

1. Triển khai Kubernetes

- Triển khai một Kubernetes Cluster gồm 1 Master node và 1 Worker node trên nền tảng Ubuntu 22.04.
- Lựa chọn triển khai **K3s – Lightweight Kubernetes**
- Mô hình hệ thống chuẩn bị:

Node	Vai trò	Hệ điều hành	IP
master1	K3s Server	Ubuntu 22.04	192.168.126.102
worker	K3s Worker	Ubuntu 22.04	192.168.126.100

- Cài đặt cơ bản:

```
sudo apt update -y
sudo apt install -y curl
```

- Cài đặt K3s (Master node):

- SSH vào master node: `ssh devops@192.168.126.102`
- Cài đặt K3s: `curl -sfL https://get.k3s.io | sh`
- Kiểm tra trạng thái: `systemctl status k3s`
- Kiểm tra node: `sudo kubectl get nodes`

```
devops@master1:~$ systemctl status k3s
● k3s.service - Lightweight Kubernetes
  Loaded: loaded (/etc/systemd/system/k3s.service; enabled; vendor prese>
  Active: active (running) since Thu 2026-01-15 07:40:11 UTC; 30s ago
    Docs: https://k3s.io
  Process: 2561 ExecStartPre=/sbin/modprobe br_netfilter (code=exited, st>
  Process: 2564 ExecStartPre=/sbin/modprobe overlay (code=exited, status=>
  Main PID: 2568 (k3s-server)
    Tasks: 23
   Memory: 610.7M
      CPU: 0.100 seconds
     CGroup: /system.slice/k3s.service
             └─2568 /usr/local/bin/k3s server
                  ├─2604 containerd
devops@master1:~$ sudo kubectl get nodes
NAME      STATUS    ROLES      AGE      VERSION
master1   Ready     control-plane   51s    v1.34.3+k3s1
```

- Lấy token để join Worker `sudo cat /var/lib/rancher/k3s/server/node-token`
- Cài đặt K3s Agent (Worker node):
- SSH vào worker node `ssh worker@192.168.126.100`
- Join worker vào cluster

```

worker@worker:~$ curl -sfL https://get.k3s.io | \
> K3S_URL=https://192.168.126.102:6443 \
> K3S_TOKEN=f7e64011be0879d362fded79497fef43 \
> sh -
[INFO] Finding release for channel stable
[INFO] Using v1.34.3+k3s1 as release
[INFO] Downloading hash https://github.com/k3s-io/k3s/releases/download/v1.
34.3+k3s1/sha256sum-amd64.txt
[INFO] Downloading binary https://github.com/k3s-io/k3s/releases/download/v
1.34.3+k3s1/k3s
[INFO] Verifying binary download
[INFO] Installing k3s to /usr/local/bin/k3s
[INFO] Skipping installation of SELinux RPM
[INFO] Creating /usr/local/bin/kubectl symlink to k3s
[INFO] Skipping /usr/local/bin/crictl symlink to k3s, already exists
[INFO] Skipping /usr/local/bin/ctr symlink to k3s, already exists
[INFO] Creating killall script /usr/local/bin/k3s-killall.sh
[INFO] Creating uninstall script /usr/local/bin/k3s-agent-uninstall.sh
[INFO] env: Creating environment file /etc/systemd/system/k3s-agent.service
.env
[INFO] systemd: Creating service file /etc/systemd/system/k3s-agent.service
[INFO] systemd: Enabling k3s-agent unit
Created symlink /etc/systemd/system/multi-user.target.wants/k3s-agent.servic
e → /etc/systemd/system/k3s-agent.service.
[INFO] systemd: Starting k3s-agent

```

- Kiểm tra trạng thái: `sudo kubectl get nodes -o wide`

NAME	STATUS	ROLES	AGE	VERSION	INTERNAL-IP	EXTERNAL-IP	OS-IMAGE	KERNEL-VERSION	CONTAINER-RUNTIME
master1	Ready	control-plane	7m44s	v1.34.3+k3s1	192.168.126.102	<none>	Ubuntu 20.04.6 LTS	5.4.0-216-generic	containerd://2.1.5-k3s1
worker	Ready	<none>	27s	v1.34.3+k3s1	192.168.126.100	<none>	Ubuntu 20.04.6 LTS	5.4.0-216-generic	containerd://2.1.5-k3s1

```

devops@master1:~$ sudo kubectl get pods -A -o wide
NAMESPACE     NAME           READY   STATUS    RESTARTS   AGE   IP          NODE      NOMINATED NODE   READINESS GATES
kube-system   coredns-7f496c8d7d-8cxnf   1/1    Running   0          35m   10.42.0.4   master1   <none>        <none>
kube-system   helm-install-traefik-4gx94   0/1    Completed  2          35m   10.42.0.3   master1   <none>        <none>
kube-system   helm-install-traefik-crd-8p2m7  0/1    Completed  0          35m   10.42.0.2   master1   <none>        <none>
kube-system   local-path-provisioner-578895bd58-chpsd  1/1    Running   0          35m   10.42.0.6   master1   <none>        <none>
kube-system   metrics-server-7b9e9c4b9c-bcsfs   1/1    Running   0          35m   10.42.0.5   master1   <none>        <none>
kube-system   svclb-traefik-0a3c4d2d-6lgnn   2/2    Running   0          33m   10.42.0.7   master1   <none>        <none>
kube-system   svclb-traefik-0a3c4d2d-sw9f8   2/2    Running   0          28m   10.42.1.2   worker   <none>        <none>
kube-system   traefik-6f5f87584-8hqq4   1/1    Running   0          33m   10.42.0.8   master1   <none>        <none>
devops@master1:~$ 
```

2. Triển khai web application sử dụng các DevOps tools & practices

K8S Helm Chart (1.5đ)

2.1. Yêu cầu 1

- Cài đặt ArgoCD lên Kubernetes Cluster, expose được qua ArgoCD qua NodePort
 - Tạo namespace và triển khai bằng cách áp dụng các file từ chính kho lưu trữ của dự án Argi Project:

```
kubectl create namespace argocd
```

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

```
devops@master1:~/k8s-practice/lab6$ kubectl create namespace argocd
namespace/argocd created
devops@master1:~/k8s-practice/lab6$ kubectl apply -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
clusterrolebinding.rbac.authorization.k8s.io/argocd-server created
configmap/argocd-cm created
configmap/argocd-cmd-params-cm created
configmap/argocd-gpg-keys-cm created
configmap/argocd-notifications-cm created
configmap/argocd-rbac-cm created
configmap/argocd-ssh-known-hosts-cm created
configmap/argocd-tls-certs-cm created
secret/argocd-notifications-secret created
```

```
pod "argocd-server-57b47b59b6-ccmnmm" deleted from argocd namespace
devops@master1:~$ kubectl get pods -n argocd
```

NAME	READY	STATUS	RESTARTS	AGE
argocd-application-controller-0	1/1	Running	0	2m48s
argocd-applicationset-controller-7d97f7b47d-ph842	1/1	Running	0	3m31s
argocd-dex-server-657f5854c4-vhpsj	1/1	Running	0	3m29s
argocd-notifications-controller-799d98bf9f-xntqb	1/1	Running	0	3m20s
argocd-redis-5b69b8d789-phg8p	1/1	Running	0	3m14s
argocd-repo-server-569b656c5-mh4wc	1/1	Running	0	3m2s
argocd-server-57b47b59b6-p4jnf	1/1	Running	0	2m48s

- Expose qua NodePort

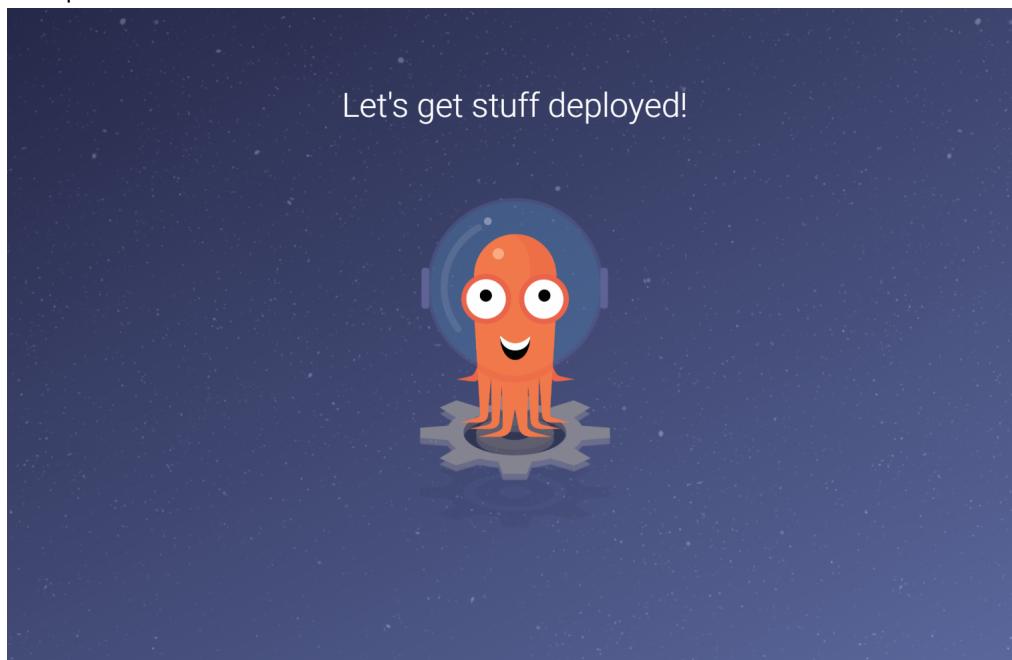
```
kubectl patch svc argocd-server -n argocd -p '{"spec":{"type":"NodePort"}}'
```

```
devops@master1:~$ kubectl patch svc argocd-server -n argocd -p '{"spec":{"type":"NodePort"}}'
service/argocd-server patched
devops@master1:~$ kubectl get svc argocd-server -n argocd
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)           AGE
argocd-server  NodePort  10.43.95.34  <none>          80:32436/TCP,443:32607/TCP  18m
```

- Lấy mật khẩu:

```
kubectl -n argocd get secret argocd-initial-admin-secret -o jsonpath=".data.password" | base64 -d
```

- Kết quả:



- **Cài đặt Jenkins** lên Kubernetes Cluster, expose được Jenkins qua NodePort

- Cài đặt và triển khai:

```
kubectl apply -f jenkins/jenkins_namespace.yml
```

```
devops@master1:~$ kubectl get pods -n jenkins
NAME           READY   STATUS    RESTARTS   AGE
jenkins-5dd88f7bf9-7vtvp   1/1     Running   0          7m14s
devops@master1:~$ kubectl get pv -n jenkins
NAME          CAPACITY   ACCESS MODES  RECLAIM POLICY  STATUS    CLAIM
S VOLUME ATTRIBUTE CLASS   REASON   AGE
jenkins-pv      5Gi       RWO          Retain        Available
<unset>          10m
pvc-ef310b11-c2ca-4562-99db-3abb5af1cb4c  5Gi       RWO          Delete        Bound    jenkins/jenkins-pvc  local-path
<unset>          9m4s
devops@master1:~$ kubectl get pvc -n jenkins
NAME           STATUS    VOLUME
AGE
jenkins-pvc   Bound    pvc-ef310b11-c2ca-4562-99db-3abb5af1cb4c  5Gi
10m
devops@master1:~$ kubectl get svc -n jenkins
NAME      TYPE      CLUSTER-IP      EXTERNAL-IP      PORT(S)      AGE
jenkins-service  NodePort  10.43.219.49  <none>        8080:32080/TCP,50000:32081/TCP  10m
devops@master1:~$ |
```

- Lấy mật khẩu:

```
kubectl exec -it jenkins-5dd88f7bf9-7vtvp -n jenkins -- cat  
/var/jenkins_home/secrets/initialAdminPassword
```

```
devops@master1:~$ kubectl exec -it jenkins-5dd88f7bf9-7vtvp -n jenkins -- cat /var/jenkins_home/secrets/initialAdminPassword  
fb4e2e37053e40709431324f6f0c5060  
devops@master1:~$ |
```

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

`/var/jenkins_home/secrets/initialAdminPassword`

Please copy the password from either location and paste it below.

Administrator password

.....

Continue

Getting Started

Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

Install suggested plugins

Install plugins the Jenkins community finds most useful.

Select plugins to install

Select and install plugins most suitable for your needs.

Jenkins 2.528.3

2.2. Yêu cầu 2

- **Source code & Helm Chart:** [fhcoffee_v2](#)
- **Config Repo:** [fh-coffee-config](#)
- Cài đặt Helm Chart:

```
curl -fsSL -o get_helm.sh
https://raw.githubusercontent.com/helm/helm/main/scripts/get-helm-3

chmod 700 get_helm.sh

./get_helm.sh
```

```
helm version
```

- Khởi tạo cấu trúc

```
mkdir -p ~/k8s-practice/cuoikhoa/fhcoffee_v2/charts  
cd ~/k8s-practice/cuoikhoa/fhcoffee_v2/charts  
helm create fh-coffee-chart
```

- Kiểm tra các cú pháp trong helm:

```
helm lint fh-coffee-chart/  
helm template fh-coffee ./fh-coffee-chart
```

- File Manifest ArgoCD Application:

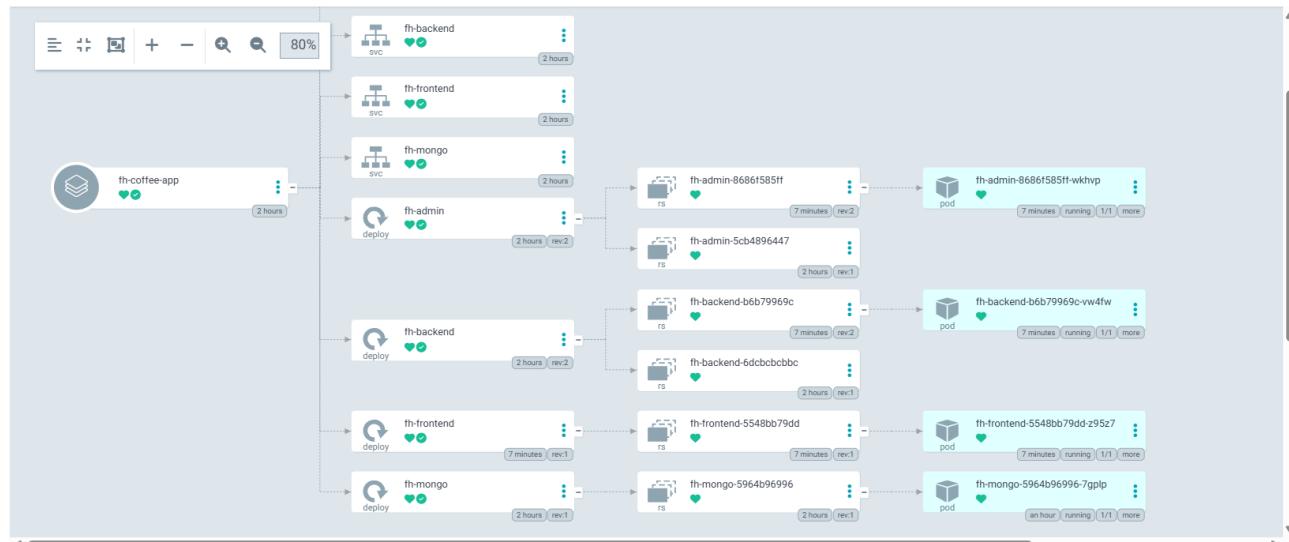
```
apiVersion: argoproj.io/v1alpha1  
kind: Application  
metadata:  
  name: fh-coffee-app  
  namespace: argocd  
spec:  
  project: default  
  sources:  
    - repoURL: 'https://github.com/fhptth/fhcoffee_v2.git'  
      targetRevision: HEAD  
      path: charts/fh-coffee-chart  
      helm:  
        valueFiles:  
          - $values/values-prod.yaml  
    - repoURL: 'https://github.com/fhptth/fh-coffee-config.git'  
      targetRevision: HEAD  
      ref: values  
  destination:  
    server: 'https://kubernetes.default.svc'  
    namespace: cloud-final  
  syncPolicy:  
    automated:  
      prune: true  
      selfHeal: true  
  syncOptions:  
    - CreateNamespace=true
```

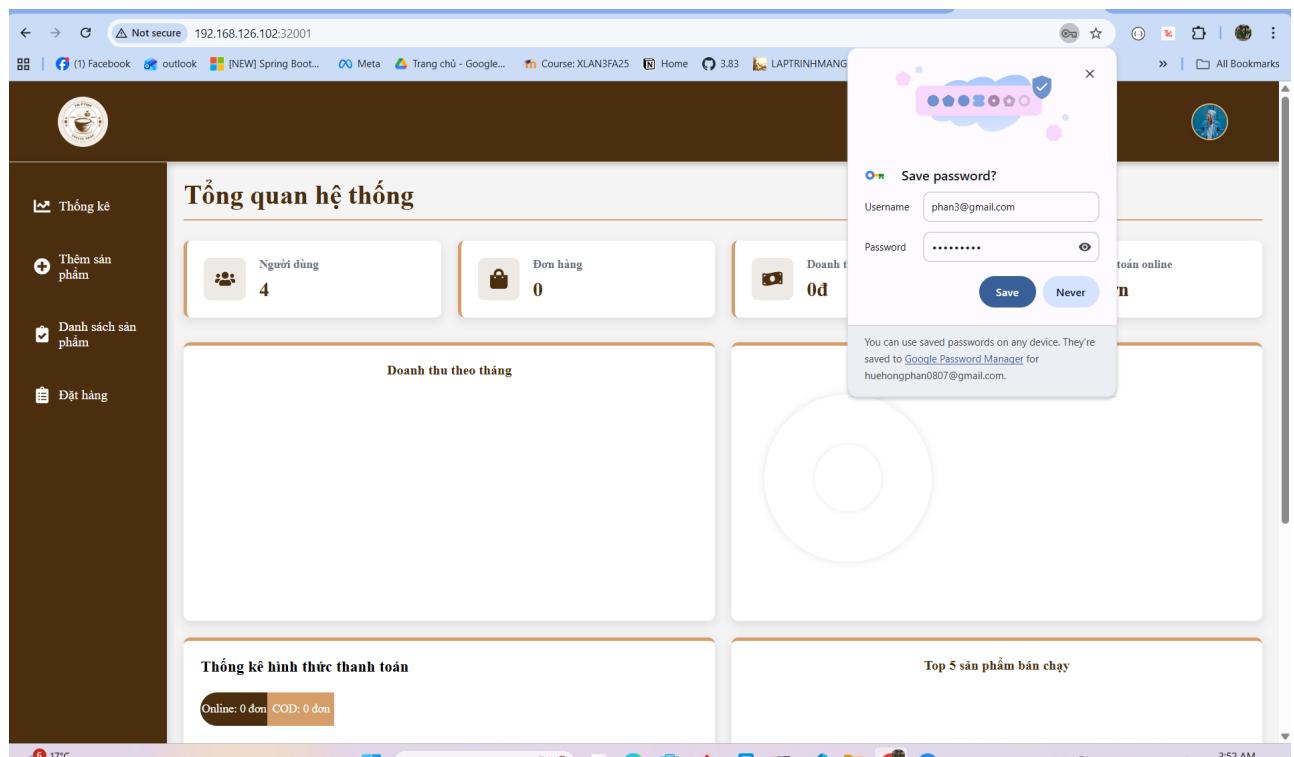
- Để chạy Helm Chart:

```
kubectl apply -f application.yaml
```

Kết quả:

The screenshot shows the Argo UI interface. On the left, there's a sidebar with the Argo logo and version (v3.2.5+c56f440). It has sections for Applications, Settings, User Info, and Documentation. Below these are filters for SYNC STATUS (Unknown: 0, Synced: 1, OutOfSync: 0) and HEALTH STATUS (Progressing: 0, Suspended: 0, Healthy: 1, Degraded: 0, Missing: 0, Unknown: 0). The main area displays the 'fh-coffee-app' application details. It includes fields like Project: default, Labels: healthy, Status: Synced, Repository: https://github.com/fhptth/fhcoffee_v2.git, Target Ref: HEAD, Path: charts/fh-coffee-chart, Destination: in-cluster, Namespace: cloud-final, Created: 01/21/2026 00:44:57 (3 hours ago), and Last Sync: 01/21/2026 03:48:44 (21 minutes ago). Buttons for SYNC, REFRESH, and DELETE are at the bottom.





CI/CD

- Trigger:
 - Đổi http sang https để add vào webhook trên github

```
ngrok http 192.168.126.100:32080
```

```
ngrok
(Ctrl+C to quit)

⚠ Free Users: Agents ≤3.19.x stop connecting 2/17/26. Update or upgrade: ht

Session Status          online
Account                  Phan Thi Hong Hue (Plan: Free)
Update                   update available (version 3.35.0, Ctrl-U to up
Version                 3.34.1
Region                  Asia Pacific (ap)
Latency                 168ms
Web Interface           http://127.0.0.1:4040
                         https://uncultivable-consistently-lavonna.ng

Connections             ttl     opn     rt1     rt5     p50     p90
                         1       0       0.01   0.00   34.82   34.82

HTTP Requests
-----
09:56:04.235 UTC POST /github-webhook/          200 OK
```

The screenshot shows the GitHub Settings page for the repository 'fhpthh/fhcoffee_v2'. The 'Webhooks' section is active. On the left, a sidebar lists settings like General, Access, Collaborators, and Code and automation. The main area displays a single webhook entry:

- URL:** https://uncultivable-consistently-i... (push)
- Status:** Last delivery was successful.
- Actions:** Edit | Delete