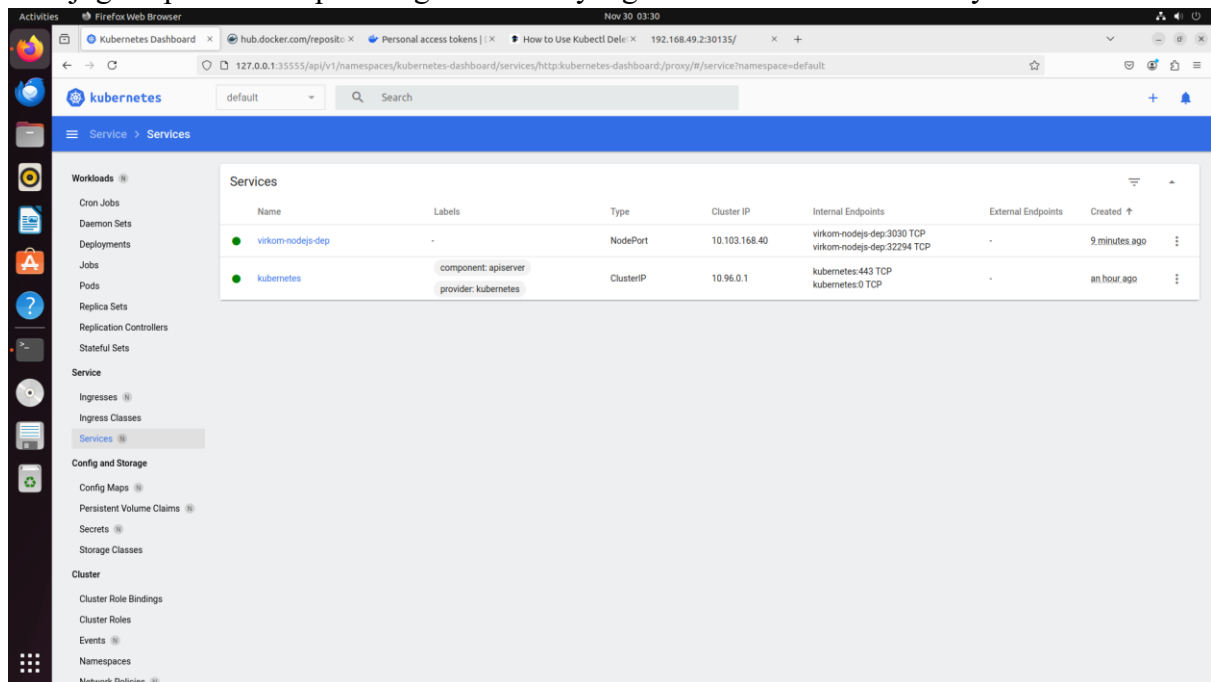


```
agus@agus:~/Kubernetes$ kubectl apply -f "*.yaml"
deployment.apps/virkom-nodejs-dep created
service/virkom-nodejs-dep unchanged
agus@agus:~/Kubernetes$
```

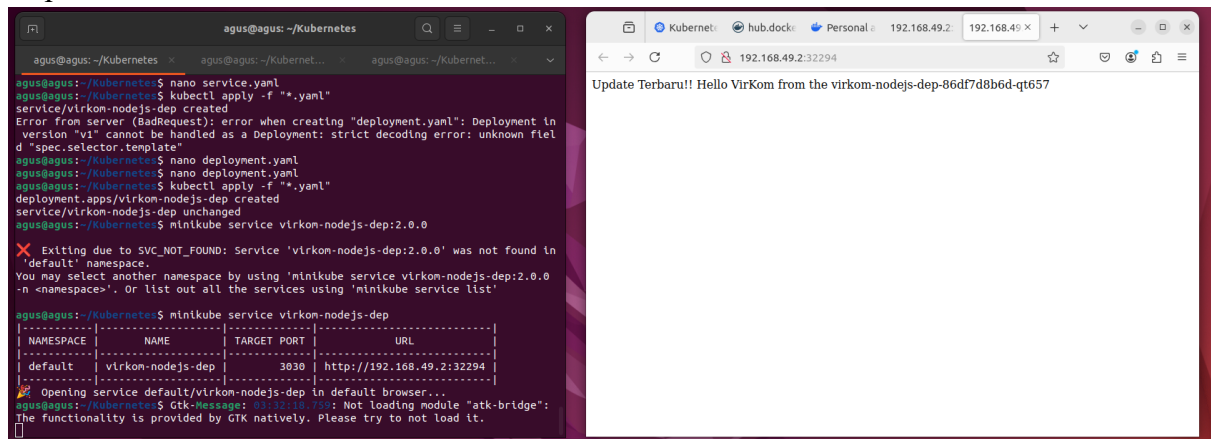
11. Kemudian, kita dapat melihat informasi detail di Dashboard Kubernetes dari deployment yang sudah kita buat sebelumnya.



12. Kita juga dapat melihat pada bagian service yang sudah kita buat sebelumnya.



13. Kemudian, kita verifikasi apakah aplikasi yang sudah kita buat tersebut memberikan respon.



The image shows a terminal window on the left and a web browser on the right. The terminal window displays the following commands and output:

```
agus@agus: ~/Kubernetes
agus@agus:~/Kubernetes$ nano service.yaml
agus@agus:~/Kubernetes$ kubectl apply -f *.yaml
service/virkon-nodejs-dep created
Error from server (BadRequest): error when creating "deployment.yaml": Deployment in
version "v1" cannot be handled as a Deployment: strict decoding error: unknown fiel
d "spec.selector.template"
agus@agus:~/Kubernetes$ nano deployment.yaml
agus@agus:~/Kubernetes$ kubectl apply -f *.yaml
deployment.apps/virkon-nodejs-dep created
service/virkon-nodejs-dep unchanged
agus@agus:~/Kubernetes$ minikube service virkon-nodejs-dep:2.0.0
✖ Exiting due to SVC_NOT_FOUND: Service 'virkon-nodejs-dep:2.0.0' was not found in
'default' namespace.
You may select another namespace by using 'minikube service virkon-nodejs-dep:2.0.0
-n <namespace>'. Or list out all the services using 'minikube service list'

agus@agus:~/Kubernetes$ minikube service virkon-nodejs-dep
|-----|-----|-----|-----|
| NAMESPACE | NAME | TARGET PORT | URL |
|-----|-----|-----|-----|
| default | virkon-nodejs-dep | 3030 | http://192.168.49.2:32294 |
|-----|-----|-----|-----|
🔗 Opening service default/virkon-nodejs-dep in default browser...
```

The web browser on the right shows the URL `192.168.49.2:32294` and displays the message: "Update Terbaru!! Hello VirKom from the virkon-nodejs-dep-86df7d8b6d-qt657".

14. Selesai!