Cloud-Native Hostel Management System: Architecture and Optimization

Instructor:
Dr. Zunnurain

Student Names:
Asma Haider BSSE23051
Uzair Abdullah BSSE23075

Table of Contents

Project Summary

This project involved the development and deployment of a containerized web application designed to modernize student record management and room allocation systems. By migrating from a legacy monolithic architecture to AWS Fargate, the solution leverages serverless container orchestration to ensure high availability and automated scaling without the overhead of managing underlying infrastructure.

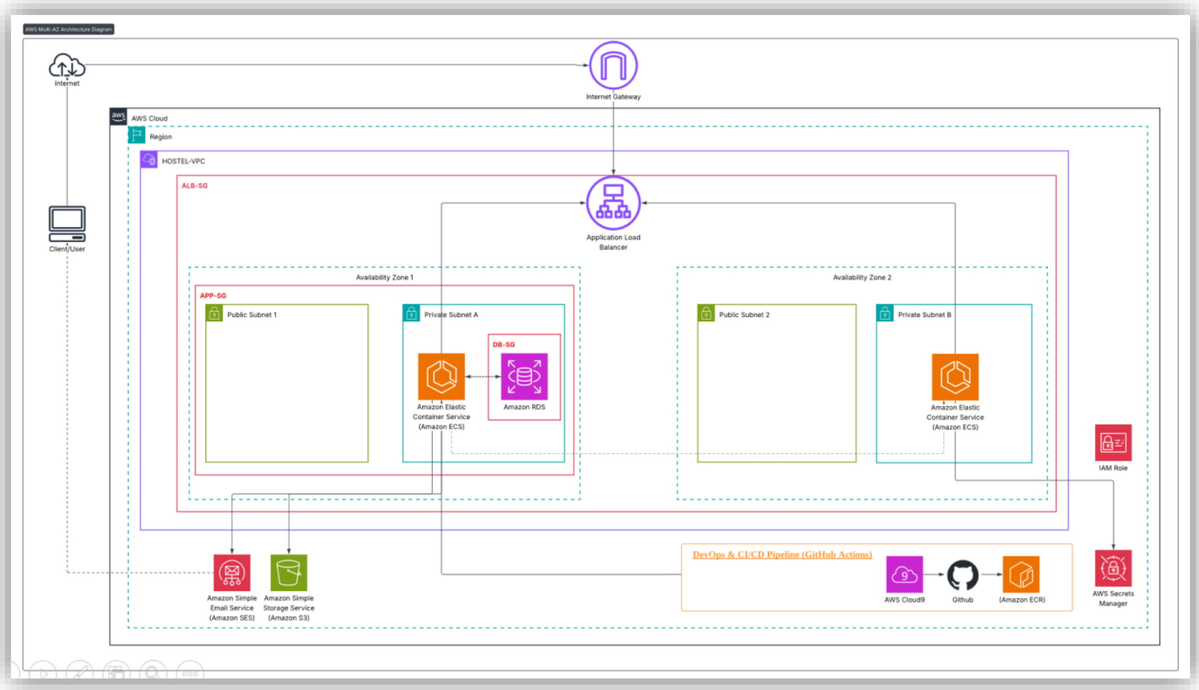
A critical component of this modernization was addressing the challenge of ephemeral storage inherent in containerized environments; this was achieved by integrating cloud-native persistence layers, ensuring that critical student data and allocation records remain durable and consistent across container lifecycles.

Aim and Objectives

This system leverages a serverless container strategy using AWS ECS Fargate, which eliminates manual infrastructure management while providing automated auto-scaling to meet fluctuating demand. The architecture is built for high availability and security, deploying resources across Multi-Availability Zones and isolating the database within private subnets to ensure it remains inaccessible from the public internet.

Media storage is externalized to Amazon S3 for durable object management, and application security is bolstered by a custom OTP logic integrated via SMTP to prevent unauthorized access. Additionally, the platform supports administrative efficiency through a bulk upload feature for processing large datasets via Excel.

Architecture Diagram


AWS Multi-AZ Architecture Diagram
Internet
Internet Gateway
Client/User
AWS Cloud
Region
HOSTEL-VPC
ALB-SG
Application Load Balancer
Availability Zone 1
APP-SG
Public Subnet 1
Private Subnet A
DB-SG
Amazon Elastic Container Service (Amazon ECS)
Amazon RDS
Availability Zone 2
Public Subnet 2
Private Subnet B
Amazon Elastic Container Service (Amazon ECS)
IAM Role
Amazon Simple Email Service (Amazon SES)
Amazon Simple Storage Service (Amazon S3)
DevOps & CI/CD Pipeline (GitHub Actions)
AWS Cloud9
Github
(Amazon ECR)
AWS Secrets Manager

This AWS architecture represents a highly available, multi-tiered web application deployed across two Availability Zones (AZs) within a single Region. It utilizes a containerized approach for the application layer and a managed relational database for the data layer, all supported by a modern CI/CD pipeline.

Core Components & Purpose

- Frontend/Backend: Flask Application (Dockerized).
- Database: Amazon RDS (MySQL).
- Storage: Amazon S3 (Profile Images).
- Traffic: Application Load Balancer (ALB).
- Auth: Amazon SES (Email OTP).

Infrastructure & Networking

- VPC (Virtual Private Cloud): Provides a logically isolated section of the AWS Cloud where you can launch AWS resources in a defined virtual network.
- Public Subnets: Host the Application Load Balancer (ALB), which acts as the single point of contact for clients and distributes incoming application traffic across multiple targets.
- Private Subnets: Host the application containers and database instances, shielding them from direct internet access for enhanced security.
- Availability Zones (AZ1 & AZ2): Ensure high availability and fault tolerance; if one AZ fails, the application remains operational in the other.

Component	Resource Name	CIDR/Setting	Purpose
VPC	Hostel-VPC	10.0.0.0/16	Network Isolation
Public Subnet	subnet-public-1/2	10.0.0.0/20	Hosts Load Balancer
Private Subnet	subnet-private-1/2	10.0.128.0/20	Hosts App Containers
Availability Zones	us-east-1a, us-east-1b	Multi-AZ	High Availability

Application & Data Layer

- Amazon ECS (Elastic Container Service): Orchestrates and runs containerized applications (often using AWS Fargate for serverless management).
- Amazon RDS (Relational Database Service): A managed database service that, in this "Multi-AZ" setup, maintains a primary instance in one AZ and a standby in another for automatic failover.
- Amazon S3: Used for scalable object storage, typically for static assets or backups.
- AWS Secrets Manager: Securely stores and manages sensitive information like database credentials and API keys.

DevOps & Integration

- GitHub Actions: Automates the CI/CD workflow, triggering builds and deployments whenever code changes are pushed to the repository.
- Amazon ECR (Elastic Container Registry): A fully managed Docker container registry used to store and manage container images.
- AWS Cloud9: A cloud-based IDE used by developers to write, run, and debug code directly within the AWS environment.

- SMTP (Simple Mail Transfer Protocol): A standard email sending service used by the application to send notifications or transactional emails through an email server.

Technology Stack Summary

Layer	**Technology**
Language	Python 3.11 (Flask Framework)
Frontend	HTML5, CSS3 (Particles.js), Bootstrap 5
Container Engine	Docker
Orchestrator	AWS Fargate (Serverless ECS)
Database	Amazon RDS (MySQL 8.0)
Cloud Storage	Amazon S3
Email Service	Amazon SES
Infrastructure	VPC, Internet Gateway, ALB
CI/CD	GitHub Actions

DevOps & CI/CD Pipeline

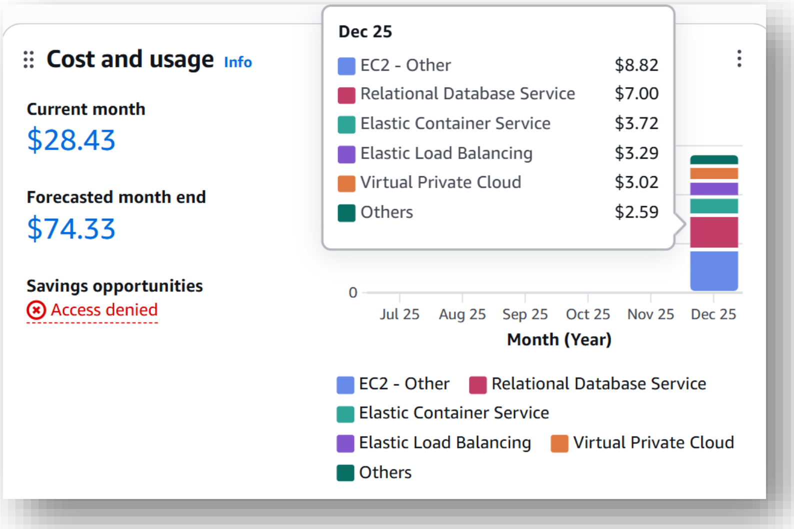
We moved away from manual deployments to a modern GitOps workflow.

- Source Control: GitHub (main branch).
- Pipeline Orchestrator: GitHub Actions.
- Container Registry: Amazon ECR.

The Pipeline (deploy.yml) Workflow:

1. Trigger: Developer pushes code to GitHub.
2. Checkout: GitHub Runner pulls the latest code.
3. Auth: Runner authenticates with AWS using secure Secrets (AWS_ACCESS_KEY_ID, etc.).
4. Build: Docker image is built from the Dockerfile.
5. Push: Image is tagged and pushed to Amazon ECR.
6. Deploy: The runner commands AWS ECS to update the service. ECS drains old connections and spins up new Fargate tasks with the updated image.

AWS Architecture Design, Implementation and Screenshots



Cost and usage Info

Current month
$28.43

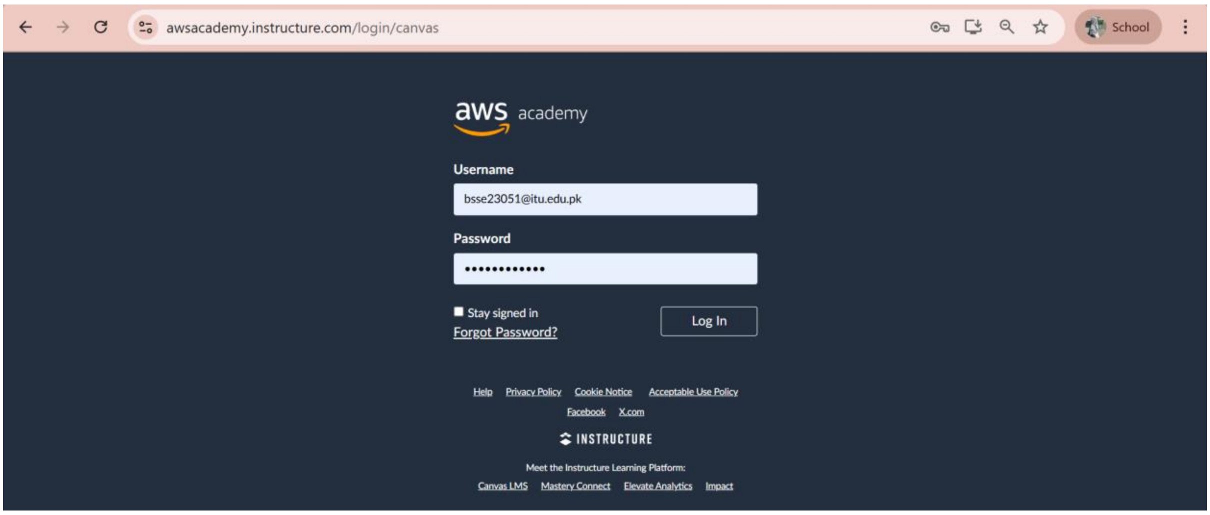
Forecasted month end
$74.33

Savings opportunities
Access denied

Dec 25

🟦 EC2 - Other	$8.82
🟥 Relational Database Service	$7.00
🟩 Elastic Container Service	$3.72
🟪 Elastic Load Balancing	$3.29
🟧 Virtual Private Cloud	$3.02
🟩 Others	$2.59

AWS account setup



awsacademy.instructure.com/login/canvas

Username
bsse23051@itu.edu.pk

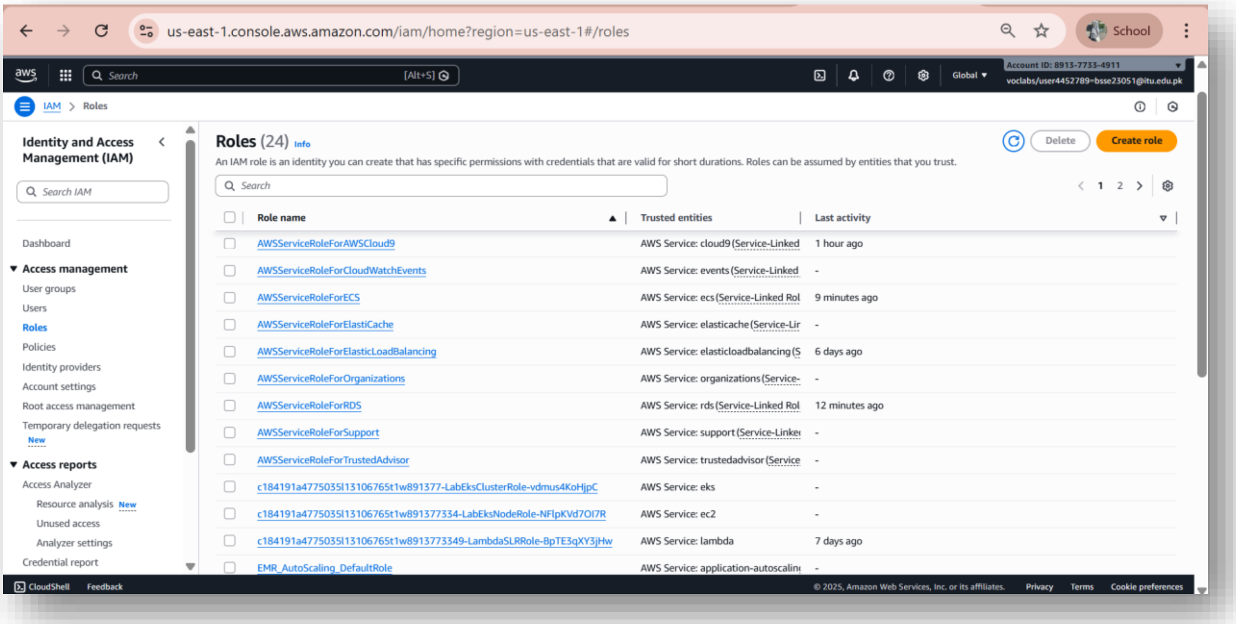
Password
••••••••••••

□ Stay signed in
Forgot Password?

Log In

Help Privacy Policy Cookie Notice Acceptable Use Policy
Facebook X.com
INSTRUCTURE
Meet the Instructure Learning Platform:
Canvas LMS Mastery Connect Elevate Analytics Impact

IAM roles and policies



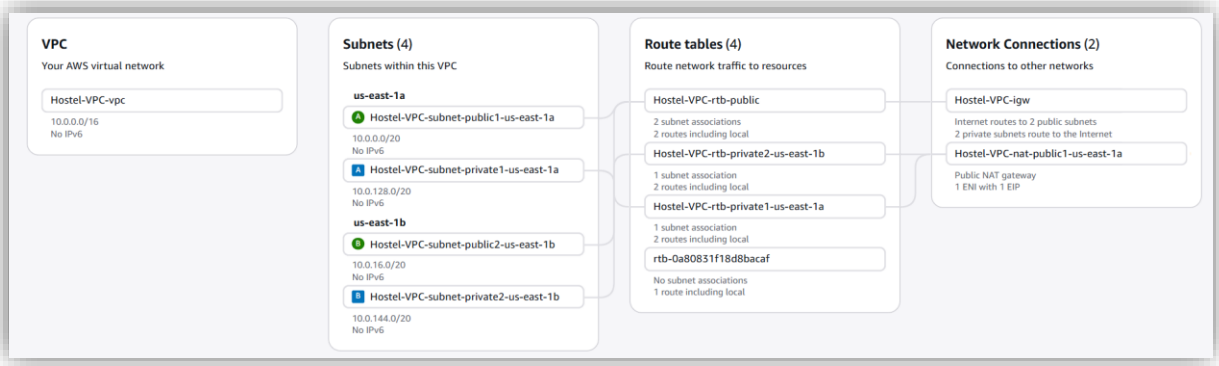
Roles (24) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

Role name	Trusted entities	Last activity
AWSServiceRoleForAWSCloud9	AWS Service: cloud9 (Service-Linked	1 hour ago
AWSServiceRoleForCloudWatchEvents	AWS Service: events (Service-Linked	-
AWSServiceRoleForECS	AWS Service: ecs (Service-Linked Rol	9 minutes ago
AWSServiceRoleForElastiCache	AWS Service: elasticache (Service-Lir	-
AWSServiceRoleForElasticLoadBalancing	AWS Service: elasticloadbalancing (S	6 days ago
AWSServiceRoleForOrganizations	AWS Service: organizations (Service-	-
AWSServiceRoleForRDS	AWS Service: rds (Service-Linked Rol	12 minutes ago
AWSServiceRoleForSupport	AWS Service: support (Service-Linke	-
AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service	-
c184191a4775035l13106765t1w891377-LabEksClusterRole-vdmus4KoHjpC	AWS Service: eks	-
c184191a4775035l13106765t1w891377334-LabEksNodeRole-NFlpKVd7OI7R	AWS Service: ec2	-
c184191a4775035l13106765t1w8913773349-LambdaSLRRole-BpTE3qXY3jHw	AWS Service: lambda	7 days ago
EMR_AutoScaling_DefaultRole	AWS Service: application-autoscalin	-

VPC/subnets/security groups

VPC



VPC

Your AWS virtual network

Hostel-VPC-vpc
10.0.0.0/16
No IPv6

Subnets (4)

Subnets within this VPC

us-east-1a

A Hostel-VPC-subnet-public1-us-east-1a
10.0.0.0/20
No IPv6

A Hostel-VPC-subnet-private1-us-east-1a
10.0.128.0/20
No IPv6

us-east-1b

B Hostel-VPC-subnet-public2-us-east-1b
10.0.16.0/20
No IPv6

B Hostel-VPC-subnet-private2-us-east-1b
10.0.144.0/20
No IPv6

Route tables (4)

Route network traffic to resources

Hostel-VPC-rtb-public
2 subnet associations
2 routes including local

Hostel-VPC-rtb-private2-us-east-1b
1 subnet association
2 routes including local

Hostel-VPC-rtb-private1-us-east-1a
1 subnet association
2 routes including local

rtb-0a80831f18d8bacaf
No subnet associations
1 route including local

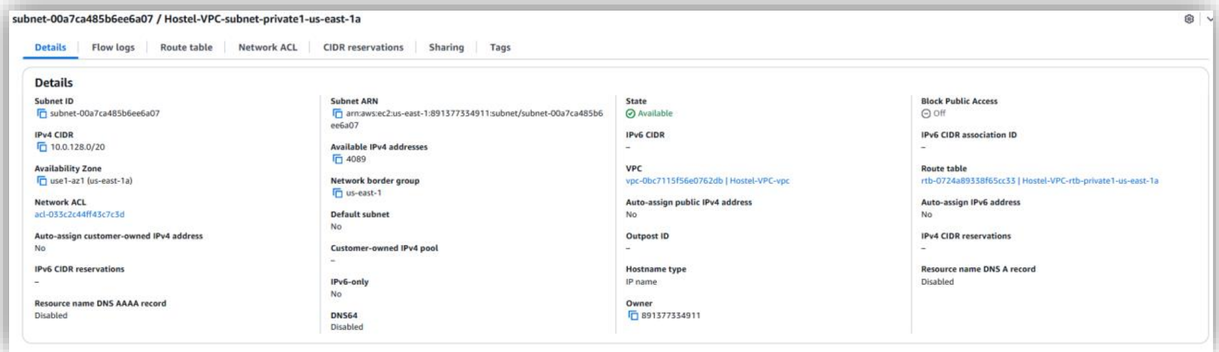
Network Connections (2)

Connections to other networks

Hostel-VPC-igw
Internet routes to 2 public subnets
2 private subnets route to the Internet

Hostel-VPC-nat-public1-us-east-1a
Public NAT gateway
1 ENI with 1 EIP

SUBNETS (PRIVATE AND PUBLIC)



subnet-00a7ca485b6ee6a07 / Hostel-VPC-subnet-private1-us-east-1a

Details

Subnet ID
subnet-00a7ca485b6ee6a07

IPv4 CIDR
10.0.128.0/20

Availability Zone
use1-az1 (us-east-1a)

Network ACL
acl-033c2c44ff43c7c3d

Auto-assign customer-owned IPv4 address
No

IPv6 CIDR reservations
–

Resource name DNS AAAA record
Disabled

Subnet ARN
arn:aws:ec2:us-east-1:891377334911:subnet/subnet-00a7ca485b6ee6a07

Available IPv4 addresses
4089

Network border group
us-east-1

Default subnet
No

Customer-owned IPv4 pool
–

IPv6-only
No

DNS64
Disabled

State
Available

IPv6 CIDR
–

VPC
vpc-0bc7115f56e0762db | Hostel-VPC-vpc

Auto-assign public IPv4 address
No

Outpost ID
–

Hostname type
IP name

Owner
891377334911

Block Public Access
Off

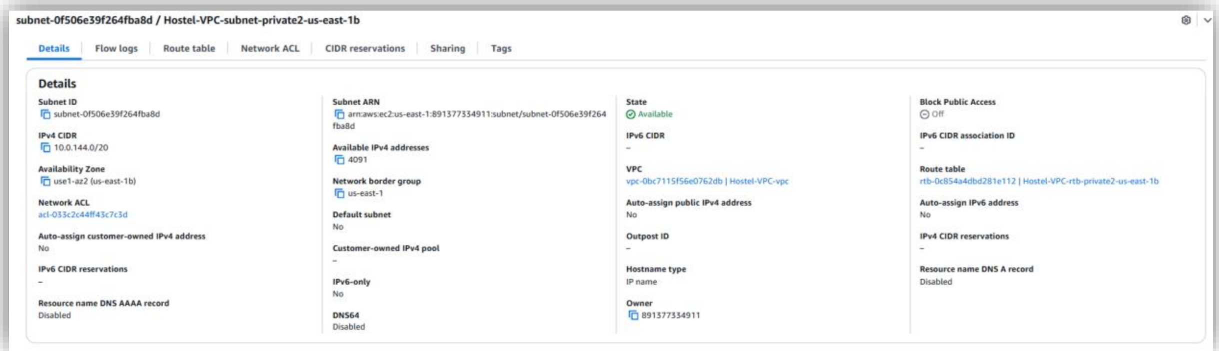
IPv6 CIDR association ID
–

Route table
rtb-0724a89338f65cc33 | Hostel-VPC-rtb-private1-us-east-1a

Auto-assign IPv6 address
No

IPv4 CIDR reservations
–

Resource name DNS A record
Disabled



subnet-0f506e39f264fba8d / Hostel-VPC-subnet-private2-us-east-1b

Details

Subnet ID
subnet-0f506e39f264fba8d

IPv4 CIDR
10.0.144.0/20

Availability Zone
use1-az2 (us-east-1b)

Network ACL
acl-033c2c44ff43c7c3d

Auto-assign customer-owned IPv4 address
No

IPv6 CIDR reservations
–

Resource name DNS AAAA record
Disabled

Subnet ARN
arn:aws:ec2:us-east-1:891377334911:subnet/subnet-0f506e39f264fba8d

Available IPv4 addresses
4091

Network border group
us-east-1

Default subnet
No

Customer-owned IPv4 pool
–

IPv6-only
No

DNS64
Disabled

State
Available

IPv6 CIDR
–

VPC
vpc-0bc7115f56e0762db | Hostel-VPC-vpc

Auto-assign public IPv4 address
No

Outpost ID
–

Hostname type
IP name

Owner
891377334911

Block Public Access
Off

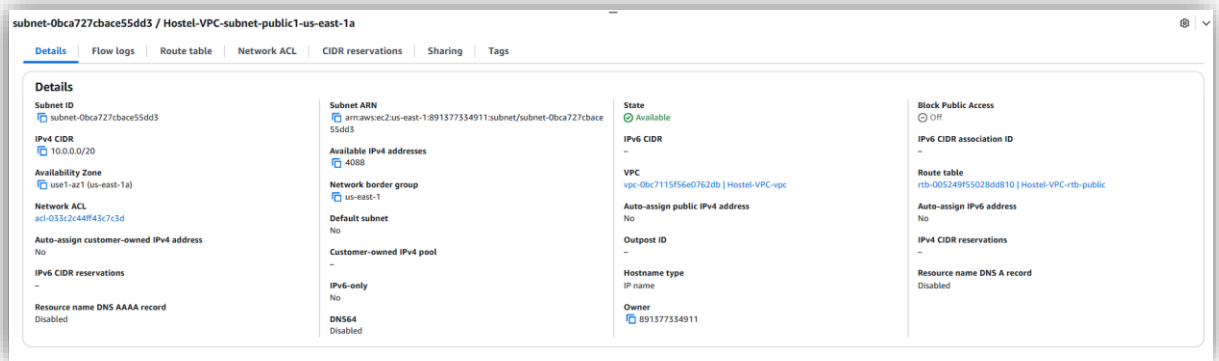
IPv6 CIDR association ID
–

Route table
rtb-0c854a4dbd281e112 | Hostel-VPC-rtb-private2-us-east-1b

Auto-assign IPv6 address
No

IPv4 CIDR reservations
–

Resource name DNS A record
Disabled



subnet-0bca727cbace55dd3 / Hostel-VPC-subnet-public1-us-east-1a

Details

Subnet ID
subnet-0bca727cbace55dd3

IPv4 CIDR
10.0.0.0/20

Availability Zone
use1-az1 (us-east-1a)

Network ACL
acl-033c2c44ff43c7c3d

Auto-assign customer-owned IPv4 address
No

IPv6 CIDR reservations
–

Resource name DNS AAAA record
Disabled

Subnet ARN
arn:aws:ec2:us-east-1:891377334911:subnet/subnet-0bca727cbace55dd3

Available IPv4 addresses
4088

Network border group
us-east-1

Default subnet
No

Customer-owned IPv4 pool
–

IPv6-only
No

DNS64
Disabled

State
Available

IPv6 CIDR
–

VPC
vpc-0bc7115f56e0762db | Hostel-VPC-vpc

Auto-assign public IPv4 address
No

Outpost ID
–

Hostname type
IP name

Owner
891377334911

Block Public Access
Off

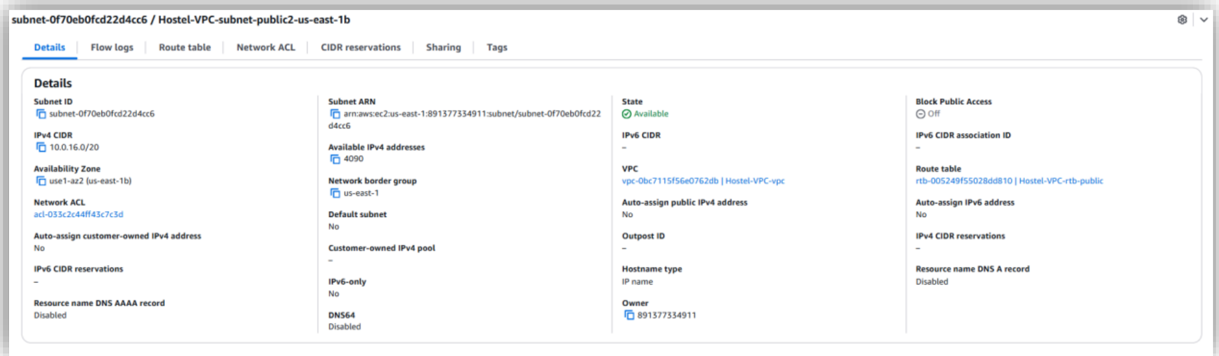
IPv6 CIDR association ID
–

Route table
rtb-005249f55028dd810 | Hostel-VPC-rtb-public

Auto-assign IPv6 address
No

IPv4 CIDR reservations
–

Resource name DNS A record
Disabled



subnet-0f70eb0fcd22d4cc6 / Hostel-VPC-subnet-public2-us-east-1b

Details

Subnet ID
subnet-0f70eb0fcd22d4cc6

IPv4 CIDR
10.0.16.0/20

Availability Zone
use1-az2 (us-east-1b)

Network ACL
acl-033c2c44ff43c7c3d

Auto-assign customer-owned IPv4 address
No

IPv6 CIDR reservations
–

Resource name DNS AAAA record
Disabled

Subnet ARN
arn:aws:ec2:us-east-1:891377334911:subnet/subnet-0f70eb0fcd22d4cc6

Available IPv4 addresses
4090

Network border group
us-east-1

Default subnet
No

Customer-owned IPv4 pool
–

IPv6-only
No

DNS64
Disabled

State
Available

IPv6 CIDR
–

VPC
vpc-0bc7115f56e0762db | Hostel-VPC-vpc

Auto-assign public IPv4 address
No

Outpost ID
–

Hostname type
IP name

Owner
891377334911

Block Public Access
Off

IPv6 CIDR association ID
–

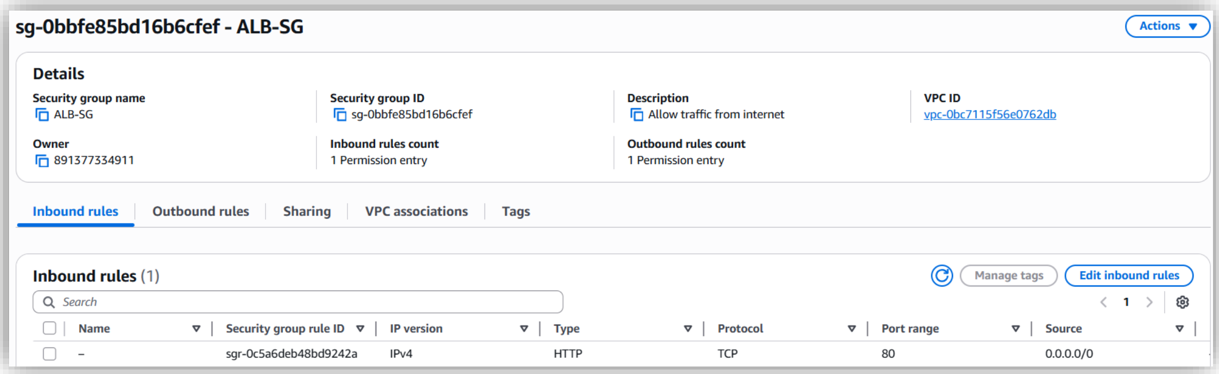
Route table
rtb-005249f55028dd810 | Hostel-VPC-rtb-public

Auto-assign IPv6 address
No

IPv4 CIDR reservations
–

Resource name DNS A record
Disabled

SECURITY GROUPS (ALB – APP – CLOUD9 – DB)



sg-0bbfe85bd16b6cfef - ALB-SG

Details

Security group name
ALB-SG

Security group ID
sg-0bbfe85bd16b6cfef

Description
Allow traffic from internet

VPC ID
vpc-0bc7115f56e0762db

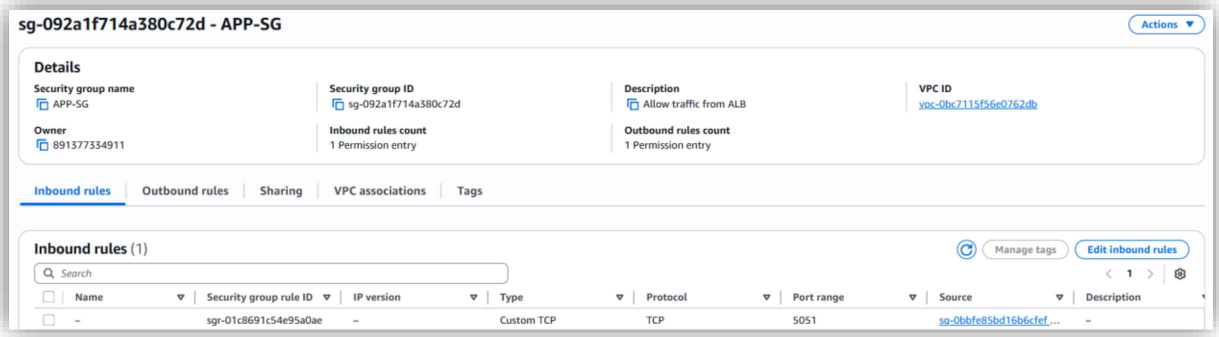
Owner
891377334911

Inbound rules count
1 Permission entry

Outbound rules count
1 Permission entry

Inbound rules (1)

□	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source
□	–	sgr-0c5a6deb48bd9242a	IPv4	HTTP	TCP	80	0.0.0.0/0



sg-092a1f714a380c72d - APP-SG

Details

Security group name
APP-SG

Security group ID
sg-092a1f714a380c72d

Description
Allow traffic from ALB

VPC ID
vpc-0bc7115f56e0762db

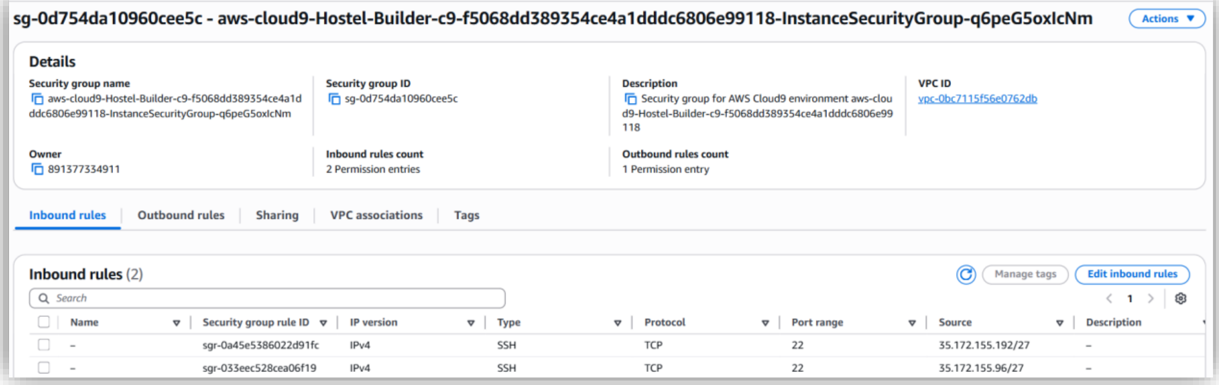
Owner
891377334911

Inbound rules count
1 Permission entry

Outbound rules count
1 Permission entry

Inbound rules (1)

□	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source	Description
□	–	sgr-01c8691c54e95a0ae	–	Custom TCP	TCP	5051	sg-0bbfe85bd16b6cfef ...	–



sg-0d754da10960cee5c - aws-cloud9-Hostel-Builder-c9-f5068dd389354ce4a1dddc6806e99118-InstanceSecurityGroup-q6peG5oxIcNm

Details

Security group name
aws-cloud9-Hostel-Builder-c9-f5068dd389354ce4a1dddc6806e99118-InstanceSecurityGroup-q6peG5oxIcNm

Security group ID
sg-0d754da10960cee5c

Description
Security group for AWS Cloud9 environment aws-cloud9-Hostel-Builder-c9-f5068dd389354ce4a1dddc6806e99118

VPC ID
vpc-0bc7115f56e0762db

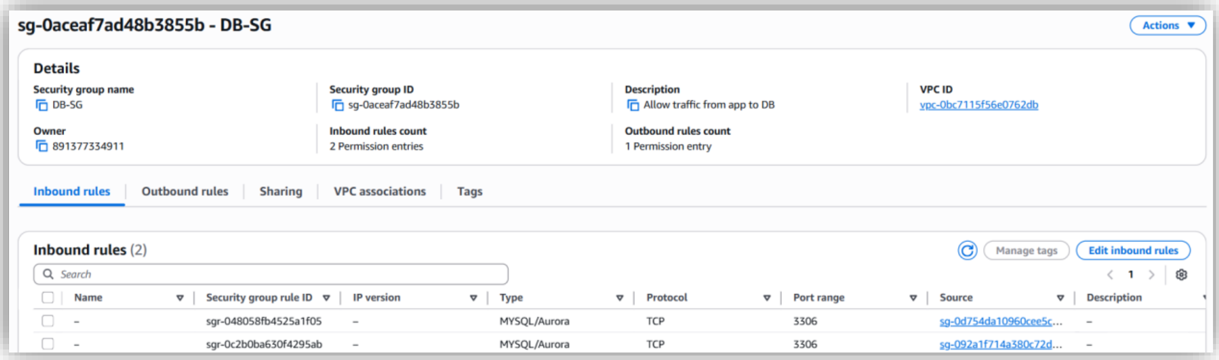
Owner
891377334911

Inbound rules count
2 Permission entries

Outbound rules count
1 Permission entry

Inbound rules (2)

□	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source	Description
□	–	sgr-0a45e5386022d91fc	IPv4	SSH	TCP	22	35.172.155.192/27	–
□	–	sgr-033eec528cea06f19	IPv4	SSH	TCP	22	35.172.155.96/27	–



sg-0aceaf7ad48b3855b - DB-SG

Details

Security group name
DB-SG

Security group ID
sg-0aceaf7ad48b3855b

Description
Allow traffic from app to DB

VPC ID
vpc-0bc7115f56e0762db

Owner
891377334911

Inbound rules count
2 Permission entries

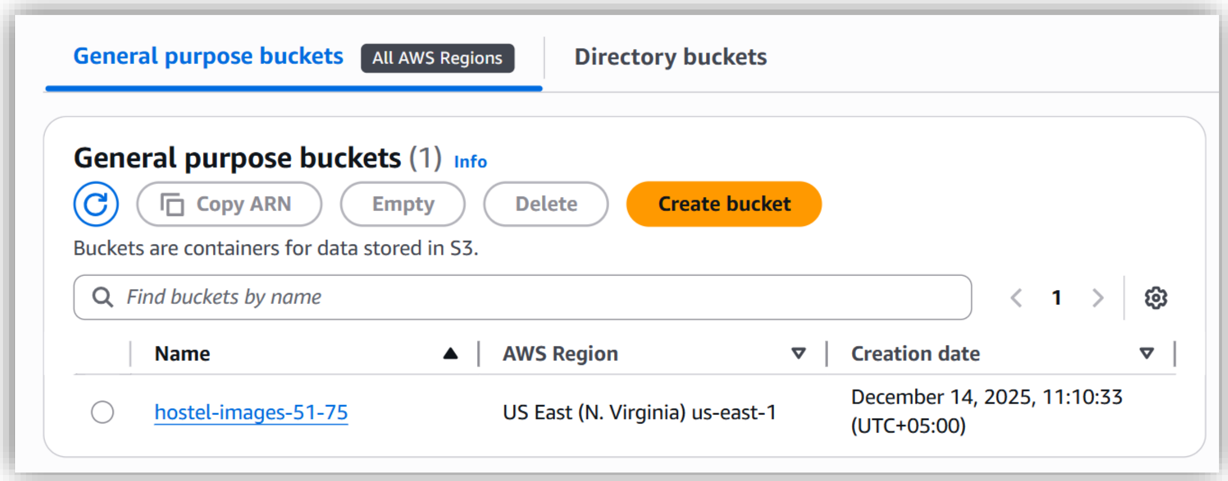
Outbound rules count
1 Permission entry

Inbound rules (2)

□	Name	Security group rule ID	IP version	Type	Protocol	Port range	Source	Description
□	–	sgr-048058fb4525a1f05	–	MYSQL/Aurora	TCP	3306	sg-0d754da10960cee5c...	–
□	–	sgr-0c2b0ba630f4295ab	–	MYSQL/Aurora	TCP	3306	sg-092a1f714a380c72d...	–

Used services

S3

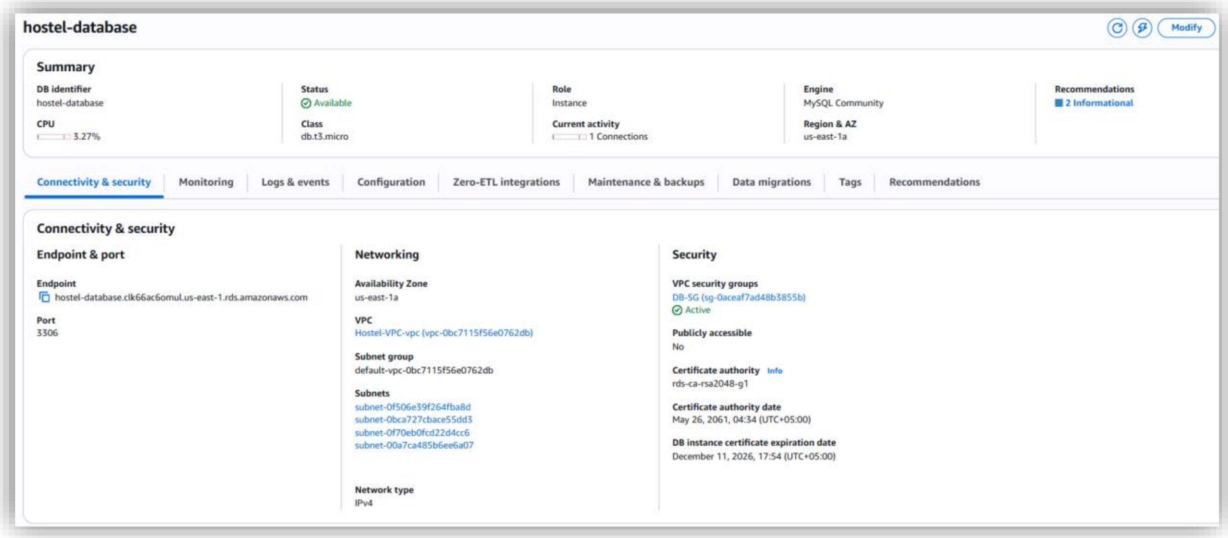


General purpose buckets (1)

Buckets are containers for data stored in S3.

	Name ▲	AWS Region ▽	Creation date ▽
○	hostel-images-51-75	US East (N. Virginia) us-east-1	December 14, 2025, 11:10:33 (UTC+05:00)

RDS



hostel-database

Summary

DB identifier
hostel-database

CPU
3.27%

Status
Available

Class
db.t3.micro

Role
Instance

Current activity
1 Connections

Engine
MySQL Community

Region & AZ
us-east-1a

Recommendations
2 Informational

Connectivity & security

Endpoint & port

Endpoint
hostel-database.clk66ac6omul.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability Zone
us-east-1a

VPC
Hostel-VPC-vpc (vpc-0bc7115f56e0762db)

Subnet group
default-vpc-0bc7115f56e0762db

Subnets
subnet-0f506e39f264fba8d
subnet-0bca727cbace55dd3
subnet-0f70eb0fcd22d4cc6
subnet-00a7ca485b6ee6a07

Network type
IPv4

Security

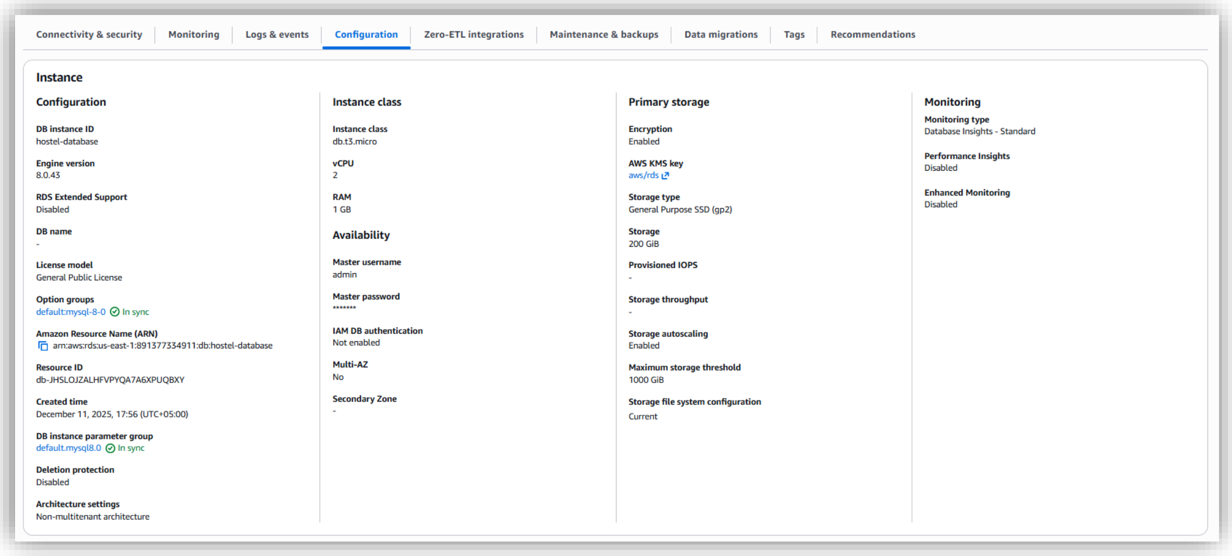
VPC security groups
DB-SG (sg-0aceaf7ad48b3855b)
Active

Publicly accessible
No

Certificate authority Info
rds-ca-rsa2048-g1

Certificate authority date
May 26, 2061, 04:34 (UTC+05:00)

DB instance certificate expiration date
December 11, 2026, 17:54 (UTC+05:00)



Instance

Configuration

DB instance ID
hostel-database

Engine version
8.0.43

RDS Extended Support
Disabled

DB name
-

License model
General Public License

Option groups
default:mysql-8-0 ✓ In sync

Amazon Resource Name (ARN)
arn:aws:rds:us-east-1:891377334911:db:hostel-database

Resource ID
db-JHSLOJZALHFVPYQA7A6XPUQBXY

Created time
December 11, 2025, 17:56 (UTC+05:00)

DB instance parameter group
default.mysql8.0 ✓ In sync

Deletion protection
Disabled

Architecture settings
Non-multitenant architecture

Instance class

Instance class
db.t3.micro

vCPU
2

RAM
1 GB

Availability

Master username
admin

Master password
****

IAM DB authentication
Not enabled

Multi-AZ
No

Secondary Zone
-

Primary storage

Encryption
Enabled

AWS KMS key
aws/rds

Storage type
General Purpose SSD (gp2)

Storage
200 GiB

Provisioned IOPS
-

Storage throughput
-

Storage autoscaling
Enabled

Maximum storage threshold
1000 GiB

Storage file system configuration
Current

Monitoring

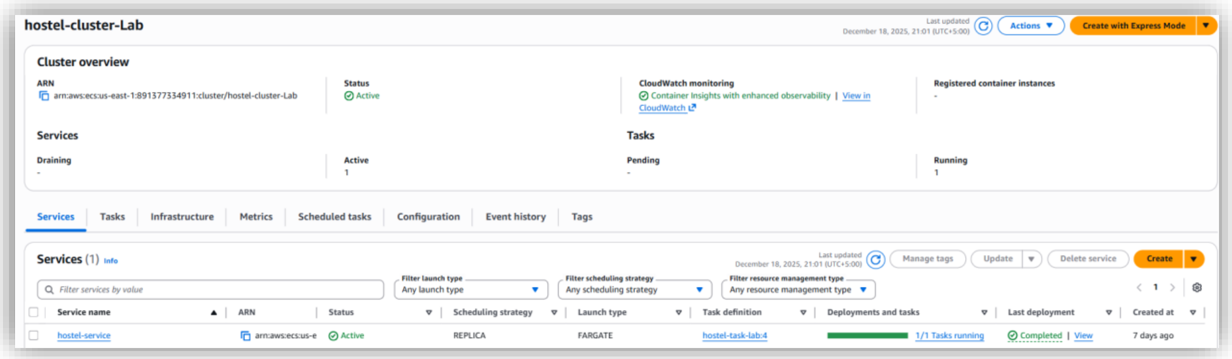
Monitoring type
Database Insights - Standard

Performance Insights
Disabled

Enhanced Monitoring
Disabled

Deployment process

ECS-CLUSTER



hostel-cluster-Lab

Last updated
December 18, 2025, 21:01 (UTC+5:00)

Cluster overview

ARN
arn:aws:ecs:us-east-1:891377334911:cluster/hostel-cluster-Lab

Status
✓ Active

CloudWatch monitoring
✓ Container Insights with enhanced observability | View in CloudWatch

Registered container instances
-

Services

Draining
-

Active
1

Tasks

Pending
-

Running
1

Services (1)

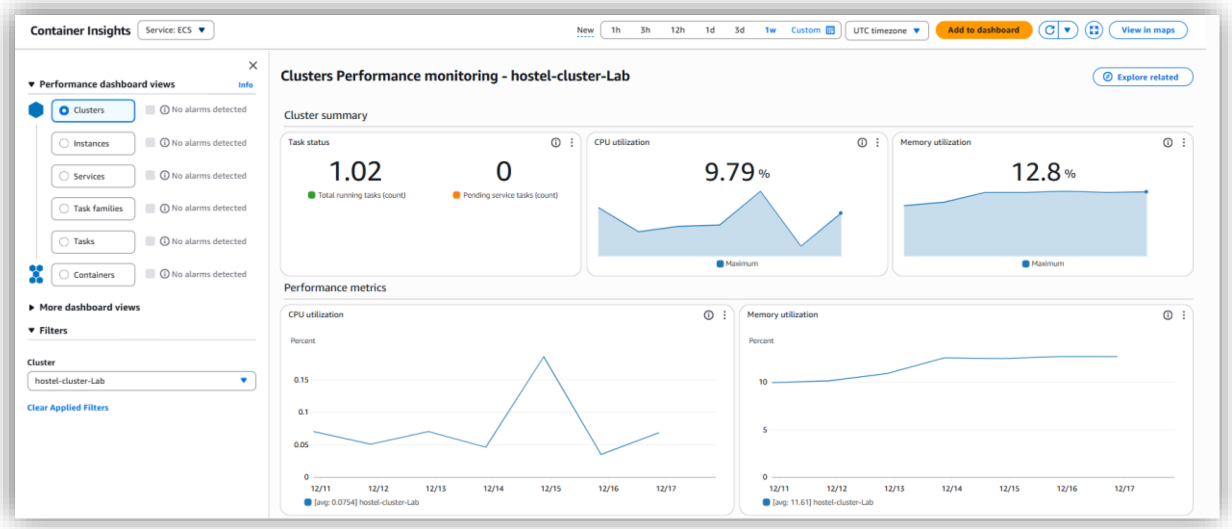
Last updated
December 18, 2025, 21:01 (UTC+5:00)

Filter launch type
Any launch type

Filter scheduling strategy
Any scheduling strategy

Filter resource management type
Any resource management type

□	Service name	ARN	Status	Scheduling strategy	Launch type	Task definition	Deployments and tasks	Last deployment	Created at
□	hostel-service	arn:aws:ecs:us-e	✓ Active	REPLICA	FARGATE	hostel-task-lab:4	1/1 Tasks running	✓ Completed | View	7 days ago



Cluster
hostel-cluster-Lab

Clusters Performance monitoring - hostel-cluster-Lab

Cluster summary

Task status
1.02 — Total running tasks (count)
0 — Pending service tasks (count)

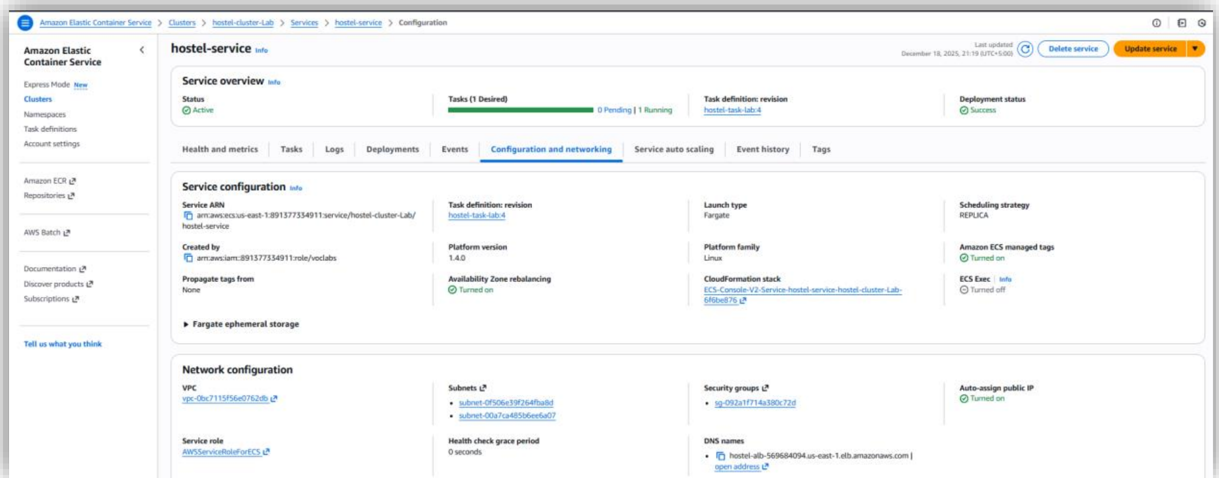
CPU utilization
9.79 %

Memory utilization
12.8 %

Performance metrics

CPU utilization
[avg: 0.0754] hostel-cluster-Lab

Memory utilization
[avg: 11.61] hostel-cluster-Lab



hostel-service

Last updated
December 18, 2025, 21:19 (UTC+5:00)

Service overview

Status	Tasks (1 Desired)	Task definition: revision	Deployment status
Active	0 Pending | 1 Running	hostel-task-lab:4	Success

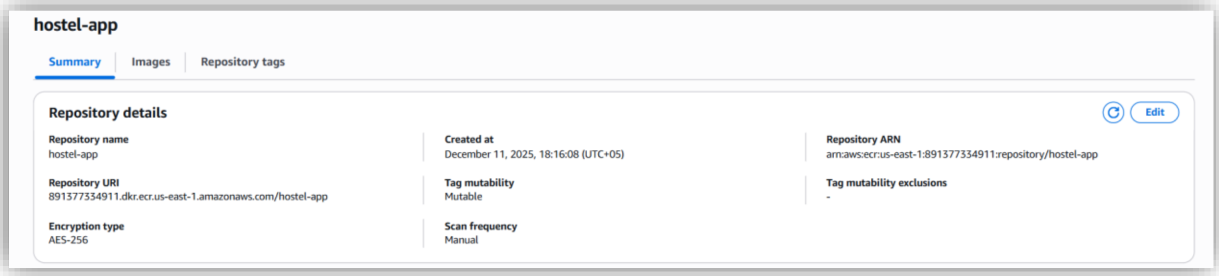
Service configuration

Service ARN	Task definition: revision	Launch type	Scheduling strategy
arn:aws:ecs:us-east-1:891377334911:service/hostel-cluster-Lab/hostel-service	hostel-task-lab:4	Fargate	REPLICA
Created by arn:aws:iam::891377334911:role/voclabs	**Platform version** 1.4.0	**Platform family** Linux	**Amazon ECS managed tags** Turned on
Propagate tags from None	**Availability Zone rebalancing** Turned on	**CloudFormation stack** ECS-Console-V2-Service-hostel-service-hostel-cluster-Lab-6f6be876	**ECS Exec** Turned off

Network configuration

VPC	Subnets	Security groups	Auto-assign public IP
vpc-0bc7115f56e0762db	subnet-0f506e39f264fba8d subnet-00a7ca485b6ee6a07	sg-092a1f714a380c72d	Turned on
Service role AWSServiceRoleForECS	**Health check grace period** 0 seconds	**DNS names** hostel-alb-569684094.us-east-1.elb.amazonaws.com | open address	

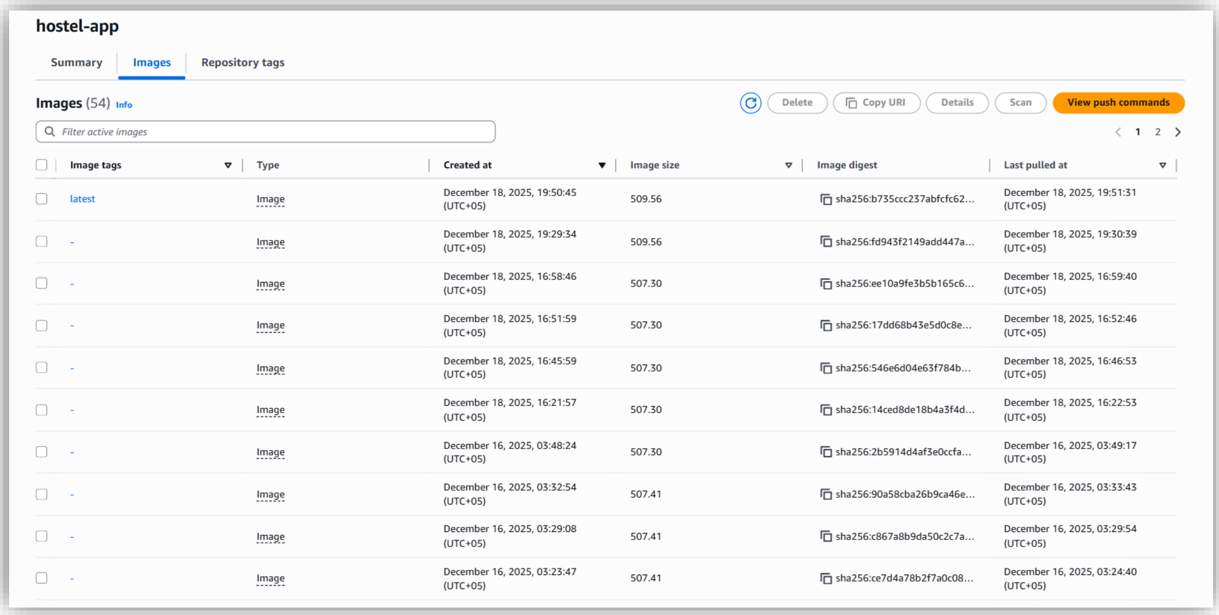
ECR- REPOSITORY



hostel-app

Repository details

Repository name	Created at	Repository ARN
hostel-app	December 11, 2025, 18:16:08 (UTC+05)	arn:aws:ecr:us-east-1:891377334911:repository/hostel-app
Repository URI 891377334911.dkr.ecr.us-east-1.amazonaws.com/hostel-app	**Tag mutability** Mutable	**Tag mutability exclusions** -
Encryption type AES-256	**Scan frequency** Manual	

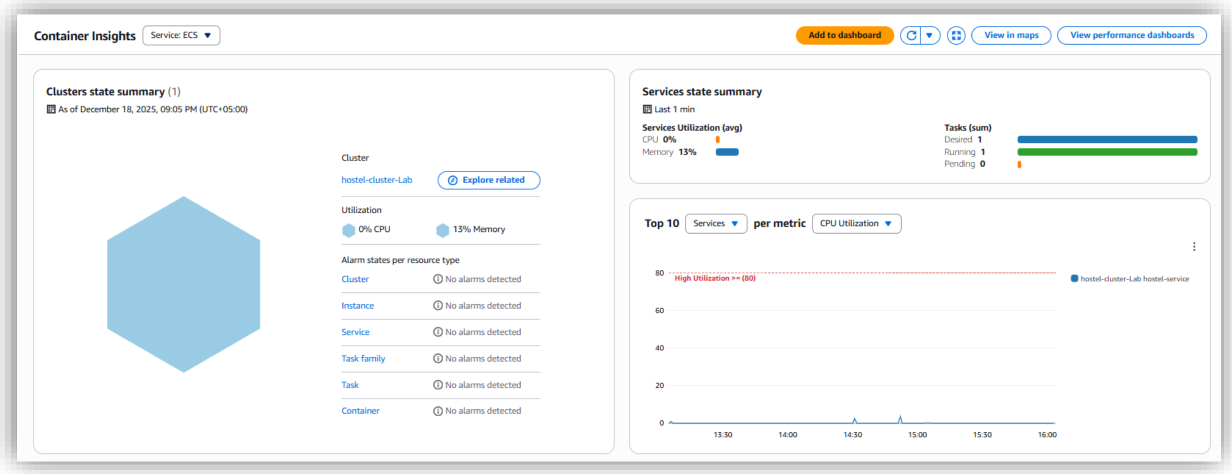


hostel-app

Images (54) Info

□	Image tags	Type	Created at	Image size	Image digest	Last pulled at
□	latest	Image	December 18, 2025, 19:50:45 (UTC+05)	509.56	sha256:b735ccc237abfcfc62...	December 18, 2025, 19:51:31 (UTC+05)
□	-	Image	December 18, 2025, 19:29:34 (UTC+05)	509.56	sha256:fd943f2149add447a...	December 18, 2025, 19:30:39 (UTC+05)
□	-	Image	December 18, 2025, 16:58:46 (UTC+05)	507.30	sha256:ee10a9fe3b5b165c6...	December 18, 2025, 16:59:40 (UTC+05)
□	-	Image	December 18, 2025, 16:51:59 (UTC+05)	507.30	sha256:17dd68b43e5d0c8e...	December 18, 2025, 16:52:46 (UTC+05)
□	-	Image	December 18, 2025, 16:45:59 (UTC+05)	507.30	sha256:546e6d04e63f784b...	December 18, 2025, 16:46:53 (UTC+05)
□	-	Image	December 18, 2025, 16:21:57 (UTC+05)	507.30	sha256:14ced8de18b4a3f4d...	December 18, 2025, 16:22:53 (UTC+05)
□	-	Image	December 16, 2025, 03:48:24 (UTC+05)	507.30	sha256:2b5914d4af3e0ccfa...	December 16, 2025, 03:49:17 (UTC+05)
□	-	Image	December 16, 2025, 03:32:54 (UTC+05)	507.41	sha256:90a58cba26b9ca46e...	December 16, 2025, 03:33:43 (UTC+05)
□	-	Image	December 16, 2025, 03:29:08 (UTC+05)	507.41	sha256:c867a8b9da50c2c7a...	December 16, 2025, 03:29:54 (UTC+05)
□	-	Image	December 16, 2025, 03:23:47 (UTC+05)	507.41	sha256:ce7d4a78b2f7a0c08...	December 16, 2025, 03:24:40 (UTC+05)

Testing workflow



Clusters state summary (1)

As of December 18, 2025, 09:05 PM (UTC+05:00)

Cluster

hostel-cluster-Lab

Utilization

0% CPU   13% Memory

Alarm states per resource type

Cluster	No alarms detected
Instance	No alarms detected
Service	No alarms detected
Task family	No alarms detected
Task	No alarms detected
Container	No alarms detected

Services state summary

Last 1 min

Services Utilization (avg)

CPU **0%**

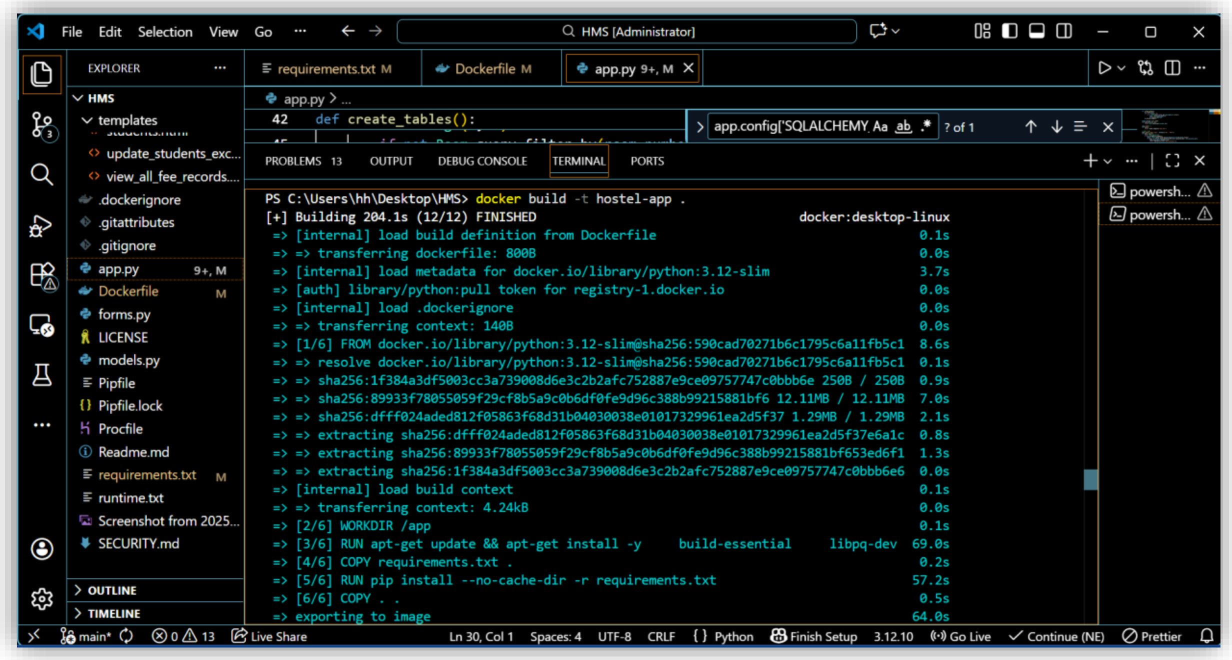
Memory **13%**

Tasks (sum)

Desired **1**

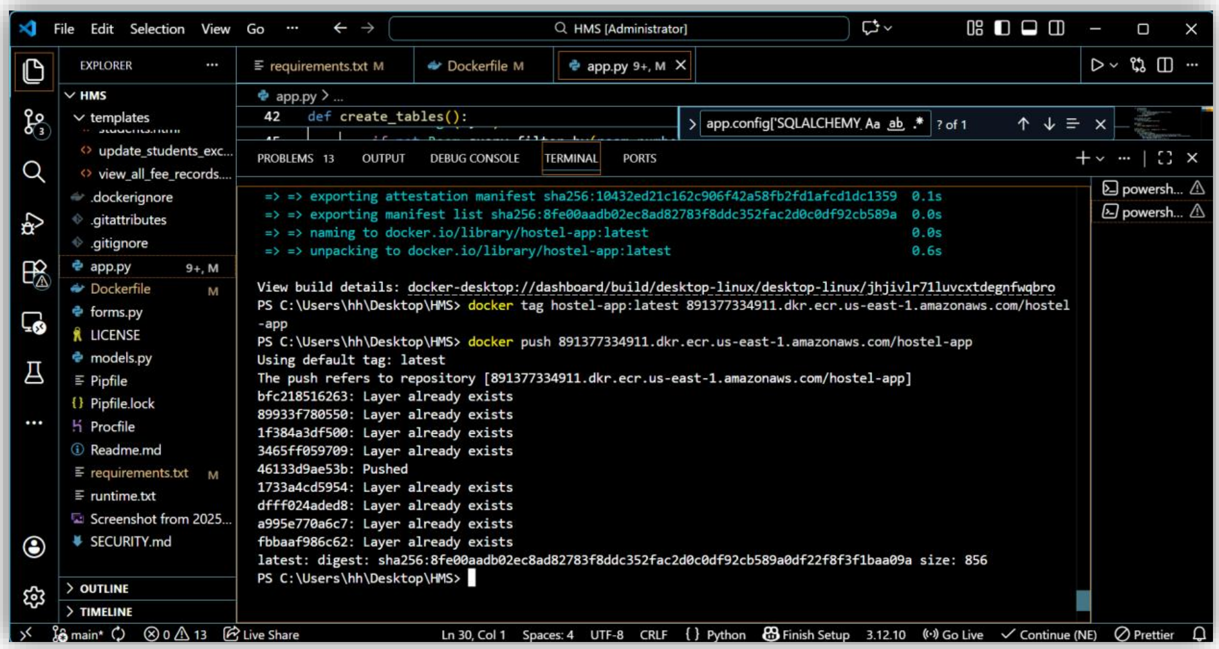
Running **1**

Pending **0**



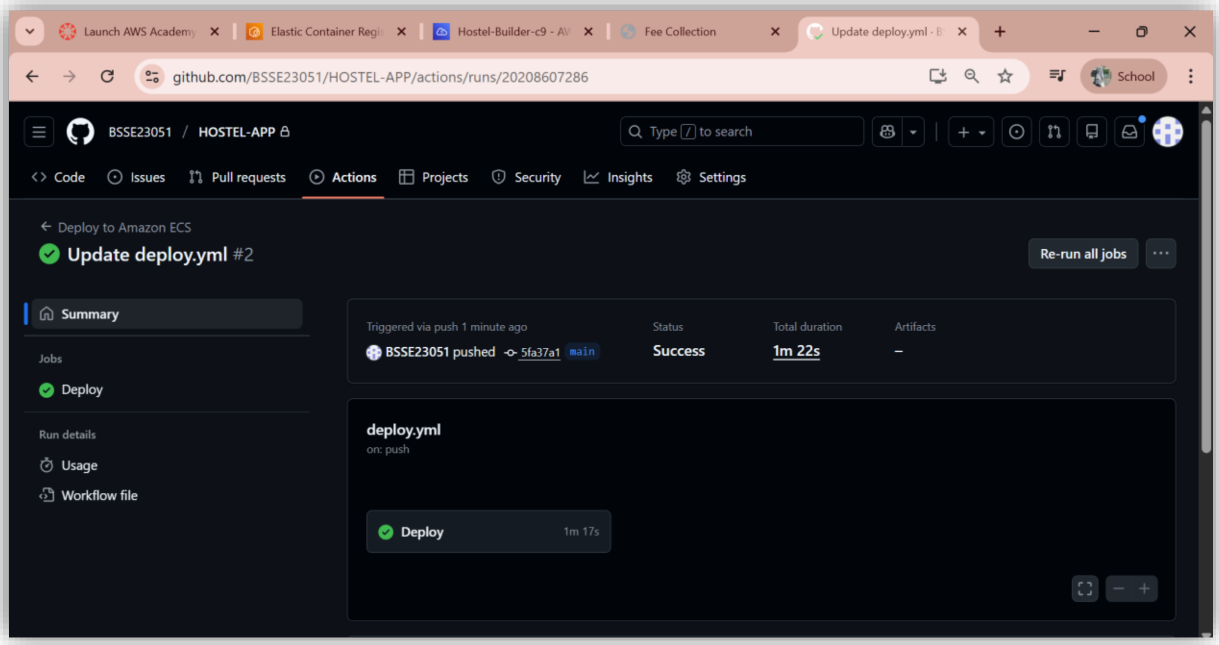
```python
42  def create_tables():
```

```
PS C:\Users\hh\Desktop\HMS> docker build -t hostel-app .
[+] Building 204.1s (12/12) FINISHED                                   docker:desktop-linux
 => [internal] load build definition from Dockerfile                                  0.1s
 => => transferring dockerfile: 800B                                                  0.0s
 => [internal] load metadata for docker.io/library/python:3.12-slim                   3.7s
 => [auth] library/python:pull token for registry-1.docker.io                         0.0s
 => [internal] load .dockerignore                                                     0.0s
 => => transferring context: 140B                                                     0.0s
 => [1/6] FROM docker.io/library/python:3.12-slim@sha256:590cad70271b6c1795c6a11fb5c1  8.6s
 => => resolve docker.io/library/python:3.12-slim@sha256:590cad70271b6c1795c6a11fb5c1  0.1s
 => => sha256:1f384a3df5003cc3a739008d6e3c2b2afc752887e9ce09757747c0bbb6e 250B / 250B  0.9s
 => => sha256:89933f78055059f29cf8b5a9c0b6df0fe9d96c388b99215881bf6 12.11MB / 12.11MB  7.0s
 => => sha256:dfff024aded812f05863f68d31b04030038e01017329961ea2d5f37 1.29MB / 1.29MB  2.1s
 => => extracting sha256:dfff024aded812f05863f68d31b04030038e01017329961ea2d5f37e6a1c  0.8s
 => => extracting sha256:89933f78055059f29cf8b5a9c0b6df0fe9d96c388b99215881bf653ed6f1  1.3s
 => => extracting sha256:1f384a3df5003cc3a739008d6e3c2b2afc752887e9ce09757747c0bbb6e6  0.0s
 => [internal] load build context                                                     0.1s
 => => transferring context: 4.24kB                                                   0.0s
 => [2/6] WORKDIR /app                                                                0.1s
 => [3/6] RUN apt-get update && apt-get install -y     build-essential     libpq-dev  69.0s
 => [4/6] COPY requirements.txt .                                                     0.2s
 => [5/6] RUN pip install --no-cache-dir -r requirements.txt                         57.2s
 => [6/6] COPY . .                                                                    0.5s
 => exporting to image                                                               64.0s
```



```
 => => exporting attestation manifest sha256:10432ed21c162c906f42a58fb2fd1afcd1dc1359  0.1s
 => => exporting manifest list sha256:8fe00aadb02ec8ad82783f8ddc352fac2d0c0df92cb589a  0.0s
 => => naming to docker.io/library/hostel-app:latest                                   0.0s
 => => unpacking to docker.io/library/hostel-app:latest                                0.6s

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/jhjivlr71luvcxtdegnfwqbro
PS C:\Users\hh\Desktop\HMS> docker tag hostel-app:latest 891377334911.dkr.ecr.us-east-1.amazonaws.com/hostel
-app
PS C:\Users\hh\Desktop\HMS> docker push 891377334911.dkr.ecr.us-east-1.amazonaws.com/hostel-app
Using default tag: latest
The push refers to repository [891377334911.dkr.ecr.us-east-1.amazonaws.com/hostel-app]
bfc218516263: Layer already exists
89933f780550: Layer already exists
1f384a3df500: Layer already exists
3465ff059709: Layer already exists
46133d9ae53b: Pushed
1733a4cd5954: Layer already exists
dfff024aded8: Layer already exists
a995e770a6c7: Layer already exists
fbbaaf986c62: Layer already exists
latest: digest: sha256:8fe00aadb02ec8ad82783f8ddc352fac2d0c0df92cb589a0df22f8f3f1baa09a size: 856
PS C:\Users\hh\Desktop\HMS> 
```



github.com/BSSE23051/HOSTEL-APP/actions/runs/20208607286

BSSE23051 / HOSTEL-APP

← Deploy to Amazon ECS

Update deploy.yml #2

Triggered via push 1 minute ago: BSSE23051 pushed 5fa37a1 main

Status: Success

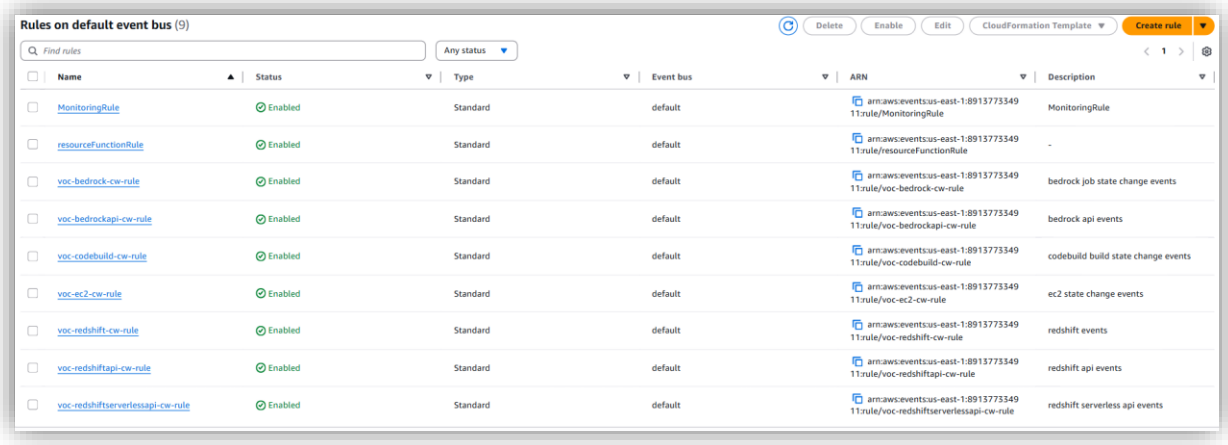
Total duration: 1m 22s

Artifacts: –

deploy.yml
on: push

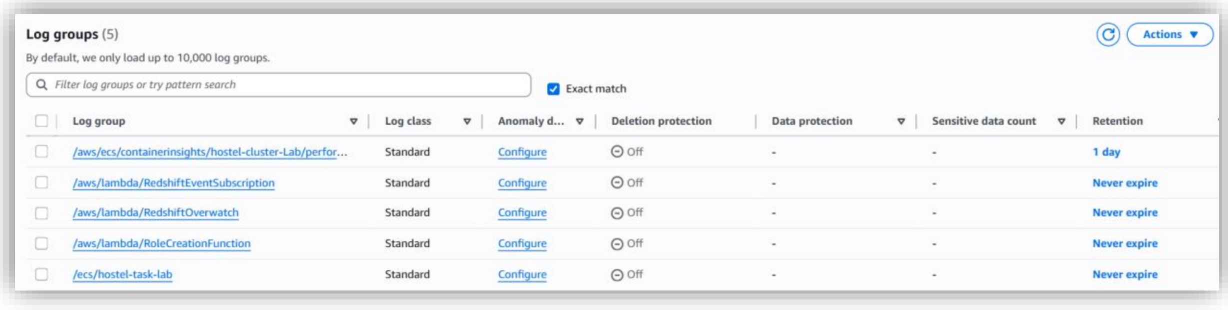
Deploy 1m 17s

Monitoring / logging (CloudWatch)



Rules on default event bus (9)

Name	Status	Type	Event bus	ARN	Description
MonitoringRule	Enabled	Standard	default	arn:aws:events:us-east-1:891377334911:rule/MonitoringRule	MonitoringRule
resourceFunctionRule	Enabled	Standard	default	arn:aws:events:us-east-1:891377334911:rule/resourceFunctionRule	-
voc-bedrock-cw-rule	Enabled	Standard	default	arn:aws:events:us-east-1:891377334911:rule/voc-bedrock-cw-rule	bedrock job state change events
voc-bedrockapi-cw-rule	Enabled	Standard	default	arn:aws:events:us-east-1:891377334911:rule/voc-bedrockapi-cw-rule	bedrock api events
voc-codebuild-cw-rule	Enabled	Standard	default	arn:aws:events:us-east-1:891377334911:rule/voc-codebuild-cw-rule	codebuild build state change events
voc-ec2-cw-rule	Enabled	Standard	default	arn:aws:events:us-east-1:891377334911:rule/voc-ec2-cw-rule	ec2 state change events
voc-redshift-cw-rule	Enabled	Standard	default	arn:aws:events:us-east-1:891377334911:rule/voc-redshift-cw-rule	redshift events
voc-redshiftapi-cw-rule	Enabled	Standard	default	arn:aws:events:us-east-1:891377334911:rule/voc-redshiftapi-cw-rule	redshift api events
voc-redshiftserverlessapi-cw-rule	Enabled	Standard	default	arn:aws:events:us-east-1:891377334911:rule/voc-redshiftserverlessapi-cw-rule	redshift serverless api events



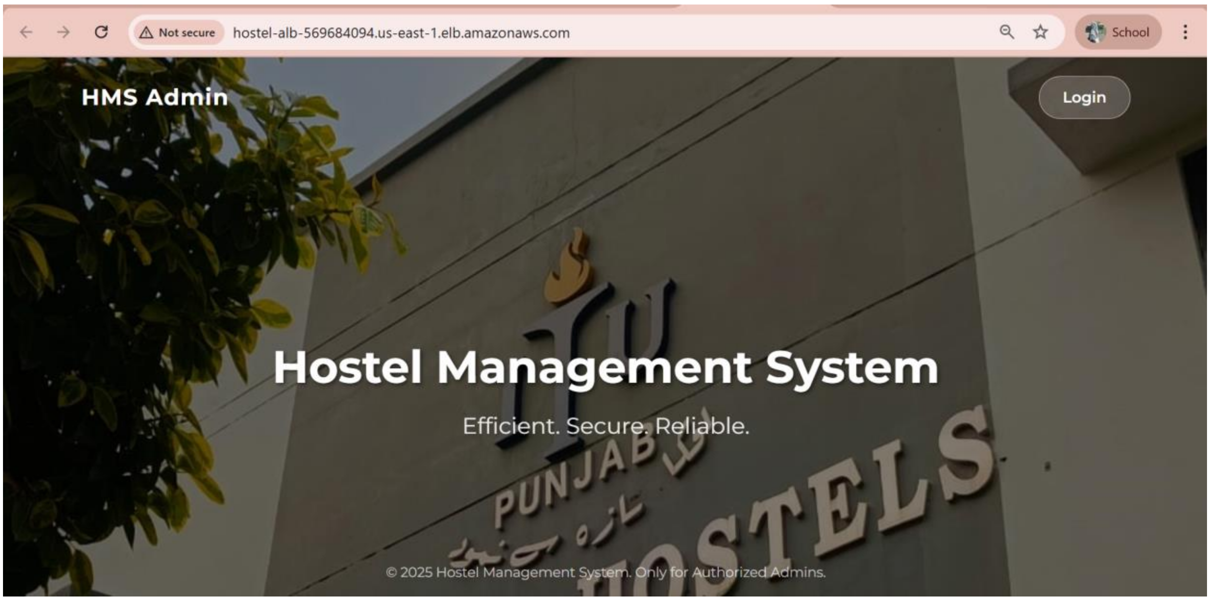
Log groups (5)

By default, we only load up to 10,000 log groups.

☑ Exact match

Log group	Log class	Anomaly d...	Deletion protection	Data protection	Sensitive data count	Retention
/aws/ecs/containerinsights/hostel-cluster-Lab/perfor...	Standard	Configure	Off	-	-	1 day
/aws/lambda/RedshiftEventSubscription	Standard	Configure	Off	-	-	Never expire
/aws/lambda/RedshiftOverwatch	Standard	Configure	Off	-	-	Never expire
/aws/lambda/RoleCreationFunction	Standard	Configure	Off	-	-	Never expire
/ecs/hostel-task-lab	Standard	Configure	Off	-	-	Never expire

Final output



Hostel Management System

Efficient. Secure. Reliable.

© 2025 Hostel Management System. Only for Authorized Admins.

List of Figures:

List of Tables:

References:

Adzic, G., & Chatley, R. (2017). Serverless computing: Economic and architectural impact. *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*, *11*(1), 884–889. https://doi.org/10.1145/3106237.3117767

Di Francesco, P., Malavolta, I., & Lago, P. (2019). Architecting microservices: A systematic mapping study. *Journal of Systems and Software*, *150*(1), 77–97. https://doi.org/10.1016/j.jss.2019.01.001

Merkel, D. (2014). Docker: Lightweight Linux containers for consistent development and deployment. *Linux Journal*, *2014*(239), 2–2. https://dl.acm.org/doi/10.5555/2600239.2600241

Shahin, M., Babar, M. A., & Zhu, L. (2017). Continuous integration, delivery and deployment: A systematic review on approaches, tools, challenges and practices. *IEEE Access*, *5*(1), 3909–3943. https://doi.org/10.1109/ACCESS.2017.2685629

Varghese, B., & Buyya, R. (2018). Next generation cloud computing: New trends and research directions. *Future Generation Computer Systems*, *79*(1), 849–861. https://doi.org/10.1016/j.future.2017.09.020