Dokumentasi deploy REST API di Kubernetes

1. Membuat file deployment.yaml

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: rest-api
  labels:
    name: rest-api
spec:
  replicas: 5
  selector:
    matchLabels:
      name: rest-api
  template:
    metadata:
      name: rest-api
      labels:
        name: rest-api
    spec:
      containers:
      - name: restapi-apps
        image: hkdock01/nodejs-restapi
        ports:
        - containerPort: 8080

---

apiVersion: v1
kind: Service
metadata:
  name: rest-api-service
spec:
  type: NodePort
  selector:
    name: rest-api
  ports:
    - port: 8080
      targetPort: 8080
      nodePort: 30001
```

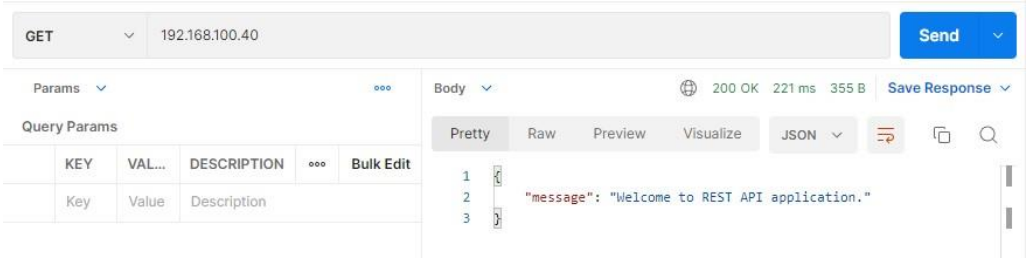
2. Deploy ke Kubernetes dengan printah kubectl apply -d deployment.yaml

```
ubuntu@ubuntu:~/nodejs-jwt-mongodb$ kubectl apply -f deployment.yaml
deployment.apps/rest-api created
service/rest-api-service created
```

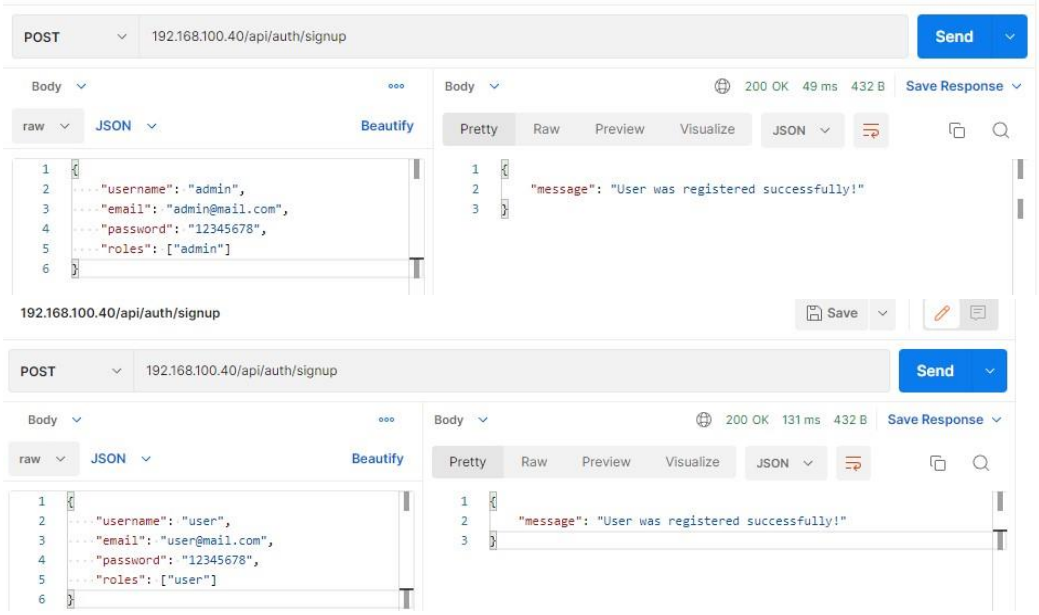
3. Cek pod dan service yang sudah running

```
ubuntu@ubuntu:~/nodejs-jwt-mongodb$ kubectl get all
NAME                            READY   STATUS    RESTARTS   AGE
pod/rest-api-574cfdf6d4-5dl4l   1/1     Running   0          27s
pod/rest-api-574cfdf6d4-bg8rh   1/1     Running   0          27s
pod/rest-api-574cfdf6d4-c46ws   1/1     Running   0          27s
pod/rest-api-574cfdf6d4-mpw9z   1/1     Running   0          27s
pod/rest-api-574cfdf6d4-ndld4   1/1     Running   0          27s

NAME                       TYPE        CLUSTER-IP    EXTERNAL-IP   PORT(S)          AGE
service/kubernetes         ClusterIP   10.96.0.1     <none>        443/TCP          9d
service/rest-api-service   NodePort    10.97.34.34   <none>        8080:30001/TCP   27s

NAME                       READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/rest-api   5/5     5            5           27s

NAME                                  DESIRED   CURRENT   READY   AGE
replicaset.apps/rest-api-574cfdf6d4   5         5         5       27s
```

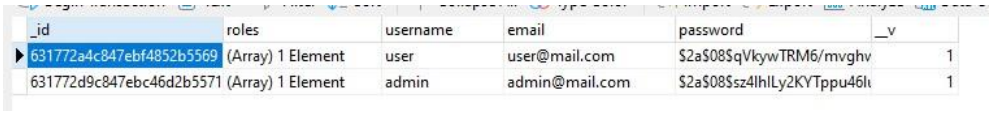
4. Access API dengan postman



5. Membuat user dengan nama admin role admin dan user role user

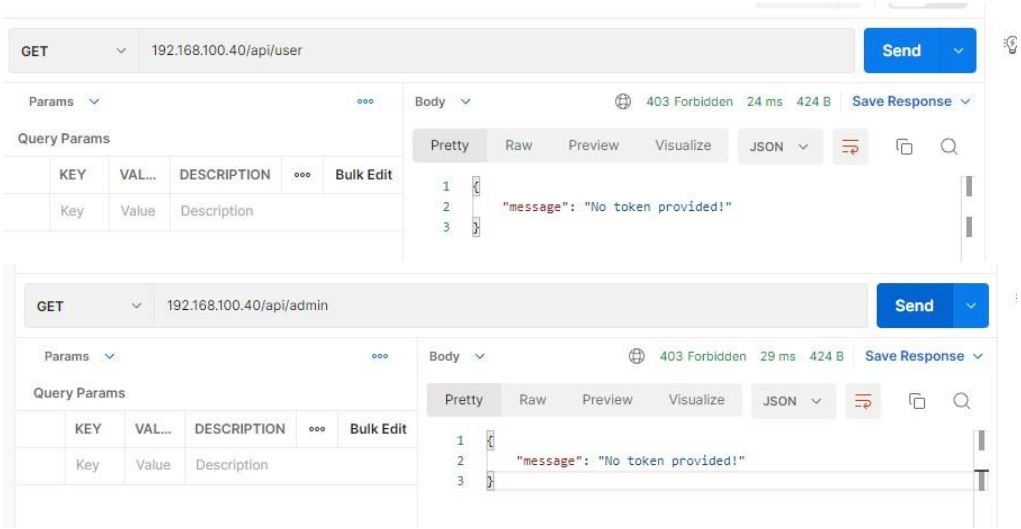


6. Cek di database, user sudah ter-create

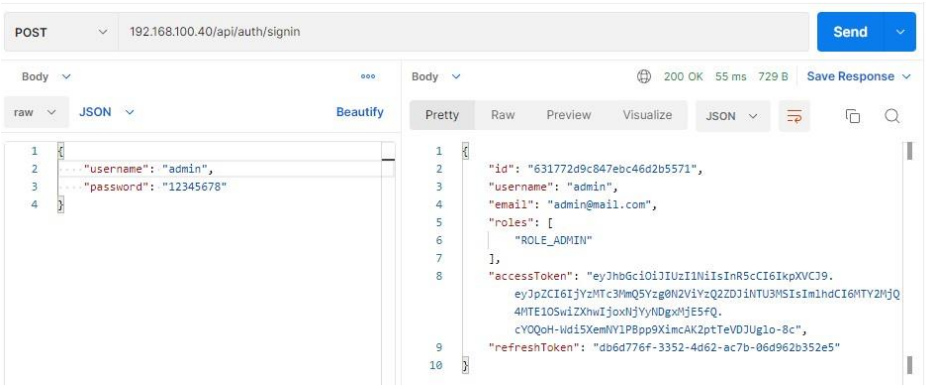


_id	roles	username	email	password	__v
631772a4c847ebf4852b5569	(Array) 1 Element	user	user@mail.com	$2a$08$qVkywTRM6/mvghv	1
631772d9c847ebc46d2b5571	(Array) 1 Element	admin	admin@mail.com	$2a$08$sz4lhlLy2KYTppu46l	1

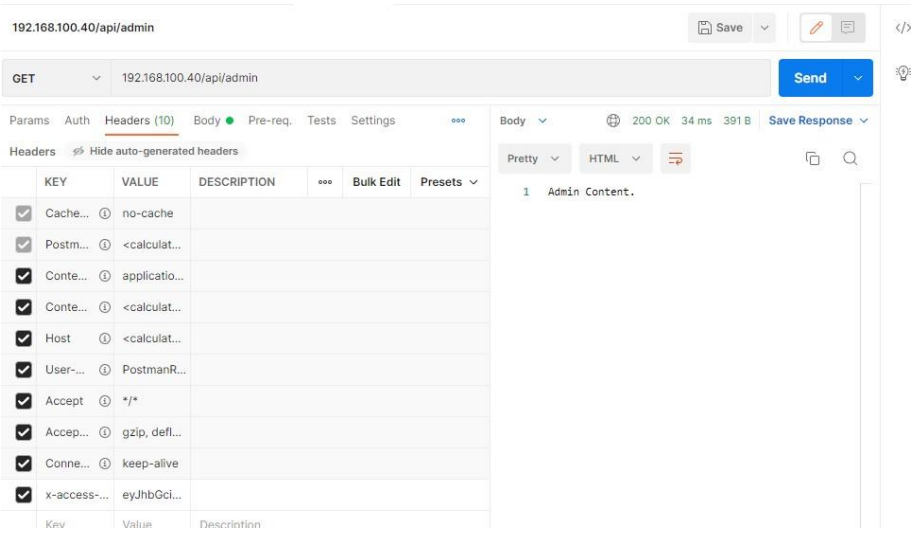
7. Untuk melihat list tutorial dengan endpoint /api/user , /api/admin. Karena belum signup maka muncul "No token provided!"



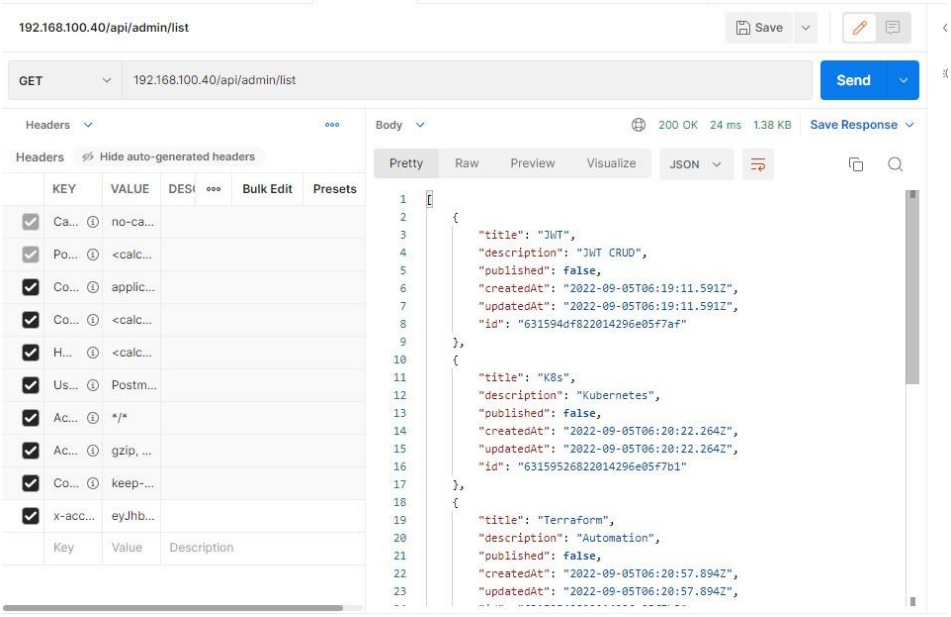
8. Signin dengan user admin



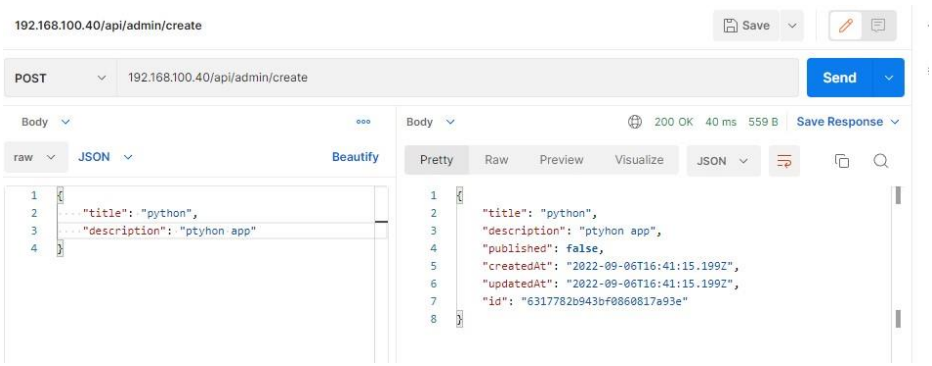
9. Tambahkan access token di header



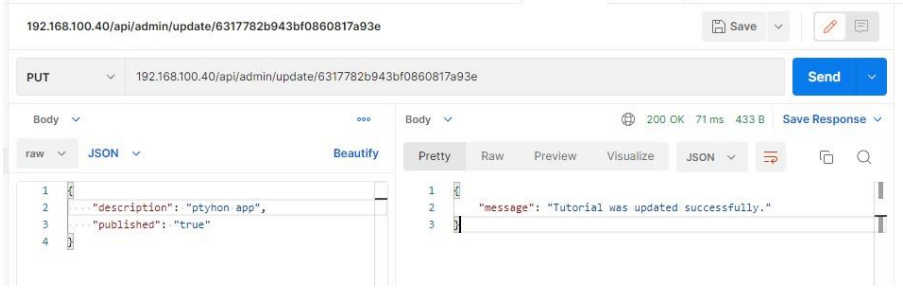
10. Melihat list tutorial sudah bisa di akses



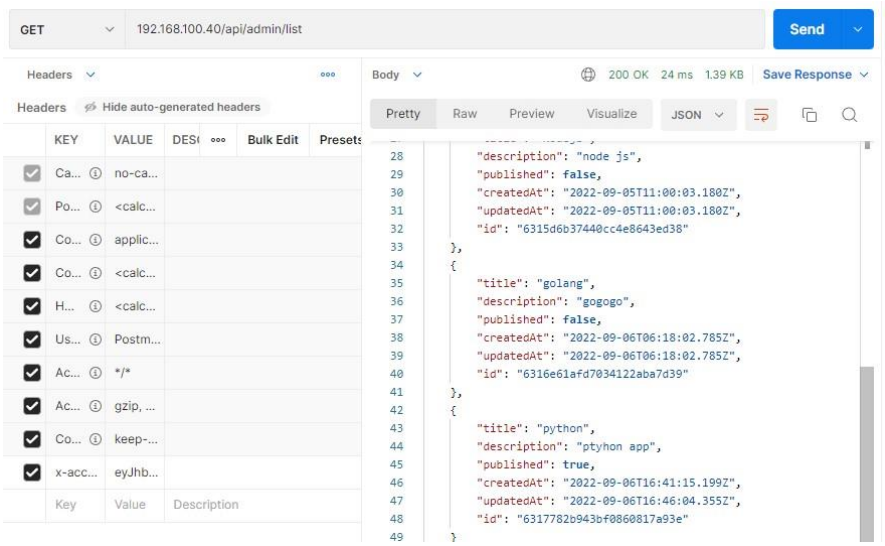
11. Create tutorial baru



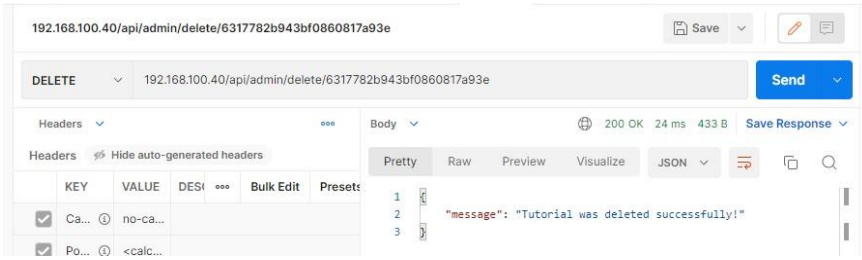
12. Update tutorial python, dengan published menjadi true



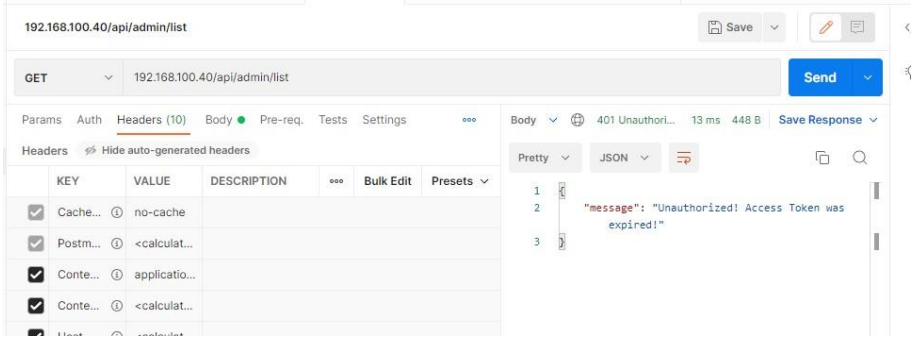
13. Cek tutorial pyhton yang sudah terupdate



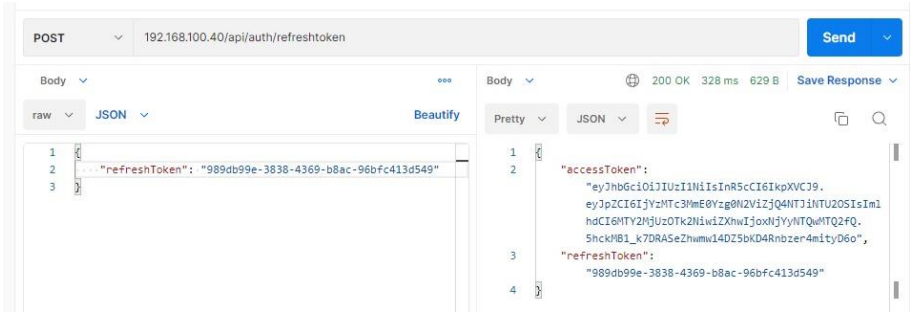
14. Delete tutorial python dengan id “6317782b943bf0860817a93e”



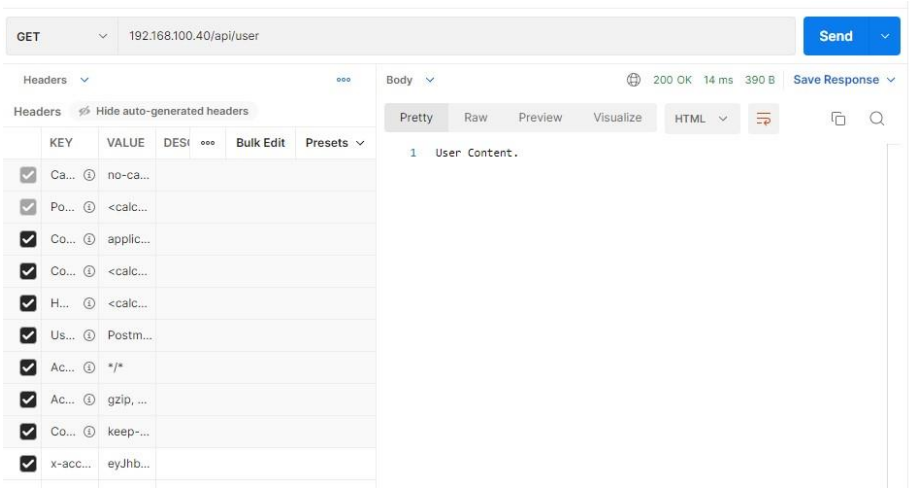
15. Access token expired



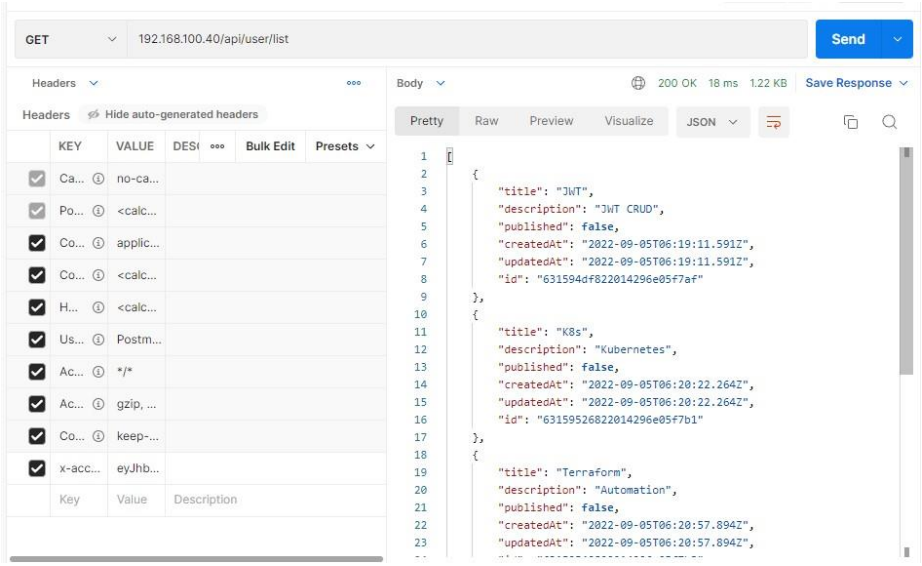
16. Refresh token



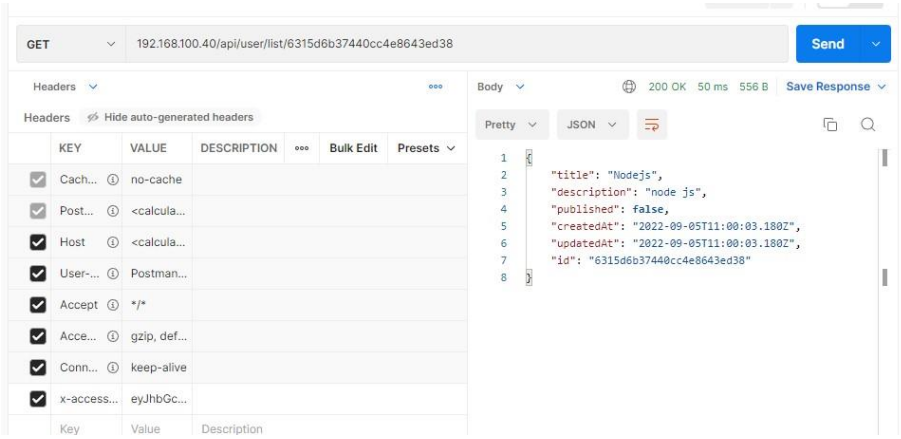
17. Signin dengan role user



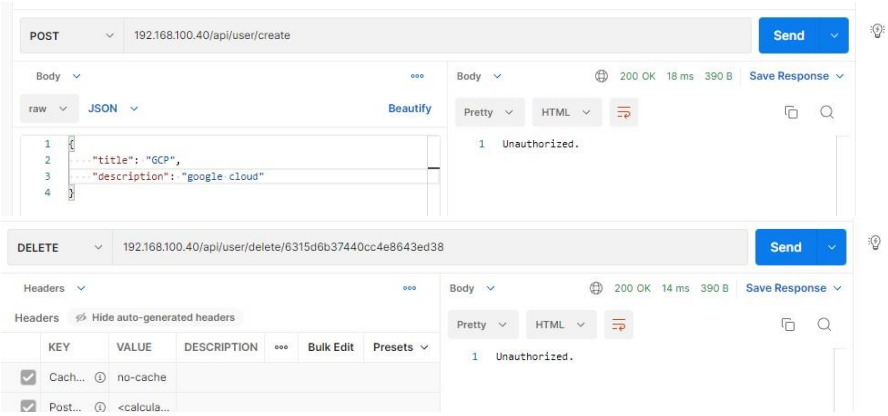
18. Melihat list tutorial dengan role user



19. Melihat list tutorial dengan spesifik id



20. Unauthorized pada role user



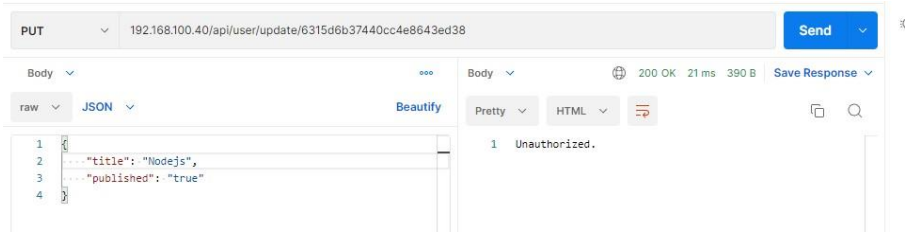


Diagram arsitektur REST API