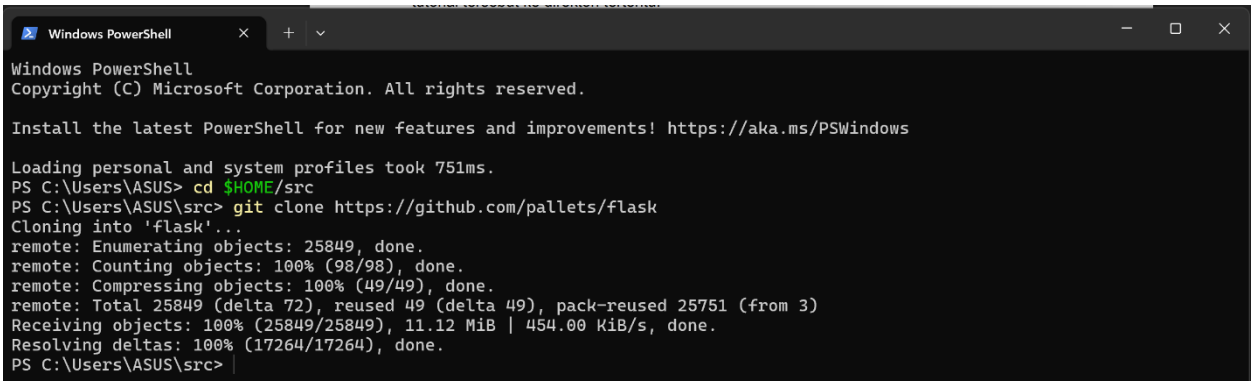
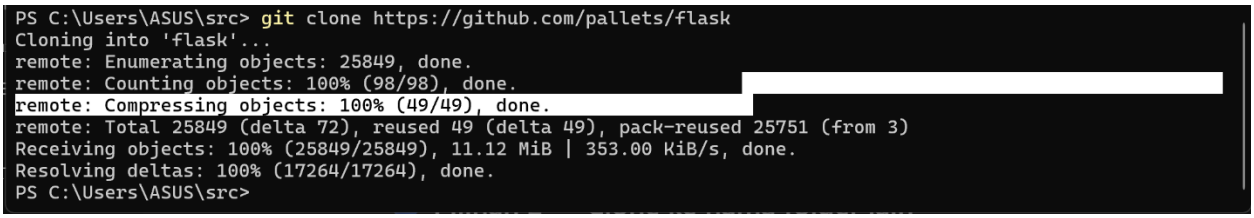
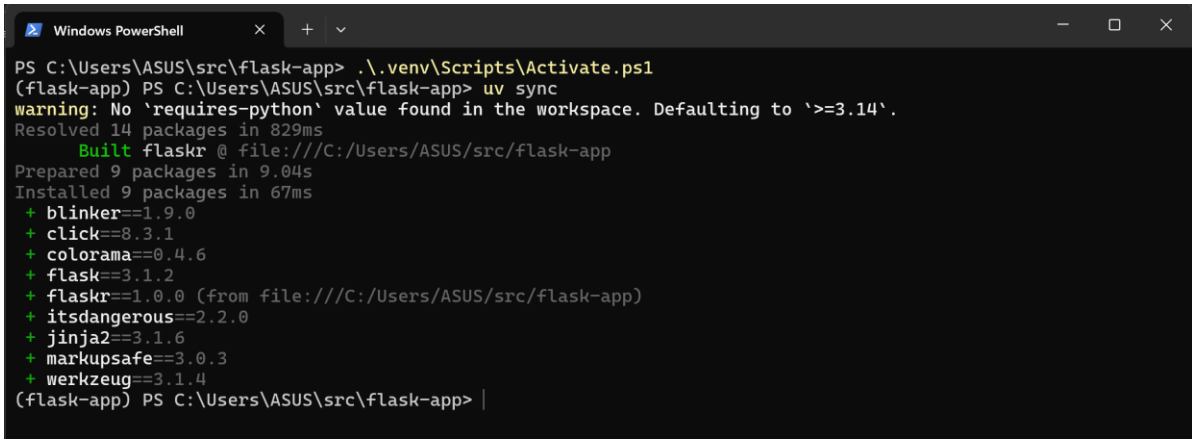
Pertemuan 9

1. Membuat env Python 3.14t (Free Threaded)

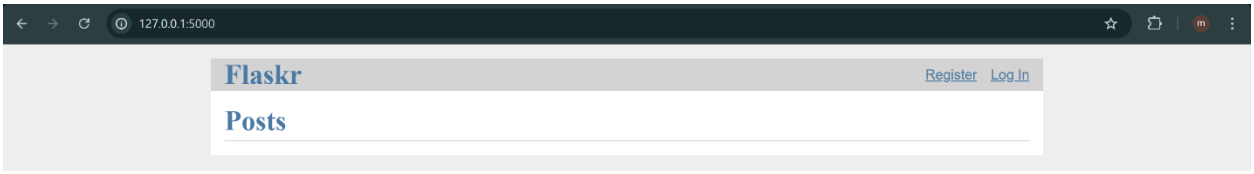


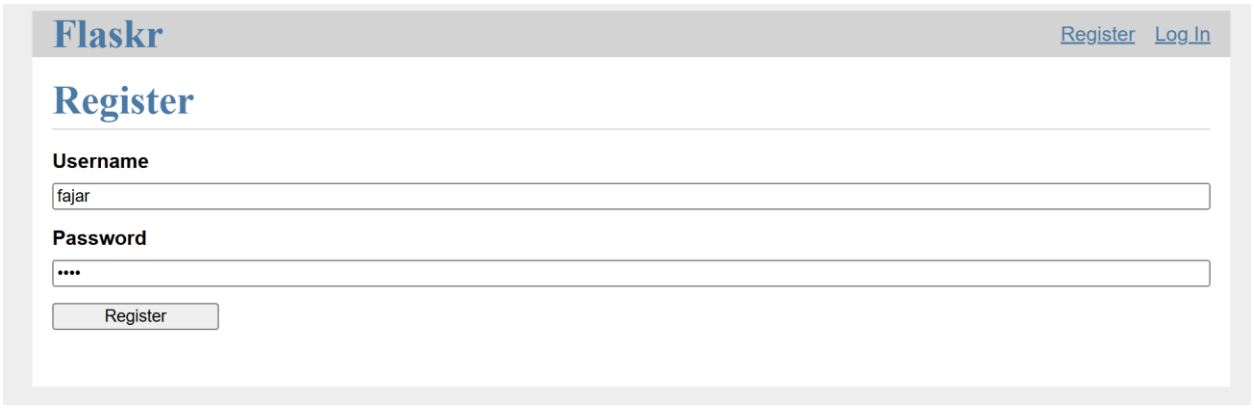


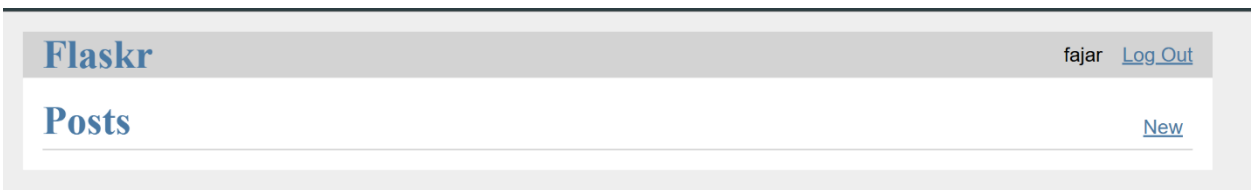


2. Menjalankan Aplikasi Flask

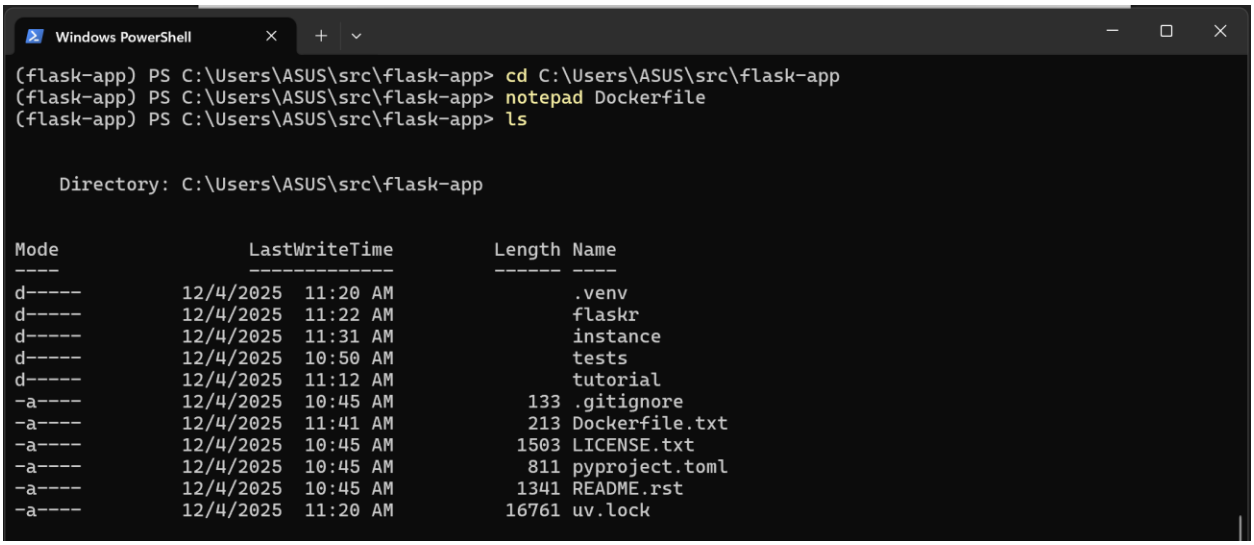
```
(flask-app) PS C:\Users\ASUS\src\flask-app> flask --app flaskr init-db
Initialized the database.
(flask-app) PS C:\Users\ASUS\src\flask-app> flask --app flaskr run
 * Serving Flask app 'flaskr'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI serve
r instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

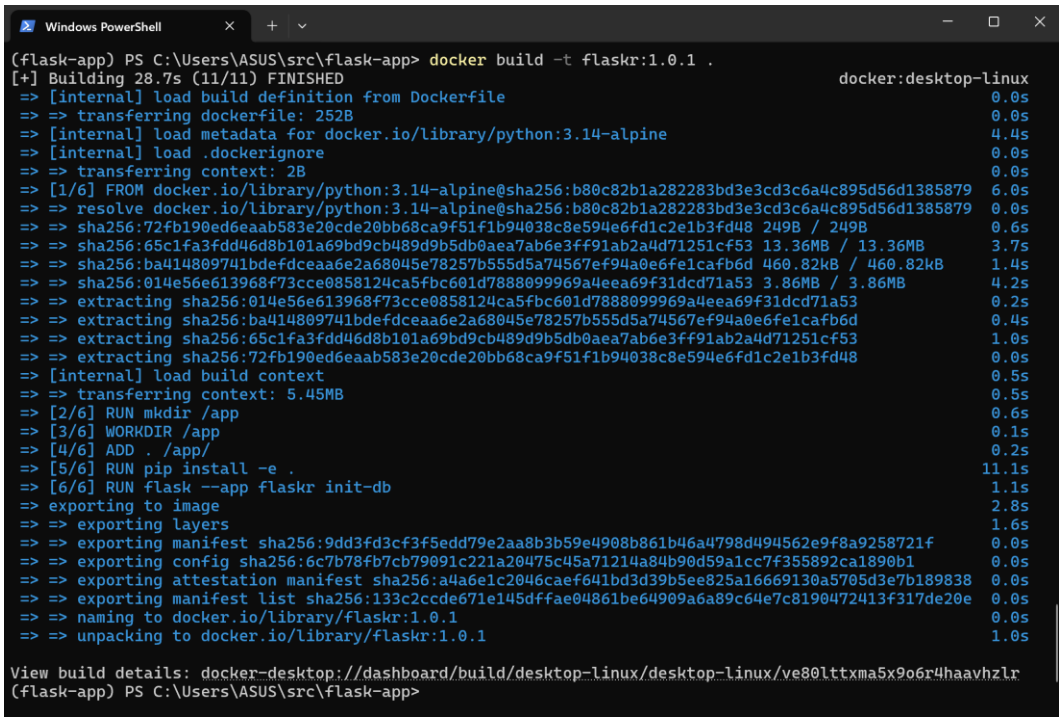




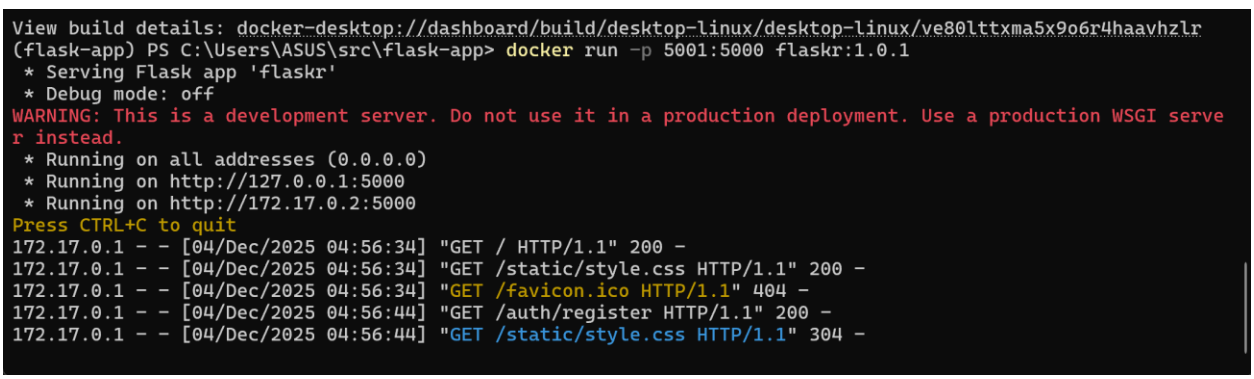


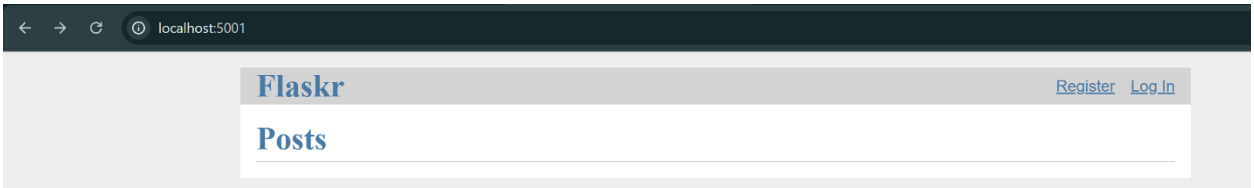
3. Buat Aplikasi Menjadi CA





4. Menjalankan Image





5. Menggunakan Kubernetes

Ambil file di Kubernetes.io/docs/tasks/tools

Penginstallan Kind

```
PS C:\Users\ASUS\src\flask-app> kind version
kind v0.30.0 go1.24.6 windows/amd64
PS C:\Users\ASUS\src\flask-app>
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\ASUS> cd C:\Users\ASUS\src\flask-app
PS C:\Users\ASUS\src\flask-app> kind version
kind v0.30.0 go1.24.6 windows/amd64
PS C:\Users\ASUS\src\flask-app> kind create cluster --name flask-cluster
Creating cluster "flask-cluster" ...
 • Ensuring node image (kindest/node:v1.34.0) 🖼 ...
 ✓ Ensuring node image (kindest/node:v1.34.0) 🖼
 • Preparing nodes 📦   ...
 ✓ Preparing nodes 📦
 • Writing configuration 📜  ...
 ✓ Writing configuration 📜
 • Starting control-plane 🕹️  ...
 ✓ Starting control-plane 🕹️
 • Installing CNI 🔌  ...
 ✓ Installing CNI 🔌
 • Installing StorageClass 💾  ...
 ✓ Installing StorageClass 💾
Set kubectl context to "kind-flask-cluster"
You can now use your cluster with:
```

```
PS C:\Users\ASUS\src\flask-app> kubectl cluster-info
Kubernetes control plane is running at https://127.0.0.1:58416
CoreDNS is running at https://127.0.0.1:58416/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
PS C:\Users\ASUS\src\flask-app>
```

```
PS C:\Users\ASUS\src\flask-app> docker container ls
CONTAINER ID   IMAGE                  COMMAND                  CREATED          STATUS          PORTS
    NAMES
c0c82a69fbcb   kindest/node:v1.34.0   "/usr/local/bin/entr…"   12 minutes ago   Up 11 minutes   127.0.0.1:58416->6443/tc
p   flask-cluster-control-plane
PS C:\Users\ASUS\src\flask-app>
```

```
PS C:\Users\ASUS\src\flask-app> kind load docker-image flaskr:1.0.1 --name flask-cluster
Image: "flaskr:1.0.1" with ID "sha256:133c2ccde671e145dffae04861be64909a6a89c64e7c8190472413f317de20eb" not yet present
on node "flask-cluster-control-plane", loading...
PS C:\Users\ASUS\src\flask-app>
```

```
PS C:\Users\ASUS\src\flask-app> kubectl apply -f deployment.yaml
error: error when retrieving current configuration of:
Resource: "apps/v1, Resource=deployments", GroupVersionKind: "apps/v1, Kind=Deployment"
Name: "", Namespace: "default"
from server for: "deployment.yaml": resource name may not be empty
PS C:\Users\ASUS\src\flask-app> kubectl apply -f service.yaml
error: error when retrieving current configuration of:
Resource: "/v1, Resource=services", GroupVersionKind: "/v1, Kind=Service"
Name: "", Namespace: "default"
from server for: "service.yaml": resource name may not be empty
```

```
PS C:\Users\ASUS\src\flask-app> kubectl get svc
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   32m
PS C:\Users\ASUS\src\flask-app>
```

```
Run 'docker inspect --help' for more information
PS C:\Users\ASUS\src\flask-app> docker inspect -f "{{.NetworkSettings.Networks.kind.IPAddress}}" flask-cluster-control-p
lane
172.18.0.2
PS C:\Users\ASUS\src\flask-app>
```

```
PS C:\Users\ASUS\src\flask-app> kubectl get svc
NAME         TYPE        CLUSTER-IP   EXTERNAL-IP   PORT(S)   AGE
kubernetes   ClusterIP   10.96.0.1    <none>        443/TCP   37m
PS C:\Users\ASUS\src\flask-app>
```