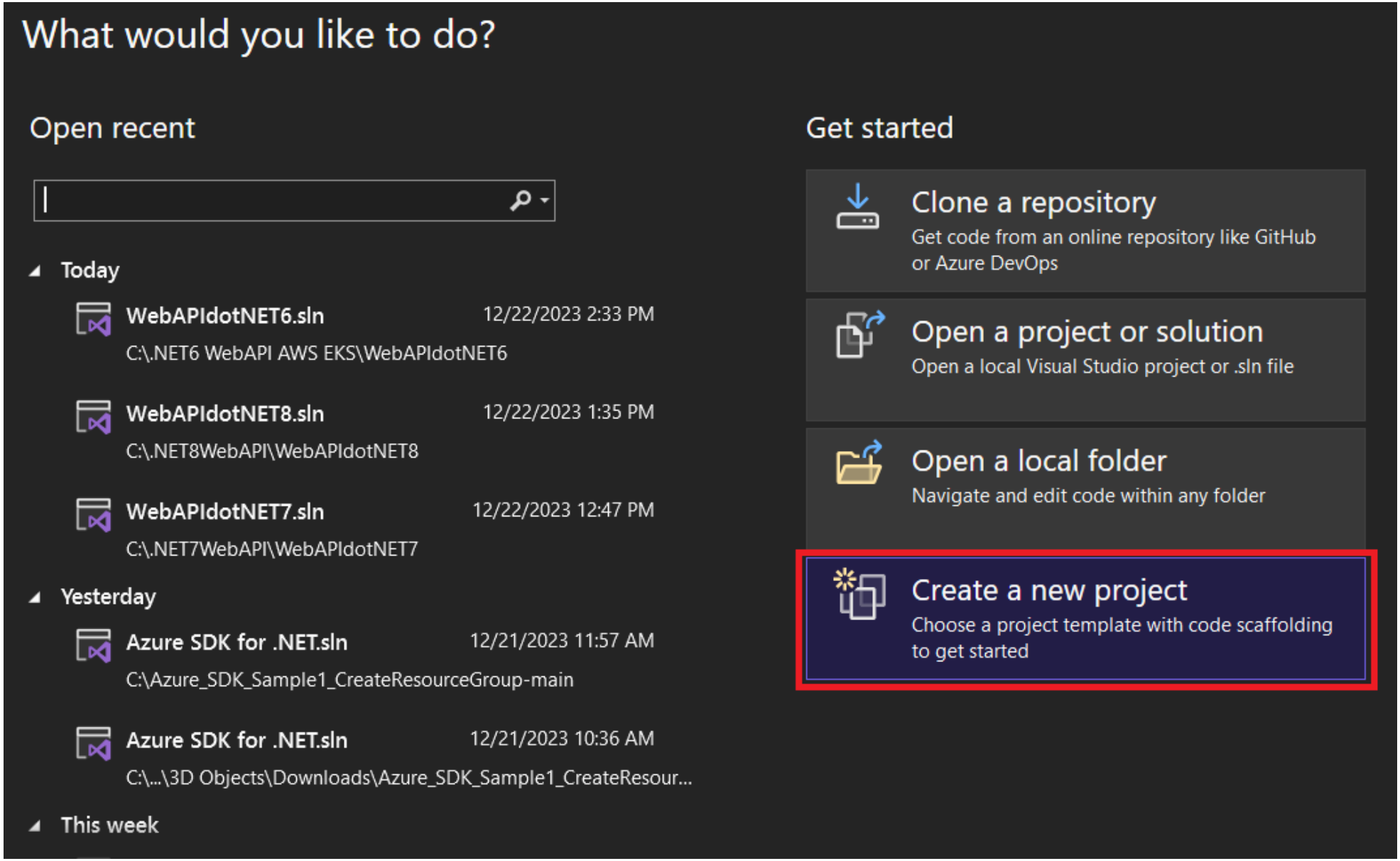
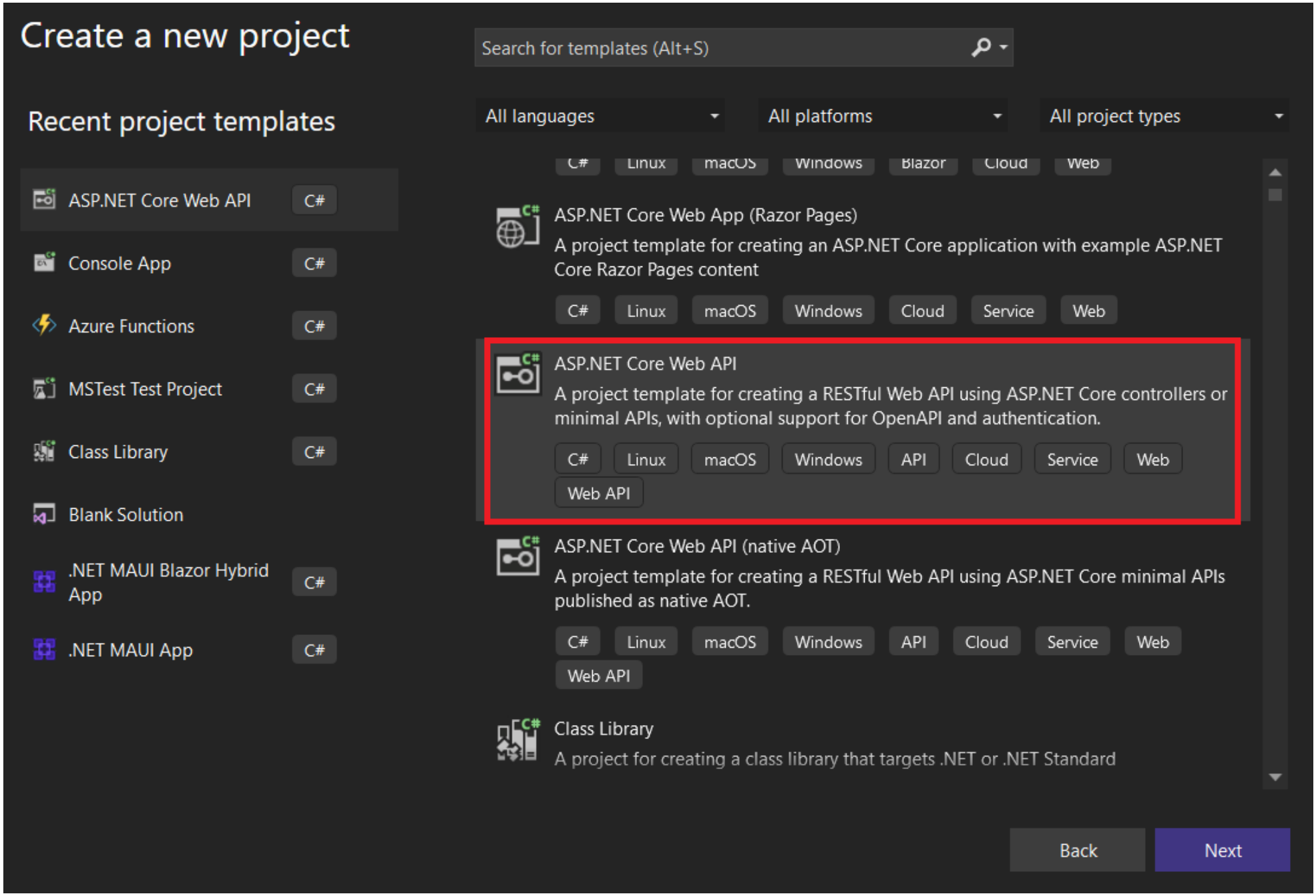
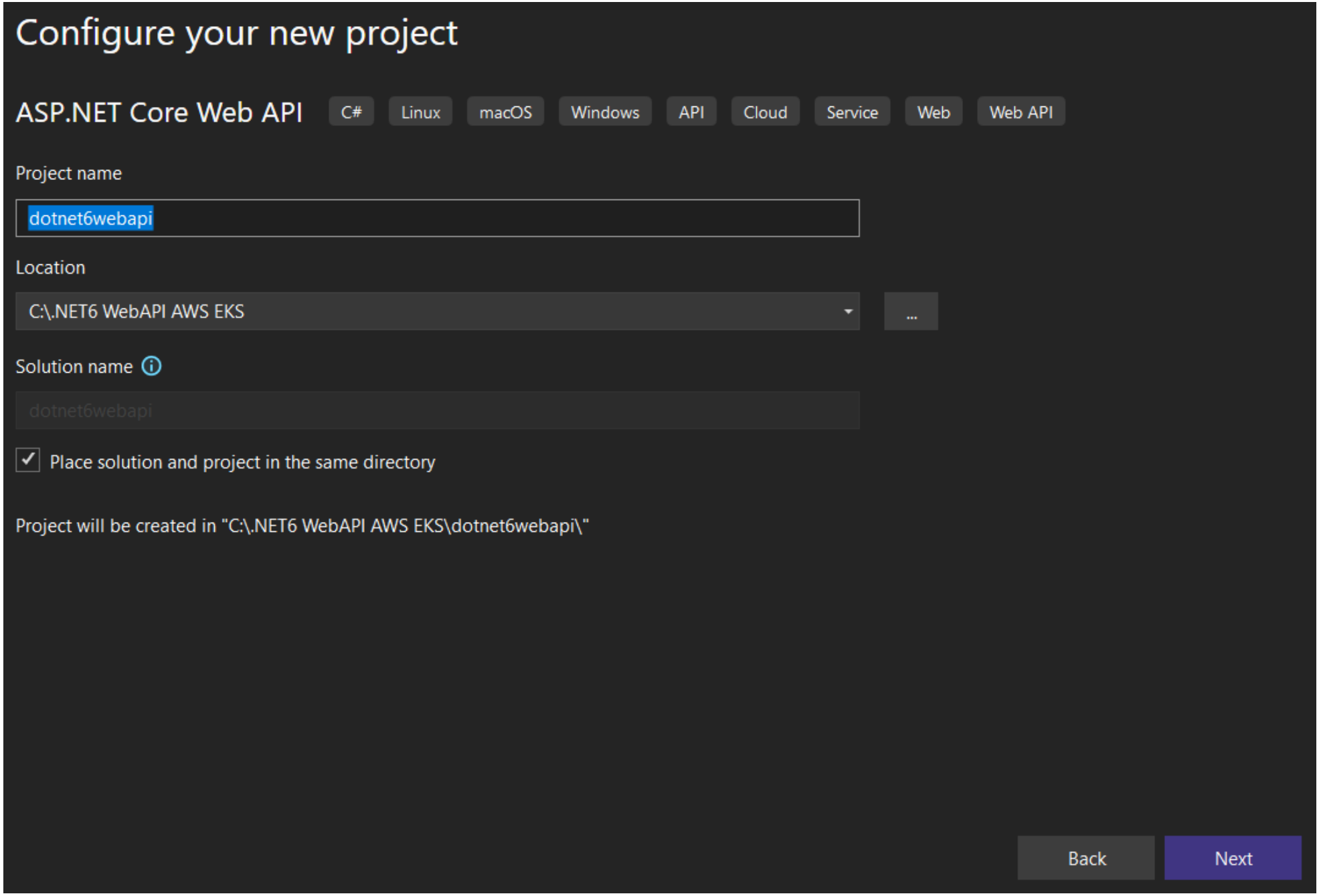
How to deploy .NET 6 WebAPI to AWS EKS cluster

0. Create a .NET 6 WebAPI with Visual Studio Community edition

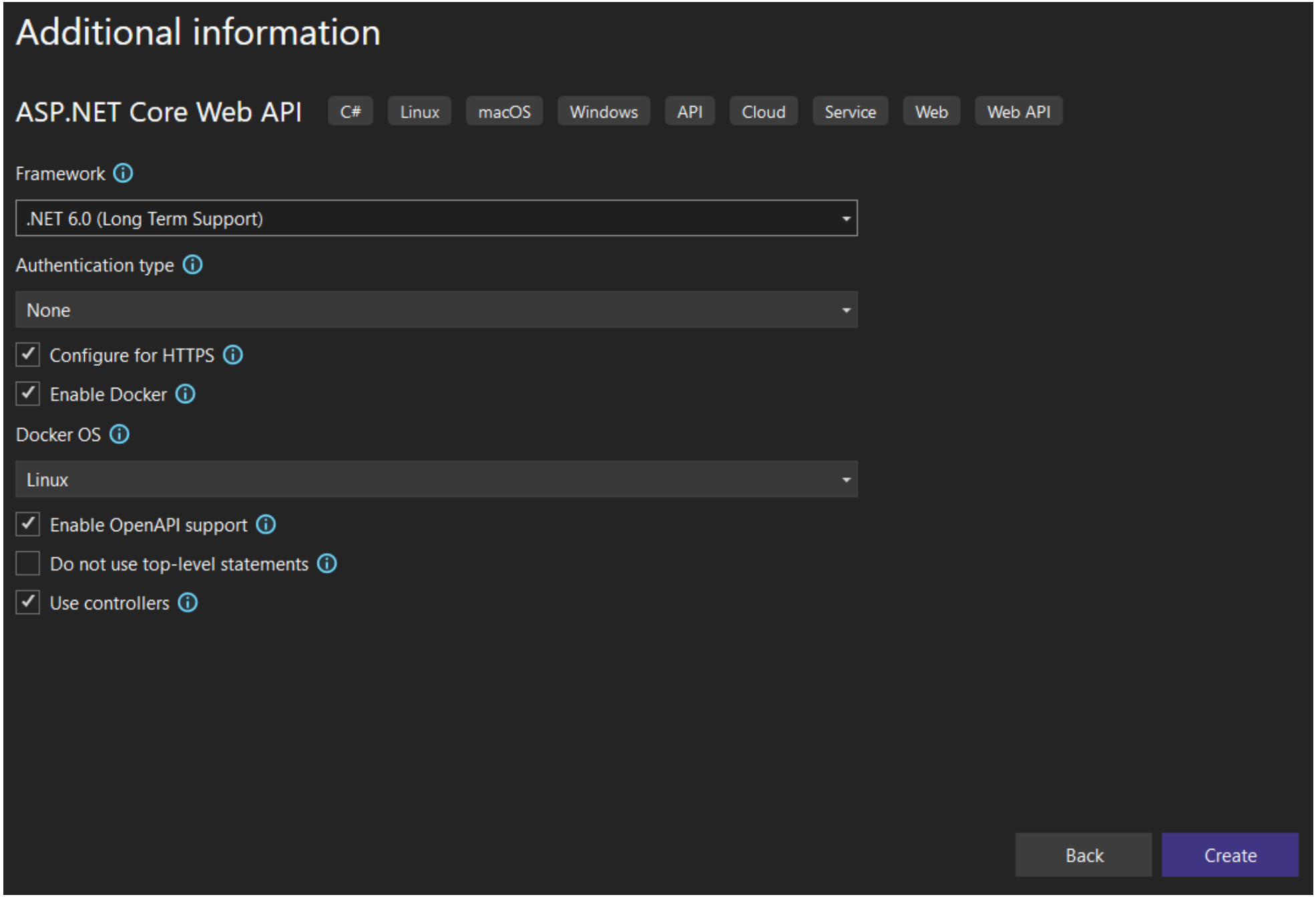

What would you like to do?
Open recent
Today
WebAPIdotNET6.sln
12/22/2023 2:33 PM
C:\.NET6 WebAPI AWS EKS\WebAPIdotNET6
WebAPIdotNET8.sln
12/22/2023 1:35 PM
C:\.NET8WebAPI\WebAPIdotNET8
WebAPIdotNET7.sln
12/22/2023 12:47 PM
C:\.NET7WebAPI\WebAPIdotNET7
Yesterday
Azure SDK for .NET.sln
12/21/2023 11:57 AM
C:\Azure_SDK_Sample1_CreateResourceGroup-main
Azure SDK for .NET.sln
12/21/2023 10:36 AM
C:\...\3D Objects\Downloads\Azure_SDK_Sample1_CreateResour...
This week
Get started
Clone a repository
Get code from an online repository like GitHub or Azure DevOps
Open a project or solution
Open a local Visual Studio project or .sln file
Open a local folder
Navigate and edit code within any folder
Create a new project
Choose a project template with code scaffolding to get started



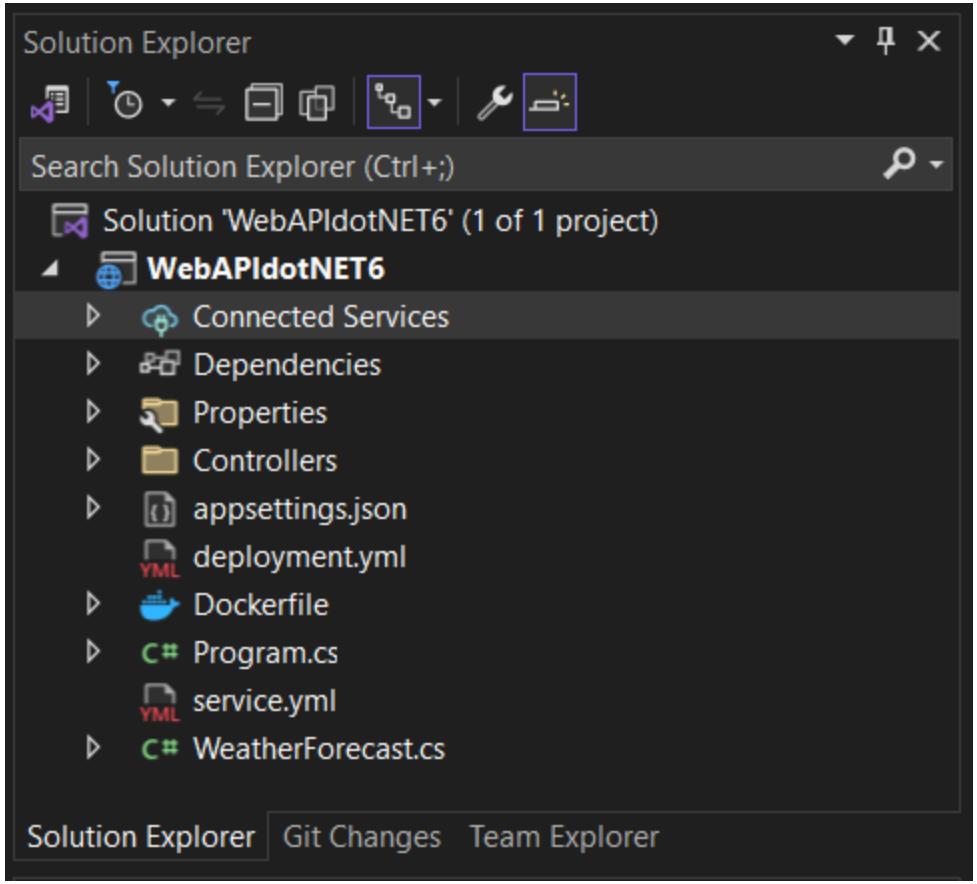
Set the project name and the location



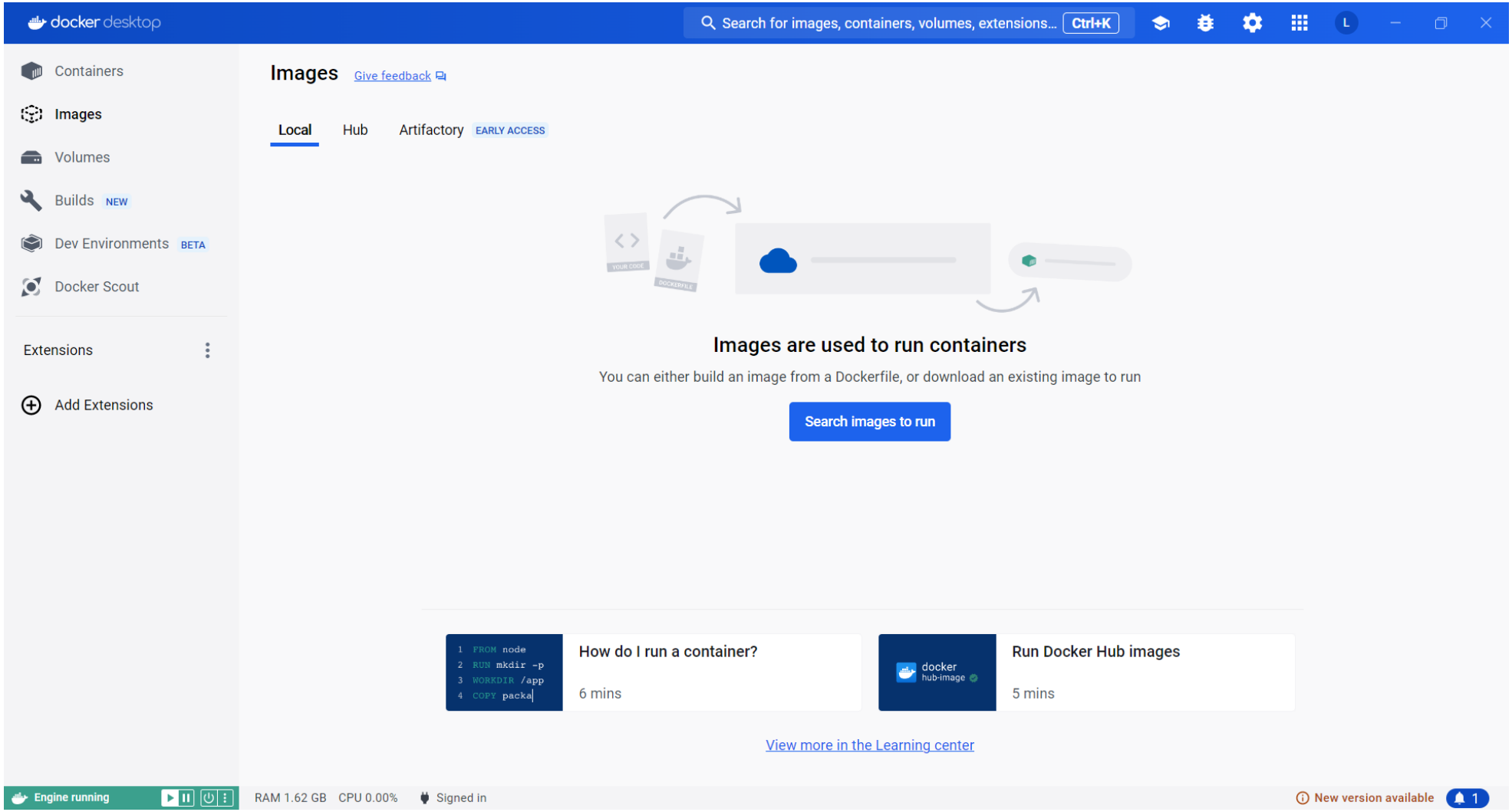
Set the project features. IMPORTANT: Enable Docker suppoort for automatically create the Dockerfile



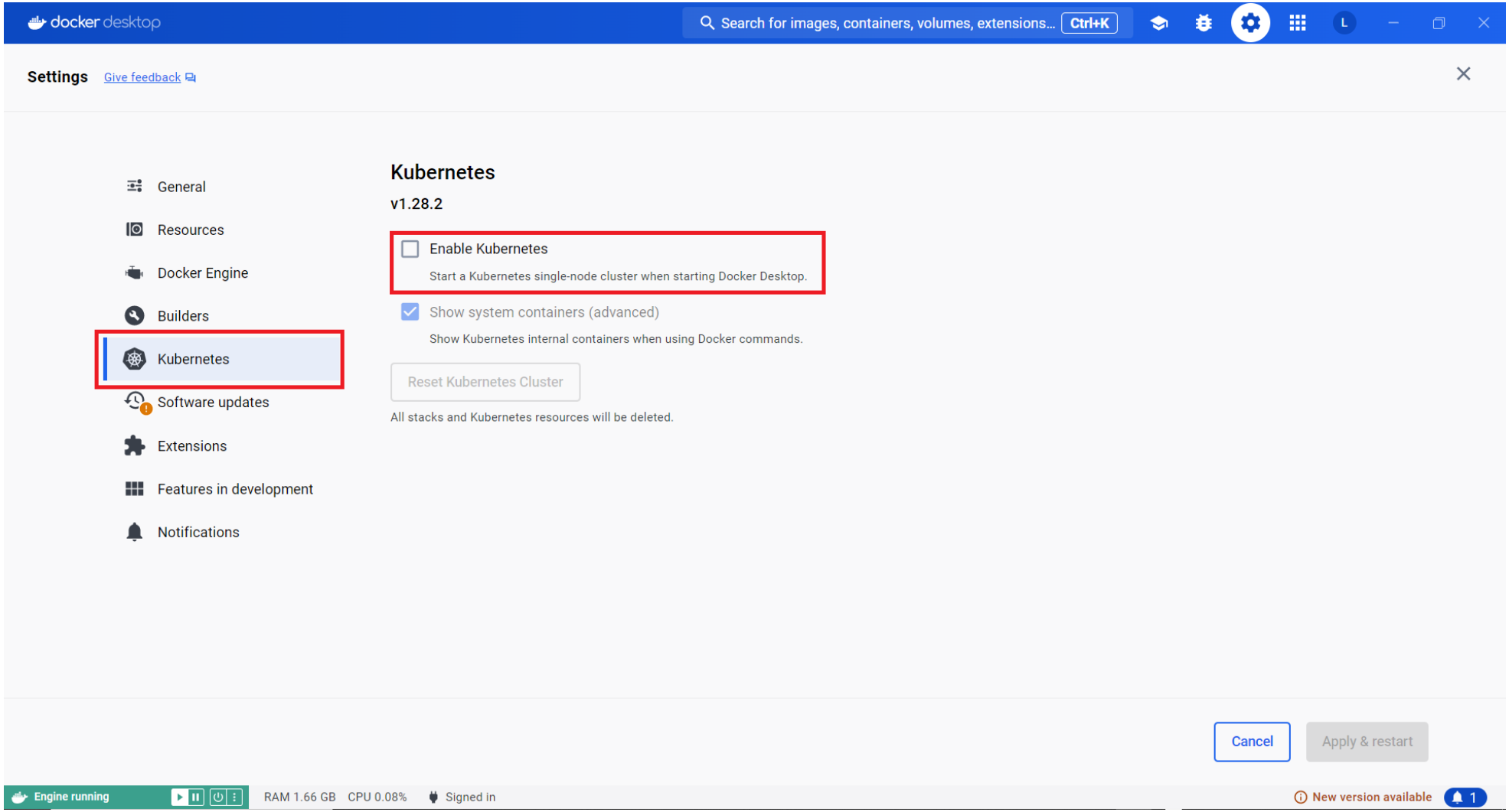
Add the kubernetes manifest files to the project: **deployment.yml** and **service.yml**

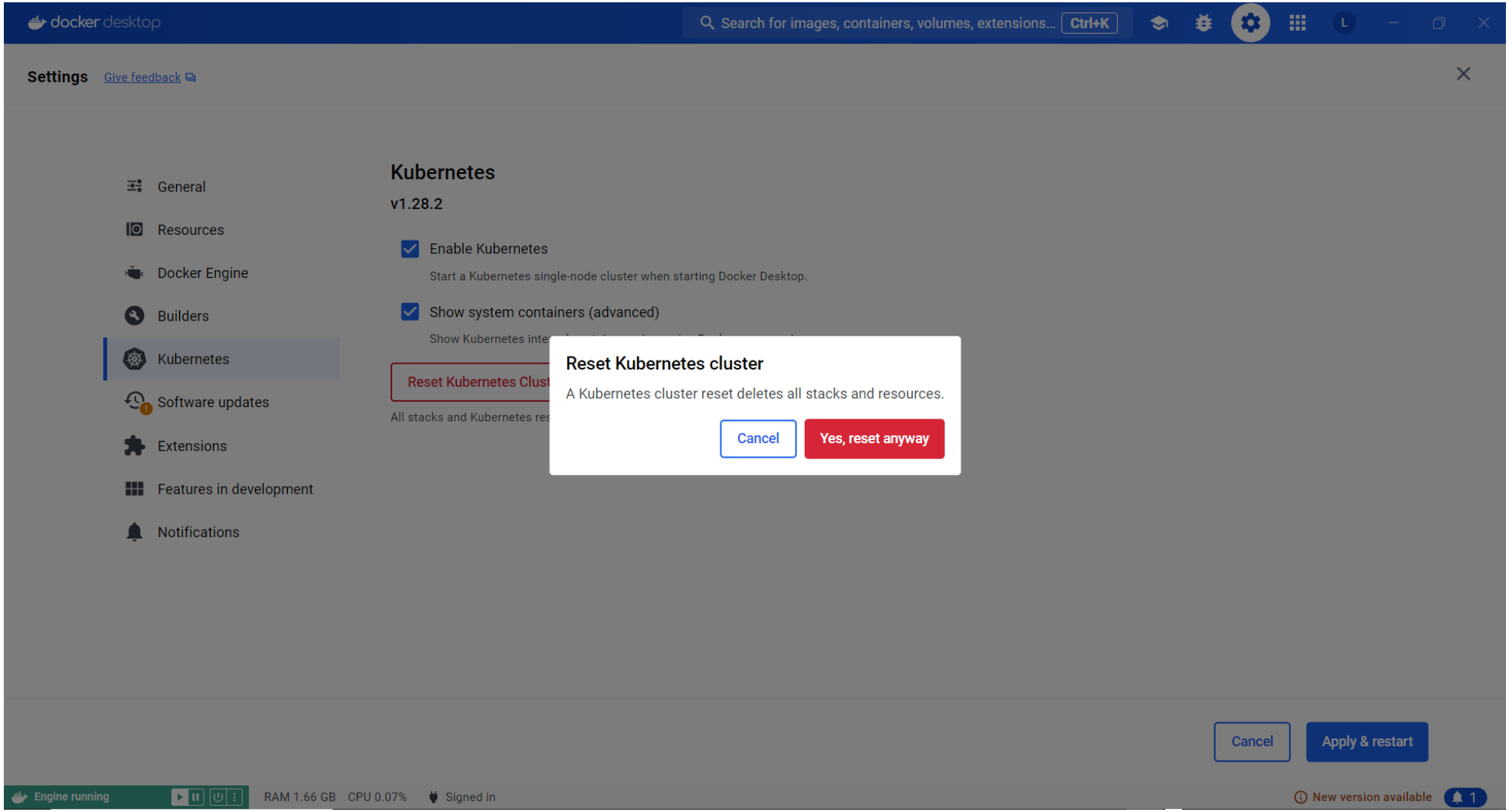


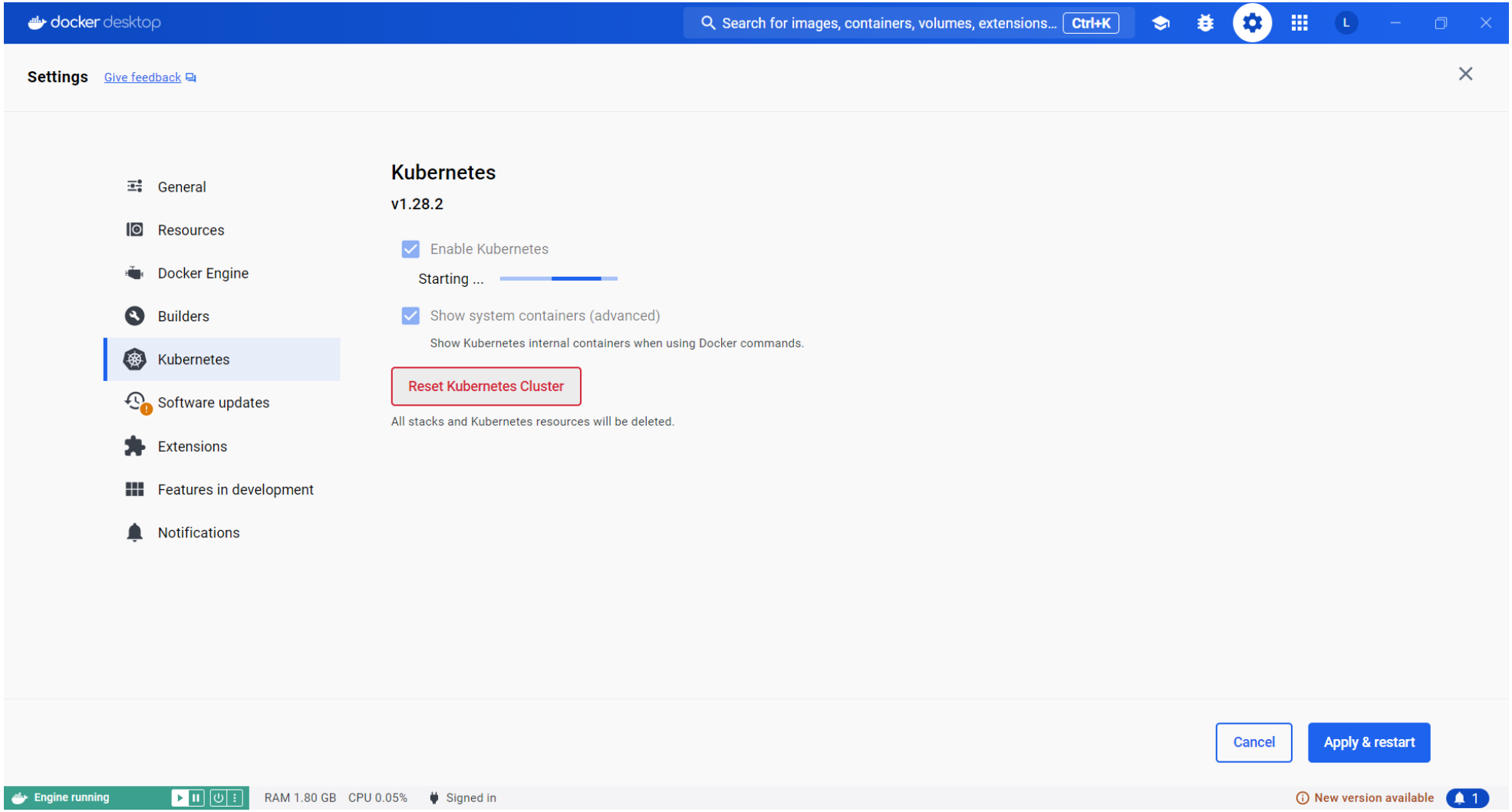
1. Run Docker Desktop



Enable Kubernetes

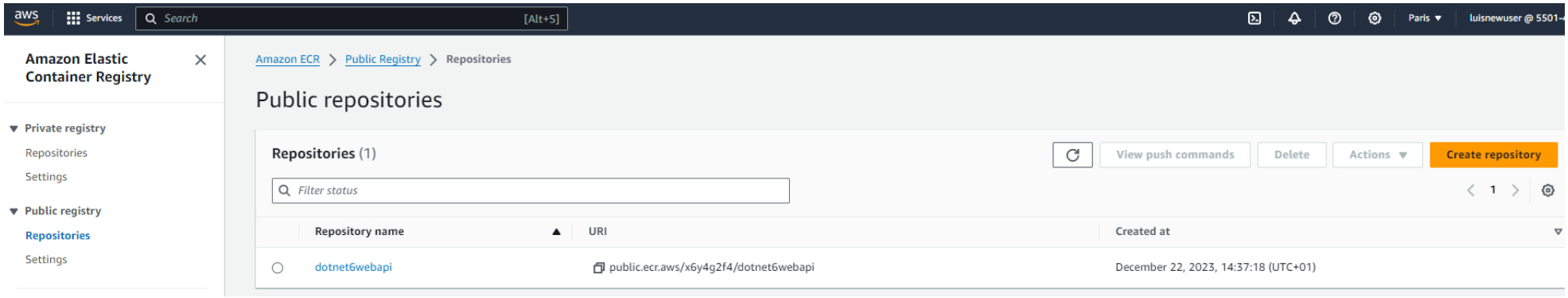

docker desktop
Search for images, containers, volumes, extensions... Ctrl+K
Settings
Give feedback
General
Resources
Docker Engine
Builders
Kubernetes
Software updates
Extensions
Features in development
Notifications
Kubernetes
v1.28.2
Enable Kubernetes
Start a Kubernetes single-node cluster when starting Docker Desktop.
Show system containers (advanced)
Show Kubernetes internal containers when using Docker commands.
Reset Kubernetes Cluster
All stacks and Kubernetes resources will be deleted.
Cancel
Apply & restart
Engine running
RAM 1.66 GB CPU 0.08%
Signed in
New version available
1


docker desktop
Search for images, containers, volumes, extensions... Ctrl+K
Settings
Give feedback
General
Resources
Docker Engine
Builders
Kubernetes
Software updates
Extensions
Features in development
Notifications
Kubernetes
v1.28.2
Enable Kubernetes
Start a Kubernetes single-node cluster when starting Docker Desktop.
Show system containers (advanced)
Reset Kubernetes Clust
All stacks and Kubernetes res
Reset Kubernetes cluster
A Kubernetes cluster reset deletes all stacks and resources.
Cancel
Yes, reset anyway
Cancel
Apply & restart
Engine running
RAM 1.66 GB CPU 0.07%
Signed in
New version available
1



2. Create .NET 6 Web API Docker image and upload to AWS ECR

Navigate to AWS ECR and create a public repo to store the .NET 6 WebAPI Docker image



Click on the created repo name and press the button **View push commands**

Push commands for dotnet6webapi

Make sure that you have the latest version of the AWS CLI and Docker installed. For more information, see Getting Started with Amazon ECR.

Use the following steps to authenticate and push an image to your repository. For additional registry authentication methods, including the Amazon ECR credential helper, see Registry Authentication.

1. Retrieve an authentication token and authenticate your Docker client to your registry.
   Use the AWS CLI:

   aws ecr-public get-login-password --region us-east-1 | docker login --username AWS --password-stdin public.ecr.aws/x6y4g2f4

   Note: If you receive an error using the AWS CLI, make sure that you have the latest version of the AWS CLI and Docker installed.
2. Build your Docker image using the following command. For information on building a Docker file from scratch see the instructions here. You can skip this step if your image is already built:

   docker build -t dotnet6webapi .
3. After the build completes, tag your image so you can push the image to this repository:

   docker tag dotnet6webapi:latest public.ecr.aws/x6y4g2f4/dotnet6webapi:latest
4. Run the following command to push this image to your newly created AWS repository:

   docker push public.ecr.aws/x6y4g2f4/dotnet6webapi:latest

Copied

Close

```
aws ecr-public get-login-password --region us-east-1 | docker login --username AWS --password-stdin public.ecr.aws/x6y4g2f4
```

```
docker build -t dotnet6webapi .
```

```
docker tag dotnet6webapi:latest public.ecr.aws/x6y4g2f4/dotnet6webapi:latest
```

```
docker push public.ecr.aws/x6y4g2f4/dotnet6webapi:latest
```

3. Create AWS EKS (Elastic Kubernetes Cluster)

```
eksctl create cluster ^
--name luiscocoenriquezdotnet6webapi-cluster ^
--version 1.25 ^
--region eu-west-3 ^
--nodegroup-name linux-nodes ^
--node-type t2.micro ^
--nodes 4
```

NOTE: if version 1.25 is not yet working in your laptop use version 1.24.

The AWS Kubernetes cluster creation takes around or more than 1 hour

NOTE: If you get an error during the cluster creation due to name is not unique, delete the cluster with the following command and input a new cluster name

```
eksctl delete cluster --region=eu-west-3 --name=luiscocoenriquezdotnet6webapi-cluster
```

4. Get and Set AWS Kubernetes Cluster context

```
kubectl config get-contexts
```

```
Developer PowerShell
Developer PowerShell
PS C:\.NET6 WebAPI AWS EKS\WebAPIdotNET6> kubectl config get-contexts
CURRENT   NAME                                                            CLUSTER                                                         AUTHINFO
                  NAMESPACE
          arn:aws:eks:eu-west-3:719220092744:cluster/education-eks-X4kWcZFq       arn:aws:eks:eu-west-3:719220092744:cluster/education-eks-X4kWcZFq   arn:aws:eks:eu-west-3:719220092744:cluster/education-e
ks-X4kWcZFq
          docker-desktop                                                  docker-desktop                                                  docker-desktop
          luisnewuser@dotnet8webapi-1974123.eu-west-3.eksctl.io           dotnet8webapi-1974123.eu-west-3.eksctl.io                       luisnewuser@dotnet8webapi-1974123.eu-west-3.eksctl.io
*         luisnewuser@luiscocoenriquezdotnet6webapi-cluster.eu-west-3.eksctl.io   luiscocoenriquezdotnet6webapi-cluster.eu-west-3.eksctl.io          luisnewuser@luiscocoenriquezdotnet6webapi-cluster.eu-w
est-3.eksctl.io
PS C:\.NET6 WebAPI AWS EKS\WebAPIdotNET6>
```

If you would like to delete a context

```
kubectl config delete-context luisnewuser@luiscocoenriquezdotnet6webapi-cluster.eu-west-3.eksctl.io
```

To select a cluster where to deploy applications, run the command:

```
kubectl config use-context luisnewuser@luiscocoenriquezdotnet6webapi-cluster.eu-west-3.eksctl.io
```

5. Verify the kubenetes parameters

We create a new namespace "dev"

```
kubectl create namespace dev
```

We verify the nodes

```
kubectl get nodes
```

We verify the namespaces

```
kubectl get ns
```

We verify the services, nodes, namespaces and pods

```
kubectl get all
```

We verify the pods

```
kubectl get pods
```

```
kubectl get all --namespace dev
```

```
kubectl get nodes --namespace dev
```

```
kubectl get ns --namespace dev
```

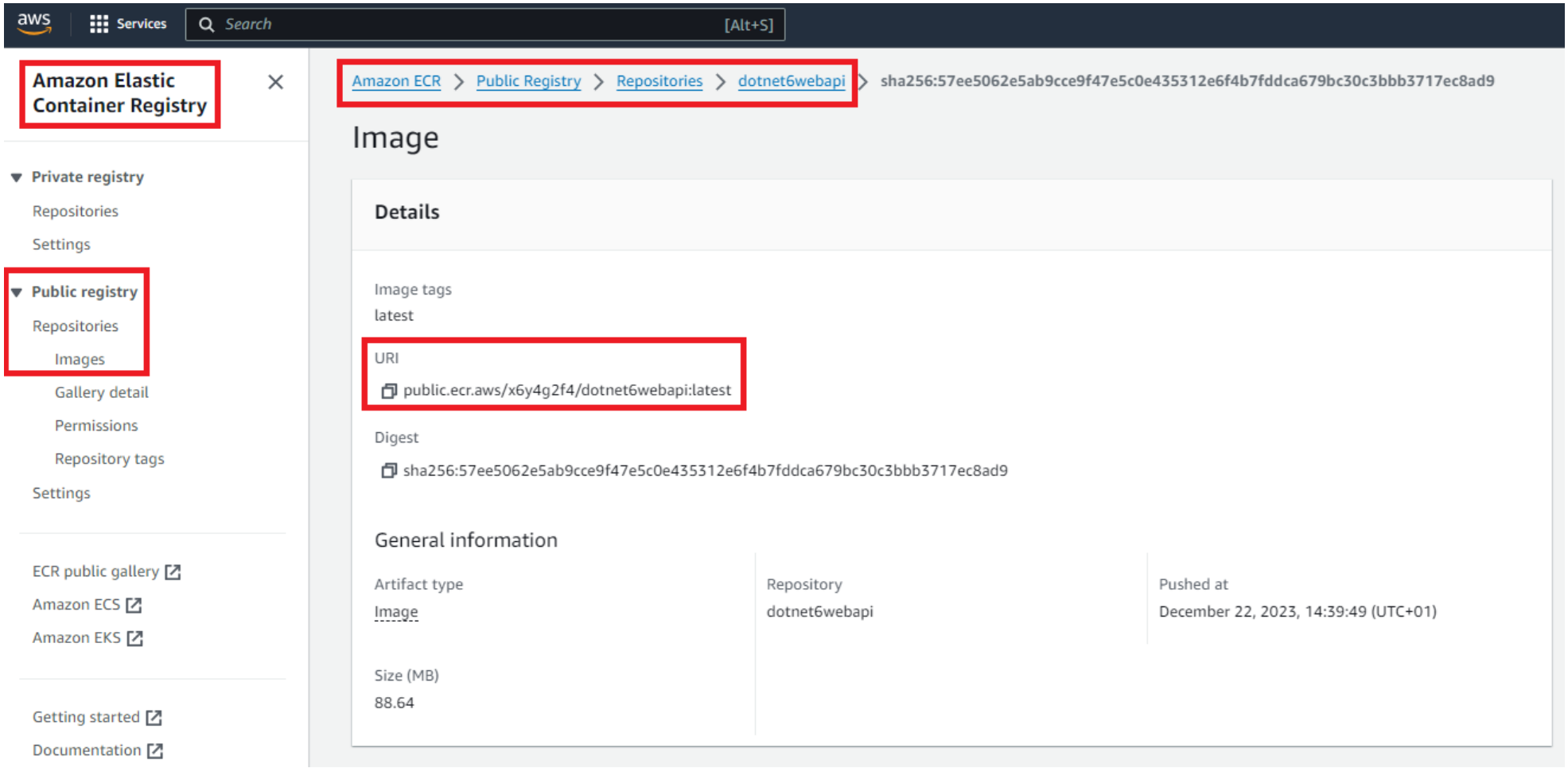
```
kubectl get pods --namespace dev
```

6. Write the deployment.yml file to deploy the Kubernetes cluster

This is the source code for the **deployment.yml** file:

IMPORTANT:

In this file do not forget to set the docker image in **AWS ECR**

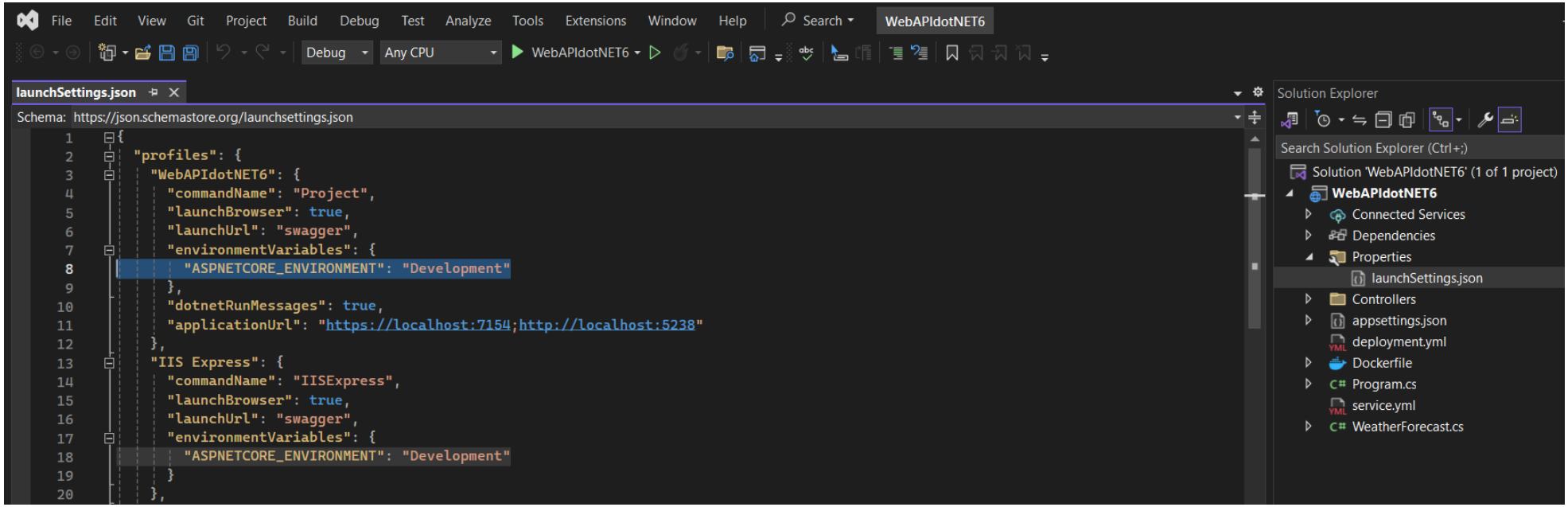


Set this image URL in the **deployment.yml** file

```yaml
spec:
      containers:
      - name: webapidotnet8
        image: public.ecr.aws/x6y4g2f4/dotnet6webapi:latest
        ports:
        - containerPort: 8080
```

Also it is very important to set the environmental variable.

Get the environmental variables values from the **launchSettings.json** file:



and set it in **deployment.yml** file

```
env:
          - name: ASPNETCORE_ENVIRONMENT
            value: Development
```

This is the **deployment.yml** source code:

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: webapidotnet6-deployment
  labels:
    app: webapidotnet6
spec:
  replicas: 1
```

```
  selector:
    matchLabels:
      app: webapidotnet6
  template:
    metadata:
      labels:
        app: webapidotnet6
    spec:
      containers:
        - name: webapidotnet6
          image: public.ecr.aws/x6y4g2f4/dotnet6webapi:latest
          imagePullPolicy: IfNotPresent
          ports:
            - containerPort: 80
          env:
            - name: ASPNETCORE_ENVIRONMENT
              value: Development
          resources:
            requests:
              memory: "64Mi"
              cpu: "250m"
            limits:
              memory: "128Mi"
              cpu: "500m"
```

IMPORTANT NOTE: take in consideration if you include in your application a connection string to a database the corresponding environmental variable also should be set (see this example)

```
env:
            - name: ASPNETCORE_ENVIRONMENT
              value: Development
            - name: ConnectionStrings__FleetManagementDb
              valueFrom:
                  configMapKeyRef:
```

```
                    name: mongo-configmap
                    key: connection_string
```

mongo-configmap.yml

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: mongo-configmap
data:
  #connection_string: mongodb://username:password@mongo-service:27017
  #connection_string: mongodb://mongodb-service:27017
  #connection_string: mongodb://mongod-0.mongodb-service.default.svc.cluster.local:27017,mongod-1.mongodb-service.default.svc.clu
  connection_string: mongodb://mongod-0.mongodb-service.dev.svc.cluster.local:27017,mongod-1.mongodb-service.dev.svc.cluster.loca
  #connection_string: mongodb://mongod-0.mongodb-service.development.svc.cluster.local:27017,mongod-1.mongodb-service.development
```

7. This is the service.yml file to deploy the Kubernetes cluster

*service.yml

```
apiVersion: v1
kind: Service
metadata:
  name: webapidotnet6-service
spec:
  type: LoadBalancer
  selector:
    app: webapidotnet6
  ports:
    - protocol: TCP
```

```
    port: 80
    targetPort: 80
```

8. Deploy the kubernetes manifest files (deployment.yml and service.yml)

```
kubectl apply -f deployment.yml --namespace dev
```

```
kubectl apply -f service.yml --namespace dev
```

9. Verify the Web API endpoint

ac813aba353aa41e7ba83b95c27

Not secure | ac813aba353aa41e7ba83b95c27e02ed-708937805.eu-west-3.elb.amazonaws.com/weatherforecast


```
[
    {
        "date": "2023-12-23T16:42:54.8228708+00:00",
        "temperatureC": -14,
        "temperatureF": 7,
        "summary": "Balmy"
    },
    {
        "date": "2023-12-24T16:42:54.8228762+00:00",
        "temperatureC": -19,
        "temperatureF": -2,
        "summary": "Balmy"
    },
    {
        "date": "2023-12-25T16:42:54.8228765+00:00",
        "temperatureC": 29,
        "temperatureF": 84,
        "summary": "Mild"
    },
    {
        "date": "2023-12-26T16:42:54.8228767+00:00",
        "temperatureC": 12,
        "temperatureF": 53,
        "summary": "Warm"
    },
    {
        "date": "2023-12-27T16:42:54.8228768+00:00",
        "temperatureC": -7,
        "temperatureF": 20,
        "summary": "Hot"
    }
]
```