This section provides a detailed overview of the infrastructure, application deployment process, CI/CD pipeline, and monitoring systems used in the project.

Table of Contents

1. Summary

2. Architecture

3. User guideline

4. The homepage of the application

5. Component Description

1. Summary

This document introduces the overall architecture, infrastructure setup, application deployment strategy, CI/CD workflow, and monitoring system used in this project.

The goal of this system is to provide a scalable, maintainable, and observable platform for deploying modern applications. It leverages cloud-native tools and best practices to ensure reliability and automation.

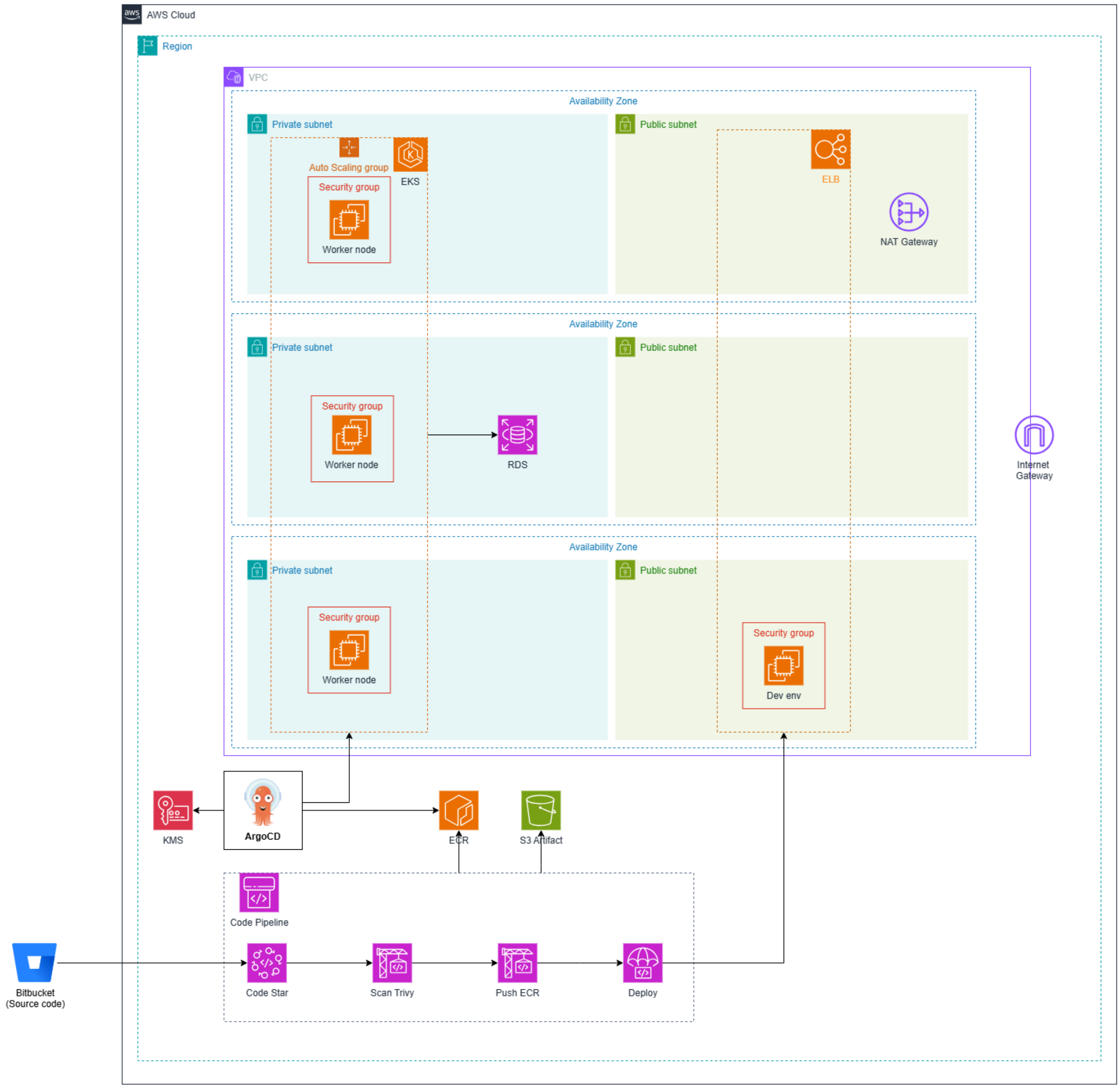
Key components include:

- **Infrastructure**: Provisioned using Infrastructure as Code (e.g., Terraform, Ansible), hosted on [e.g., AWS/GCP/Azure].
- **Application Deployment**: Containerized using Docker and orchestrated with Kubernetes.
- **CI/CD Pipeline**: Automates scan, and deployment using [AWS CodePipeline, ArgoCD].
- **Monitoring System**: Provides observability through tools likes CloudWatch

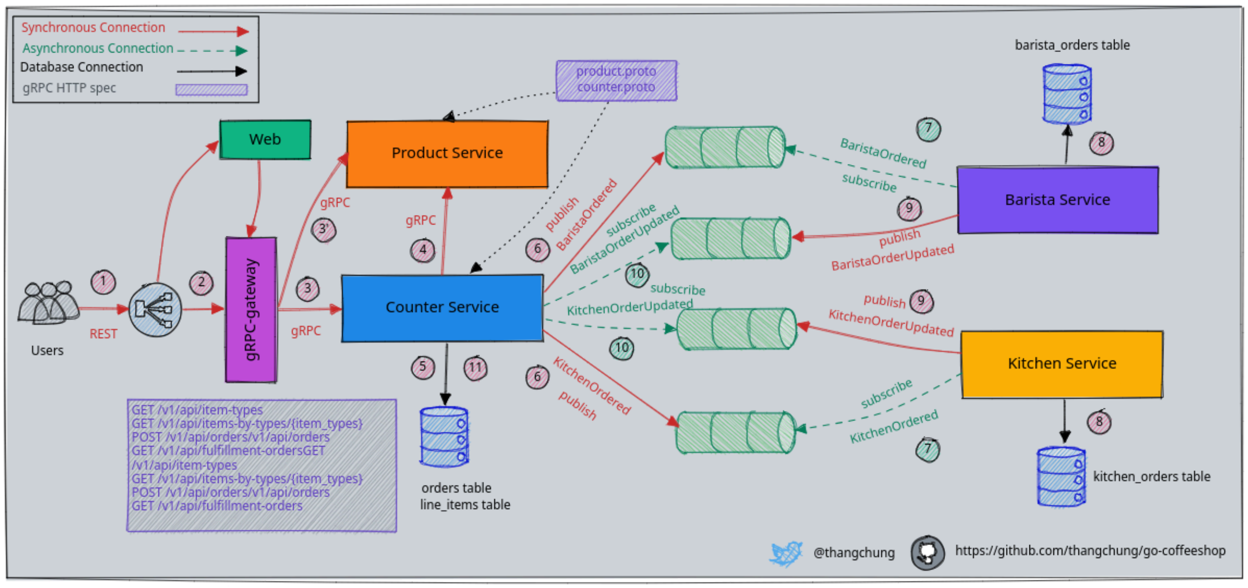
This guide is intended for developers, DevOps engineers, and platform engineers who need to understand, deploy, or maintain the system.

2. Architecture

2.1. AWS Infrastructure Architecture



2.2. Application Services Architecture



2.3 Monitoring & Observability Architecture

3. User guideline

3.1. Structure Overview

The root directory contains all necessary components to provision infrastructure, deploy applications, configure monitoring, manage secrets

Here is an overview of each top-level folder and file:

```
├── argo
├── coffeeshop
├── docker-compose.yaml
├── docs
├── eks-setup
├── infrastructure
├── monitor-setup
├── README.md
└── secrets
```

3.2. `infrastructure/`

This directory contains all Terraform files to provision the necessary resources. It includes both custom and public modules.

```
├── backend.tf
├── common_variables.tf
├── locals.tf
├── main.tf
├── provider.tf
├── README.md
├── trainee.tfvars
├── variables.tf
├── modules
│   ├── ci_cd_pipeline
│   │   ├── main.tf
│   │   ├── outputs.tf
│   │   └── variables.tf
│   ├── eks
│   │   ├── main.tf
│   │   ├── outputs.tf
│   │   └── vairables.tf
│   ├── eks_iam
│   │   ├── main.tf
│   │   ├── outputs.tf
│   │   └── variables.tf
│   ├── elasticache
│   │   ├── main.tf
│   │   ├── outputs.tf
│   │   └── variables.tf
│   ├── rds
│   │   ├── main.tf
│   │   ├── outputs.tf
│   │   └── variables.tf
│   └── vpc
│       ├── main.tf
│       ├── outputs.tf
│       └── vairables.tf
└──
```

There are two main workspaces:

- dev: EC2, VPC
- prod: EKS, RDS, VPC, EKS, KMS, CICD

Preparation:

- terraform
- awscli
- eksctl
- Create a CodeStar connection to your Git repo and get `codestar_connection_arn`

Before creating infrastructure, update the `trainee.tfvars` file with your information:

```
region = "<YOUR-REGION>"

name = "<YOUR-PROJECT-NAME>"

availability_zones = [
  "<YOUR-AVAILABILITY-ZONE-1>",
  "<YOUR-AVAILABILITY-ZONE-2>"
]

db_name = "<YOUR-DB-NAME>"

enable_nat_gateway = true

github_owner = "<YOUR-GITHUB-OWNER>"

github_repo = "<YOUR-GITHUB-REPO>"

codestar_connection_arn = "<YOUR-CODESTAR-CONNECTION-ARN>"

account_id = "<YOUR-AWS-ACCOUNT-ID>"

services = {
  product = {
    image = "baominh/go-coffeeshop-product:latest"
  },
  counter = {
    image = "baominh/go-coffeeshop-counter:latest"
  },
  barista = {
    image = "baominh/go-coffeeshop-barista:latest"
  },
  kitchen = {
    image = "baominh/go-coffeeshop-kitchen:latest"
  },
  proxy = {
    image = "baominh/go-coffeeshop-proxy:latest"
  },
  web = {
    image = "baominh/go-coffeeshop-web:latest"
  },
  vulnerables = {
    image = "vulnerables/web-dvwa:latest"
  },
  rabbitmq = {
    image = "rabbitmq:3.11-management-alpine"
  }
}
```

Update the S3 backend in `backend.tf`:

```
terraform {
  backend "s3" {
    bucket = "<YOUR-BUCKETNAME>"
    key    = "terraform.tfstate"
    region = "<YOUR-REGION>"
  }
}
```

Create Terraform workspaces:

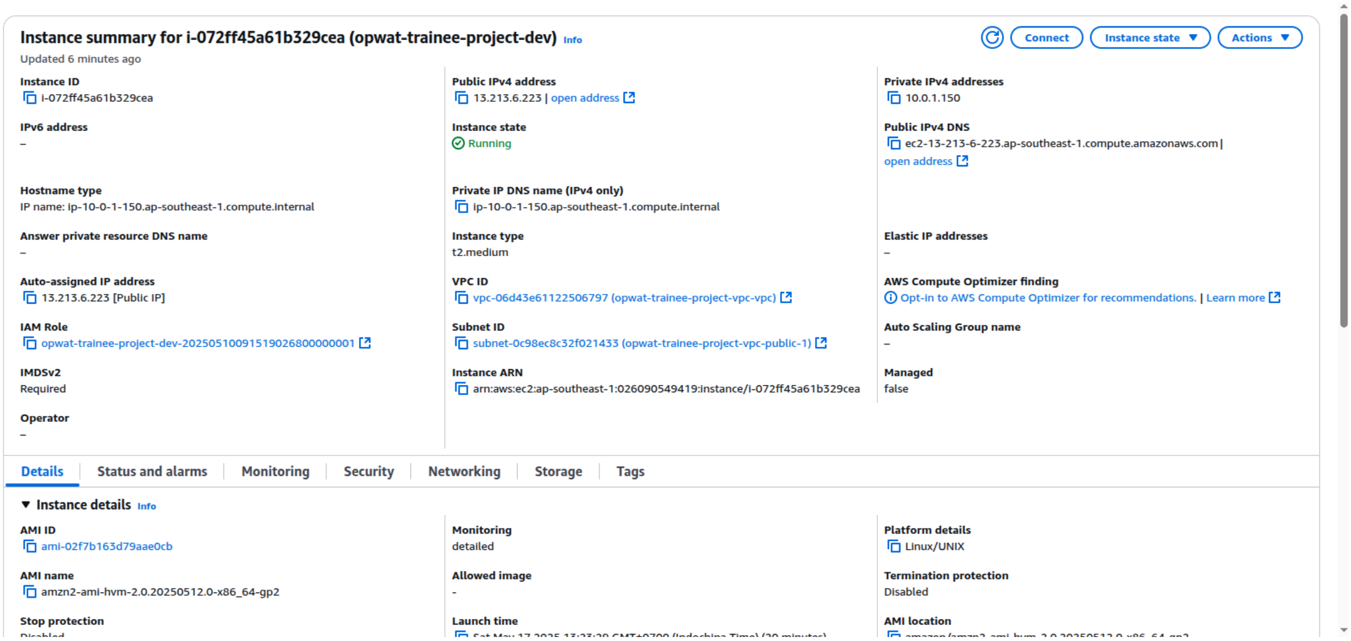
```
terraform workspace create prod
terraform workspace create dev
```

Switch to `prod` workspace and create resources:

```
# Switch to prod workspace
terraform workspace select dev

terraform init
terraform apply -varfile=trainee.tfvars

terraform workspace select prod

terraform init
terraform apply -varfile=trainee.tfvars
```

After creation, you can check your infrastructure:

Dev environment



Instance summary for i-072ff45a61b329cea (opwat-trainee-project-dev) Info

Updated 6 minutes ago

Instance ID
i-072ff45a61b329cea

IPv6 address
–

Hostname type
IP name: ip-10-0-1-150.ap-southeast-1.compute.internal

Answer private resource DNS name
–

Auto-assigned IP address
13.213.6.223 [Public IP]

IAM Role
opwat-trainee-project-dev-20250510091519026800000001

IMDSv2
Required

Operator
–

Public IPv4 address
13.213.6.223 | open address

Instance state
Running

Private IP DNS name (IPv4 only)
ip-10-0-1-150.ap-southeast-1.compute.internal

Instance type
t2.medium

VPC ID
vpc-06d43e61122506797 (opwat-trainee-project-vpc-vpc)

Subnet ID
subnet-0c98ec8c32f021433 (opwat-trainee-project-vpc-public-1)

Instance ARN
arn:aws:ec2:ap-southeast-1:026090549419:instance/i-072ff45a61b329cea

Private IPv4 addresses
10.0.1.150

Public IPv4 DNS
ec2-13-213-6-223.ap-southeast-1.compute.amazonaws.com | open address

Elastic IP addresses
–

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more

Auto Scaling Group name
–

Managed
false

▼ Instance details Info

AMI ID
ami-02f7b163d79aae0cb

AMI name
amzn2-ami-hvm-2.0.20250512.0-x86_64-gp2

Stop protection
Disabled

Monitoring
detailed

Allowed image
-

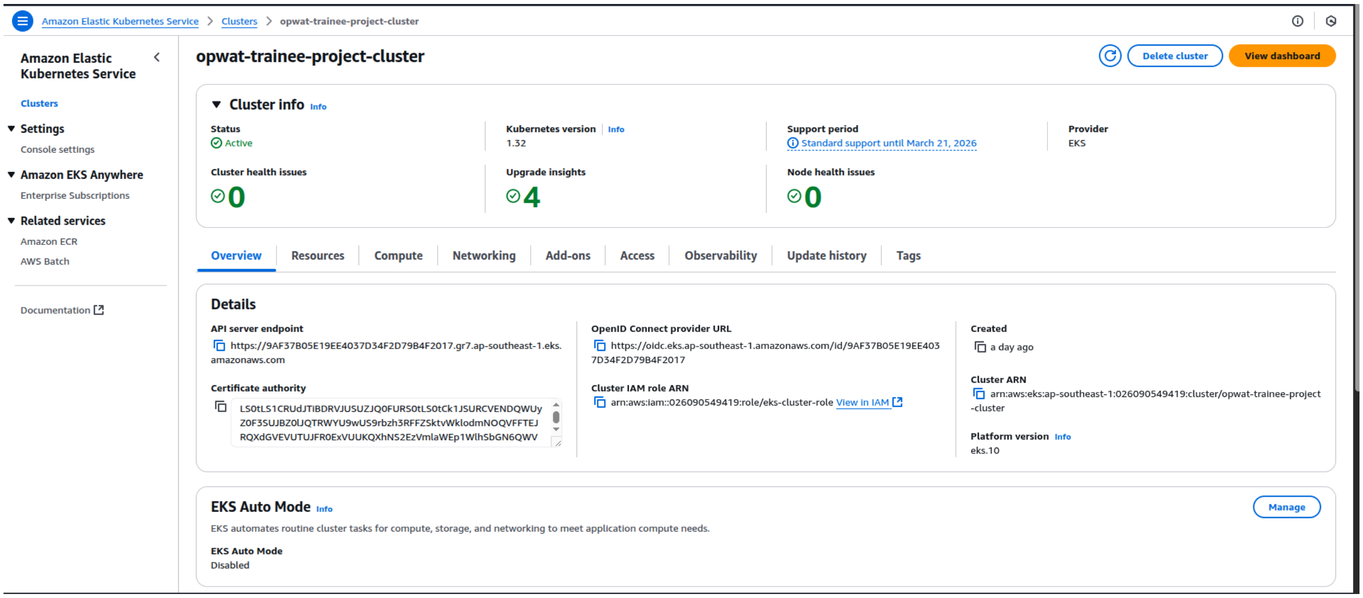
Launch time
Sat May 17 2025 13:23:39 GMT+0700 (Indochina Time) (20 minutes)

Platform details
Linux/UNIX

Termination protection
Disabled

AMI location
amazon/amzn2-ami-hvm-2.0.20250512.0-x86_64-gp2

Prod environment



opwat-trainee-project-cluster

▼ Cluster info Info

Status
Active

Kubernetes version | Info
1.32

Support period
Standard support until March 21, 2026

Provider
EKS

Cluster health issues
0

Upgrade insights
4

Node health issues
0

Details

API server endpoint
https://9AF37B05E19EE4037D34F2D79B4F2017.gr7.ap-southeast-1.eks.amazonaws.com

Certificate authority
LS0tLS1CRUdJTiBDRVJUSUZJQ0FURS0tLS0tCk1JSURCVENDQWUy
Z0F3SUJBZ0lJQTRWYU9wUS9rbzh3RFFZSktvWklodmNOQVFFTEJ
RQXdGVEVUTUJFR0ExVUUKQXhNS2EzVmlaWEp1WlhSbGN6QWV

OpenID Connect provider URL
https://oidc.eks.ap-southeast-1.amazonaws.com/id/9AF37B05E19EE4037D34F2D79B4F2017

Cluster IAM role ARN
arn:aws:iam::026090549419:role/eks-cluster-role View in IAM

Created
a day ago

Cluster ARN
arn:aws:eks:ap-southeast-1:026090549419:cluster/opwat-trainee-project-cluster

Platform version Info
eks.10

EKS Auto Mode Info

EKS automates routine cluster tasks for compute, storage, and networking to meet application compute needs.

EKS Auto Mode
Disabled

3.3. Get kube config file

```
eksctl utils write-kubeconfig --cluster opwat-trainee-project-cluster --
region ap-southeast-1
```

3.4. Create AWS cred use for SOPS

```
kubectl create secret generic argocd-aws-credentials \
  --from-literal=accesskey=XXXX \
  --from-literal=secretkey=XXXX \
  -n default
```

3.5. ArgoCD

3.5.1 Install ArgoCD

We will install ArgoCD to manage all resources using GitOps. In the `argo/` directory, run:

```
helm repo add argo https://argoproj.github.io/argo-helm
helm install argocd argo/argo-cd -f values.yaml -n default
```

Forward port to local:

```
kubectl port-forward svc/argocd-server 8080:443
```

Get ArgoCD initial password:
username: admin
password:

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

Change default password:

```
argocd login localhost:8080 --username admin --password <OLD_PASSWORD>
argocd account update-password
```

3.5.2 Create argoCD application

Ở phần này chúng ta quan tâm với 2 folder chính là `argo/` và `iam-role/`

```
.
├── application
│   ├── app-project.yaml
│   ├── aws-cloudwatch-metrics.yaml
│   ├── aws-load-balancer-controller.yaml
│   ├── coffeeshop-barista-app.yaml
│   ├── coffeeshop-counter-app.yaml
│   ├── coffeeshop-kitchen-app.yaml
│   ├── coffeeshop-product-app.yaml
│   ├── coffeeshop-proxy-app.yaml
│   ├── coffeeshop-rabbitmq.yaml
│   ├── coffeeshop-secret.yaml
│   ├── coffeeshop-web-app.yaml
│   ├── external-dns.yaml
│   └── image-updater.yaml
├── README.md
└── values.yaml
...
.
├── aws-load-balancer-controller
│   ├── iam-policy.json
│   └── trust-policy.json
├── cloudwatch-agnet
│   ├── iam-policy.json
│   ├── README.md
│   └── trust-policy.json
├── external-dns
│   ├── iam-policy.json
│   ├── README.md
│   └── trust-policy.json
└── image-updater
    ├── iam-policy.json
    └── trust-policy.json
```

Step to deploy 1 application:

- Create policy, trust-policy
- Create Role
- Add policy to created role
- Add role arn to serviceaccount

argocd bitbucket key

```
sops -e argocd-bitbucket-key.yaml > argocd-bitbucket-key.enc.yaml
```

aws-load-balancer-controller

- Create role in `iam-role/`

- Make sure you update OIDC of your cluster

```
eksctl utils associate-iam-oidc-provider \
  --region ap-southeast-1 \
  --cluster opwat-trainee-project-cluster \
  --approve
```

```
aws iam create-policy \
    --policy-name AWSLoadBalancerControllerIAMPolicy \
    --policy-document file://iam-policy.json
```

```
aws iam attach-role-policy \
  --policy-arn
arn:aws:iam::026090549419:policy/AWSLoadBalancerControllerIAMPolicy \
  --role-name AmazonEKSLoadBalancerControllerRole
```

```
aws iam update-assume-role-policy \
  --role-name AmazonEKSLoadBalancerControllerRole \
  --policy-document file://trust-policy.json
```

```
k apply -g aws-load-balancer-controller.yaml
```

CloudWatch agent

```
aws iam create-policy \
  --policy-name CloudwatchAgentPolicy \
  --policy-document file://iam-policy.json
```

```
aws iam create-role \
  --role-name CloudwatchAgentRole \
  --assume-role-policy-document file://trust-policy.json
```

```
aws iam attach-role-policy \
  --role-name CloudwatchAgentRole \
  --policy-arn arn:aws:iam::026090549419:policy/CloudwatchAgentPolicy
```

external-dns

```
aws iam create-policy \
  --policy-name ExternalDNSPolicy \
  --policy-document file://iam-policy.json
```

```
aws iam create-role \
  --role-name ExternalDNSRole \
  --assume-role-policy-document file://trust-policy.json
```

```
aws iam update-assume-role-policy \
  --role-name ExternalDNSRole \
  --policy-document file://trust-policy.json
```

```
aws iam attach-role-policy \
  --role-name ExternalDNSRole \
  --policy-arn arn:aws:iam::026090549419:policy/ExternalDNSPolicy
```

```
k apply -g external-dns.yaml
```

argocd image updater

```
aws iam create-policy \
  --policy-name argoCDImageUpdaterPolicy \
  --policy-document file://iam-policy.json
```

```
aws iam create-role \
  --role-name argoCDImageUpdaterRole \
  --assume-role-policy-document file://trust-policy.json
```

```
aws iam attach-role-policy \
  --role-name argoCDImageUpdaterRole \
  --policy-arn arn:aws:iam::026090549419:policy/argoCDImageUpdaterPolicy
```

```
k apply -g image-updater.yaml
```

sops

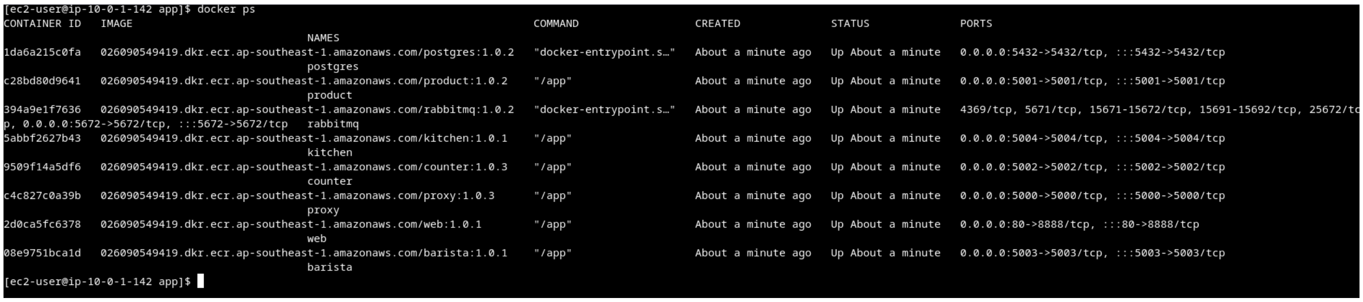
```
k apply -g coffeeshop-secret.yaml
```

application

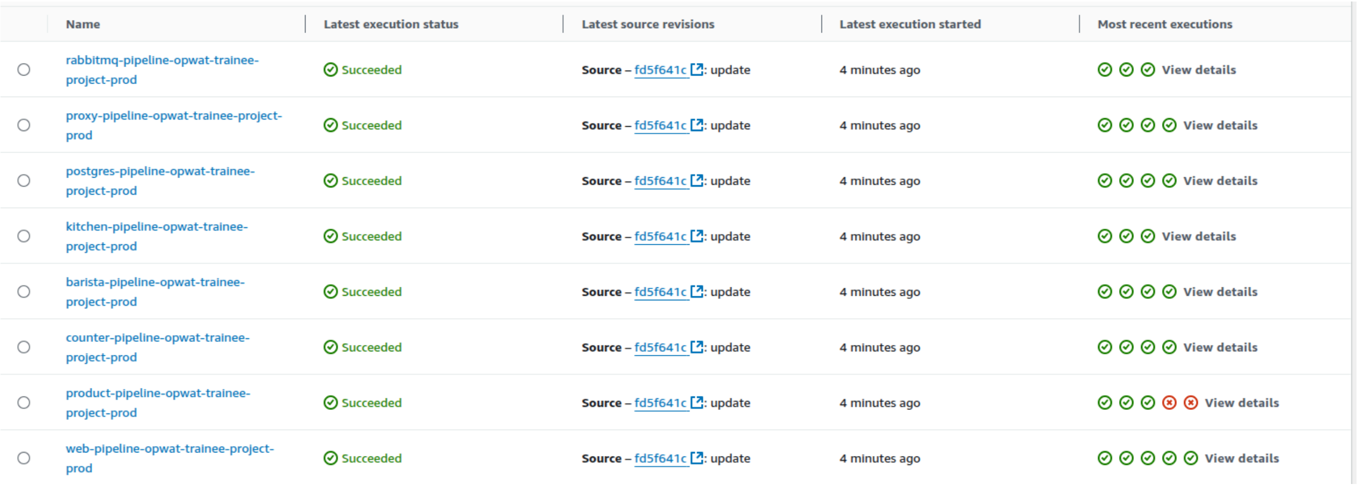
```
kubectl apply -f app-project.yaml \
  -f aws-load-balancer-controller.yaml \
  -f coffeeshop-barista-app.yaml \
  -f coffeeshop-counter-app.yaml \
  -f coffeeshop-kitchen-app.yaml \
  -f coffeeshop-product-app.yaml \
  -f coffeeshop-proxy-app.yaml \
  -f coffeeshop-rabbitmq.yaml \
  -f coffeeshop-secret.yaml \
  -f coffeeshop-web-app.yaml \
  -f external-dns.yaml \
  -f image-updater.yaml
```

3.6 Result

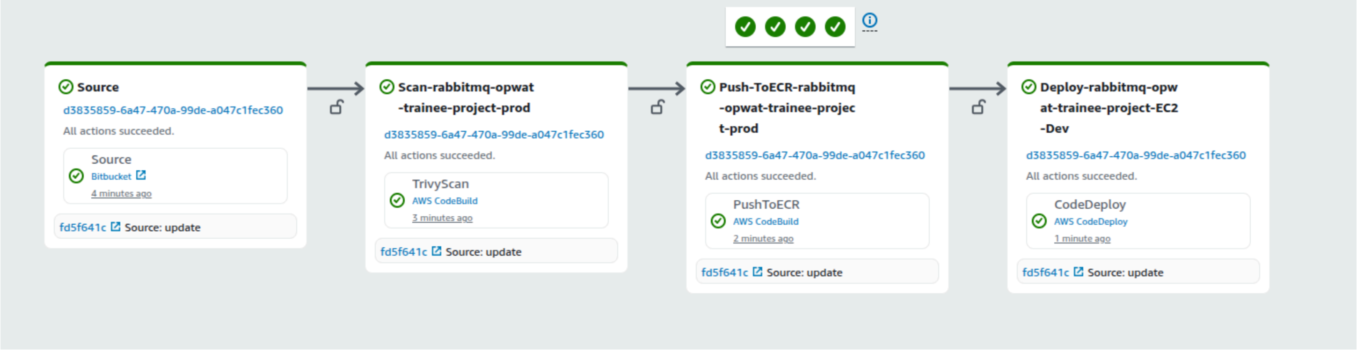
You can verify after the pipelines complete



```
[ec2-user@ip-10-0-1-142 app]$ docker ps
CONTAINER ID   IMAGE                                                              COMMAND                  CREATED              STATUS              PORTS
                                               NAMES
1da6a215c0fa   026090549419.dkr.ecr.ap-southeast-1.amazonaws.com/postgres:1.0.2   "docker-entrypoint.s…"   About a minute ago   Up About a minute   0.0.0.0:5432->5432/tcp, :::5432->5432/tcp
                                               postgres
c28bd80d9641   026090549419.dkr.ecr.ap-southeast-1.amazonaws.com/product:1.0.2    "/app"                   About a minute ago   Up About a minute   0.0.0.0:5001->5001/tcp, :::5001->5001/tcp
                                               product
394a9e1f7636   026090549419.dkr.ecr.ap-southeast-1.amazonaws.com/rabbitmq:1.0.2   "docker-entrypoint.s…"   About a minute ago   Up About a minute   4369/tcp, 5671/tcp, 15671-15672/tcp, 15691-15692/tcp, 25672/tc
p, 0.0.0.0:5672->5672/tcp, :::5672->5672/tcp   rabbitmq
5abbf2627b43   026090549419.dkr.ecr.ap-southeast-1.amazonaws.com/kitchen:1.0.1    "/app"                   About a minute ago   Up About a minute   0.0.0.0:5004->5004/tcp, :::5004->5004/tcp
                                               kitchen
9509f14a5df6   026090549419.dkr.ecr.ap-southeast-1.amazonaws.com/counter:1.0.3    "/app"                   About a minute ago   Up About a minute   0.0.0.0:5002->5002/tcp, :::5002->5002/tcp
                                               counter
c4c827c0a39b   026090549419.dkr.ecr.ap-southeast-1.amazonaws.com/proxy:1.0.3      "/app"                   About a minute ago   Up About a minute   0.0.0.0:5000->5000/tcp, :::5000->5000/tcp
                                               proxy
2d0ca5fc6378   026090549419.dkr.ecr.ap-southeast-1.amazonaws.com/web:1.0.1        "/app"                   About a minute ago   Up About a minute   0.0.0.0:80->8888/tcp, :::80->8888/tcp
                                               web
08e9751bca1d   026090549419.dkr.ecr.ap-southeast-1.amazonaws.com/barista:1.0.1    "/app"                   About a minute ago   Up About a minute   0.0.0.0:5003->5003/tcp, :::5003->5003/tcp
                                               barista
[ec2-user@ip-10-0-1-142 app]$ 
```



	Name	Latest execution status	Latest source revisions	Latest execution started	Most recent executions
○	rabbitmq-pipeline-opwat-trainee-project-prod	✓ Succeeded	**Source** – fd5f641c: update	4 minutes ago	✓ ✓ ✓ View details
○	proxy-pipeline-opwat-trainee-project-prod	✓ Succeeded	**Source** – fd5f641c: update	4 minutes ago	✓ ✓ ✓ ✓ View details
○	postgres-pipeline-opwat-trainee-project-prod	✓ Succeeded	**Source** – fd5f641c: update	4 minutes ago	✓ ✓ ✓ ✓ View details
○	kitchen-pipeline-opwat-trainee-project-prod	✓ Succeeded	**Source** – fd5f641c: update	4 minutes ago	✓ ✓ ✓ View details
○	barista-pipeline-opwat-trainee-project-prod	✓ Succeeded	**Source** – fd5f641c: update	4 minutes ago	✓ ✓ ✓ ✓ View details
○	counter-pipeline-opwat-trainee-project-prod	✓ Succeeded	**Source** – fd5f641c: update	4 minutes ago	✓ ✓ ✓ ✓ View details
○	product-pipeline-opwat-trainee-project-prod	✓ Succeeded	**Source** – fd5f641c: update	4 minutes ago	✓ ✓ ✓ ✗ ✗ View details
○	web-pipeline-opwat-trainee-project-prod	✓ Succeeded	**Source** – fd5f641c: update	4 minutes ago	✓ ✓ ✓ ✓ ✓ View details



✓ ✓ ✓ ✓

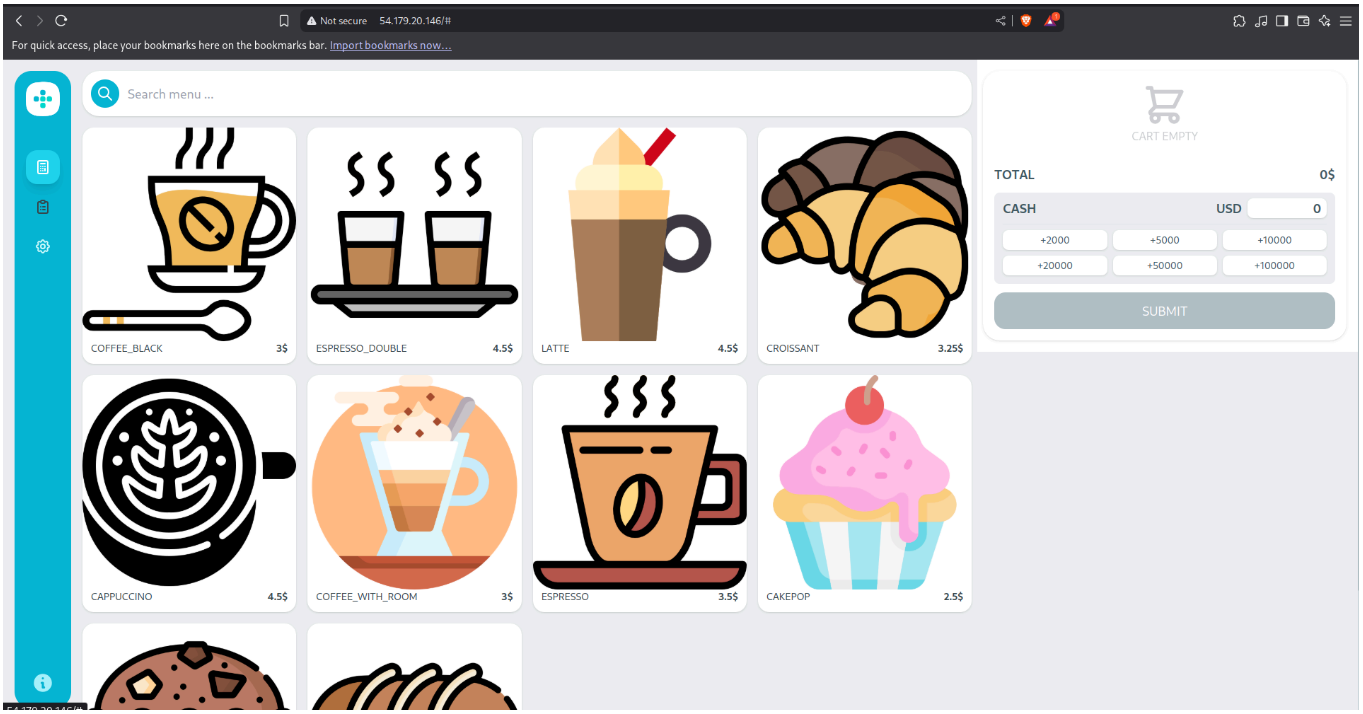
✓ **Source**
d3835859-6a47-470a-99de-a07c1fec360
All actions succeeded.
Source
✓ Bitbucket
4 minutes ago
fd5f641c Source: update

✓ **Scan-rabbitmq-opwat-trainee-project-prod**
d3835859-6a47-470a-99de-a07c1fec360
All actions succeeded.
TrivyScan
✓ AWS CodeBuild
3 minutes ago
fd5f641c Source: update

✓ **Push-ToECR-rabbitmq-opwat-trainee-project-prod**
d3835859-6a47-470a-99de-a07c1fec360
All actions succeeded.
PushToECR
✓ AWS CodeBuild
2 minutes ago
fd5f641c Source: update

✓ **Deploy-rabbitmq-opwat-trainee-project-EC2-Dev**
d3835859-6a47-470a-99de-a07c1fec360
All actions succeeded.
CodeDeploy
✓ AWS CodeDeploy
1 minute ago
fd5f641c Source: update

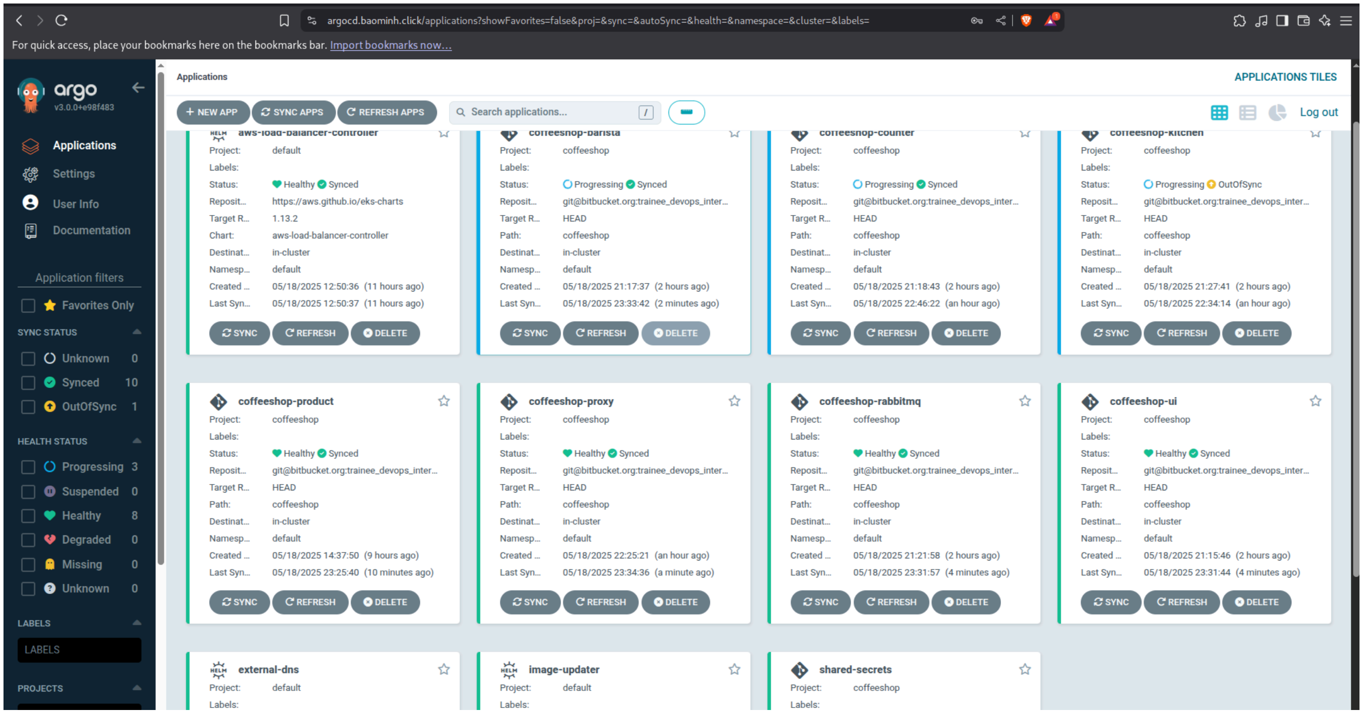
You can access the dev environment via the EC2 IP



Not secure 54.179.20.146/#

For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

And you will see all applications running successfully



aws-load-balancer-controller
Project: default
Labels:
Status: Healthy Synced
Reposit... https://aws.github.io/eks-charts
Target R... 1.13.2
Chart: aws-load-balancer-controller
Destinat... in-cluster
Namesp... default
Created ... 05/18/2025 12:50:36 (11 hours ago)
Last Syn... 05/18/2025 12:50:37 (11 hours ago)

coffeeshop-barista
Project: coffeeshop
Labels:
Status: Progressing Synced
Reposit... git@bitbucket.org:trainee_devops_inter...
Target R... HEAD
Path: coffeeshop
Destinat... in-cluster
Namesp... default
Created ... 05/18/2025 21:17:37 (2 hours ago)
Last Syn... 05/18/2025 23:33:42 (2 minutes ago)

coffeeshop-counter
Project: coffeeshop
Labels:
Status: Progressing Synced
Reposit... git@bitbucket.org:trainee_devops_inter...
Target R... HEAD
Path: coffeeshop
Destinat... in-cluster
Namesp... default
Created ... 05/18/2025 21:18:43 (2 hours ago)
Last Syn... 05/18/2025 22:46:22 (an hour ago)

coffeeshop-kitchen
Project: coffeeshop
Labels:
Status: Progressing OutOfSync
Reposit... git@bitbucket.org:trainee_devops_inter...
Target R... HEAD
Path: coffeeshop
Destinat... in-cluster
Namesp... default
Created ... 05/18/2025 21:27:41 (2 hours ago)
Last Syn... 05/18/2025 22:34:14 (an hour ago)

coffeeshop-product
Project: coffeeshop
Labels:
Status: Healthy Synced
Reposit... git@bitbucket.org:trainee_devops_inter...
Target R... HEAD
Path: coffeeshop
Destinat... in-cluster
Namesp... default
Created ... 05/18/2025 14:37:50 (9 hours ago)
Last Syn... 05/18/2025 23:25:40 (10 minutes ago)

coffeeshop-proxy
Project: coffeeshop
Labels:
Status: Healthy Synced
Reposit... git@bitbucket.org:trainee_devops_inter...
Target R... HEAD
Path: coffeeshop
Destinat... in-cluster
Namesp... default
Created ... 05/18/2025 22:25:21 (an hour ago)
Last Syn... 05/18/2025 23:34:36 (a minute ago)

coffeeshop-rabbitmq
Project: coffeeshop
Labels:
Status: Healthy Synced
Reposit... git@bitbucket.org:trainee_devops_inter...
Target R... HEAD
Path: coffeeshop
Destinat... in-cluster
Namesp... default
Created ... 05/18/2025 21:21:58 (2 hours ago)
Last Syn... 05/18/2025 23:31:57 (4 minutes ago)

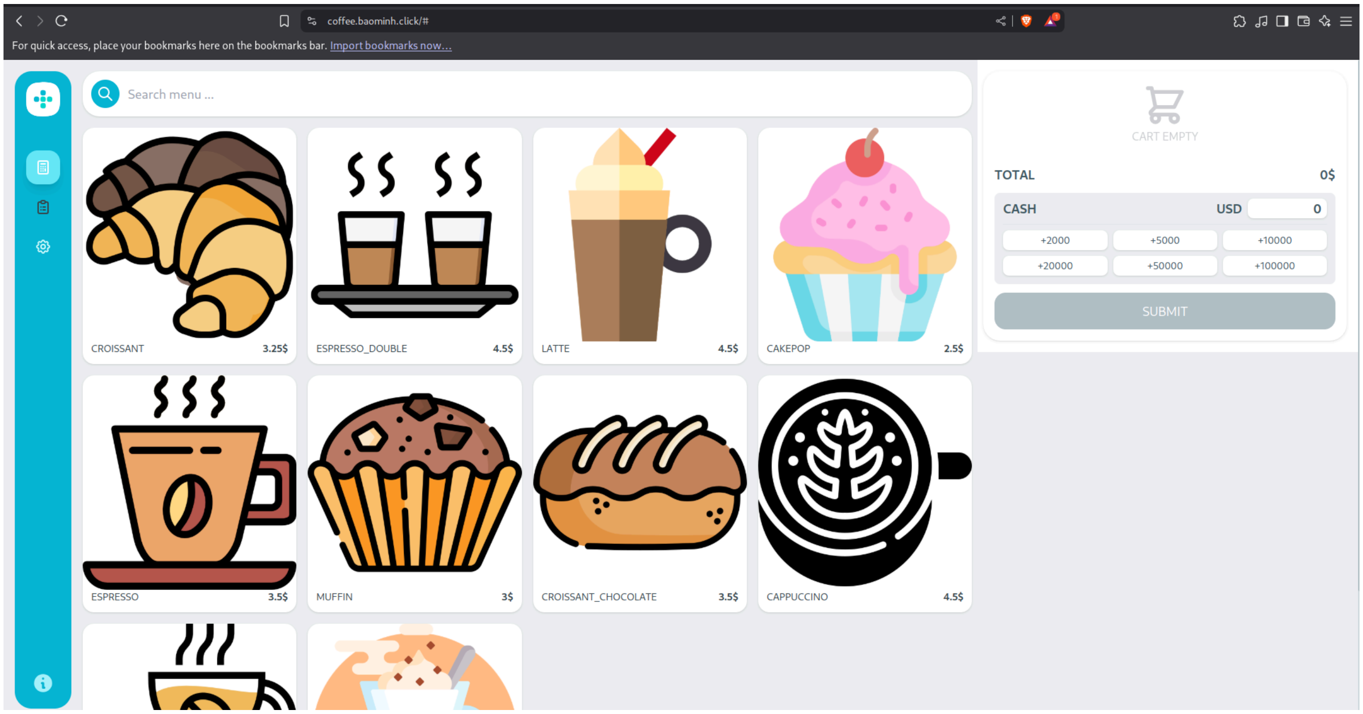
coffeeshop-ui
Project: coffeeshop
Labels:
Status: Healthy Synced
Reposit... git@bitbucket.org:trainee_devops_inter...
Target R... HEAD
Path: coffeeshop
Destinat... in-cluster
Namesp... default
Created ... 05/18/2025 21:15:46 (2 hours ago)
Last Syn... 05/18/2025 23:31:44 (4 minutes ago)

external-dns
Project: default
Labels:

image-updater
Project: default
Labels:

shared-secrets
Project: coffeeshop
Labels:

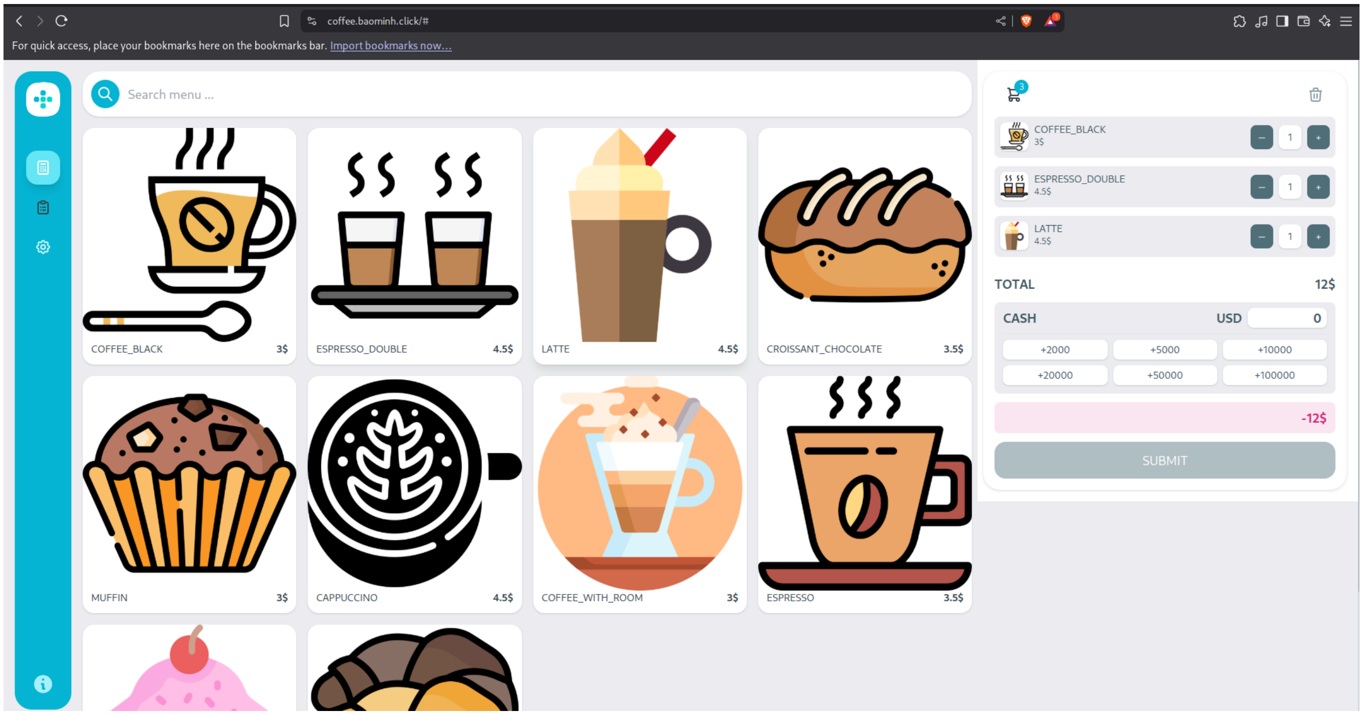
You can access:



CROISSANT 3.25$
ESPRESSO_DOUBLE 4.5$
LATTE 4.5$
CAKEPOP 2.5$
ESPRESSO 3.5$
MUFFIN 3$
CROISSANT_CHOCOLATE 3.5$
CAPPUCCINO 4.5$

TOTAL 0$
CASH USD 0

4. The homepage of the application



coffee.baominh.click/#

For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

COFFEE_BLACK 3$

ESPRESSO_DOUBLE 4.5$

LATTE 4.5$

CROISSANT_CHOCOLATE 3.5$

MUFFIN 3$

CAPPUCCINO 4.5$

COFFEE_WITH_ROOM 3$

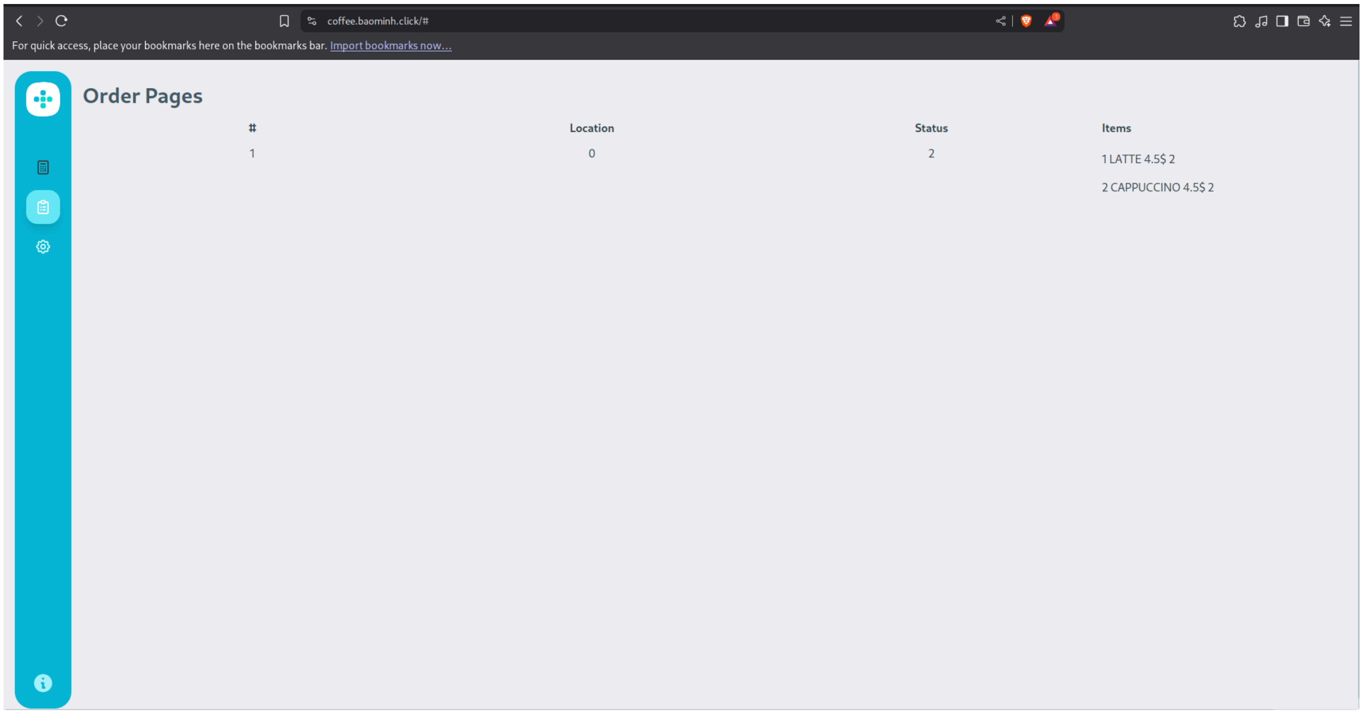
ESPRESSO 3.5$

COFFEE_BLACK 3$	– 1 +	
ESPRESSO_DOUBLE 4.5$	– 1 +	
LATTE 4.5$	– 1 +	

TOTAL 12$

CASH USD 0

-12$



coffee.baominh.click/#

For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

Order Pages

#	Location	Status	Items
1	0	2	1 LATTE 4.5$ 2 2 CAPPUCCINO 4.5$ 2