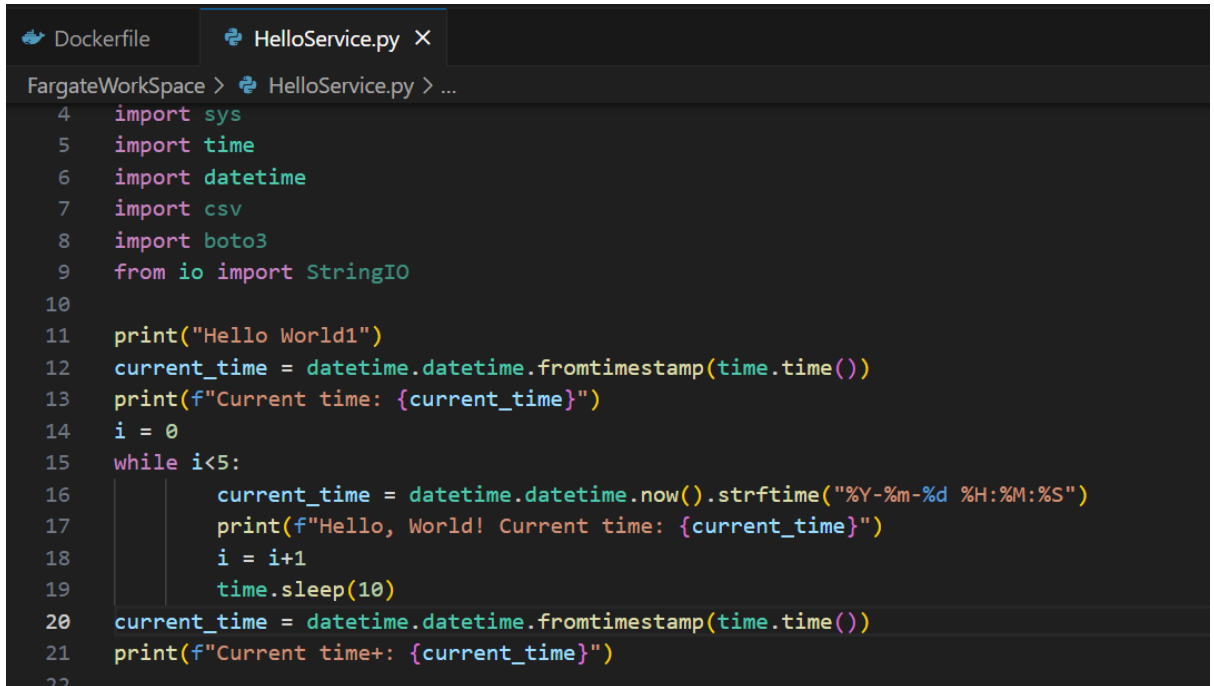
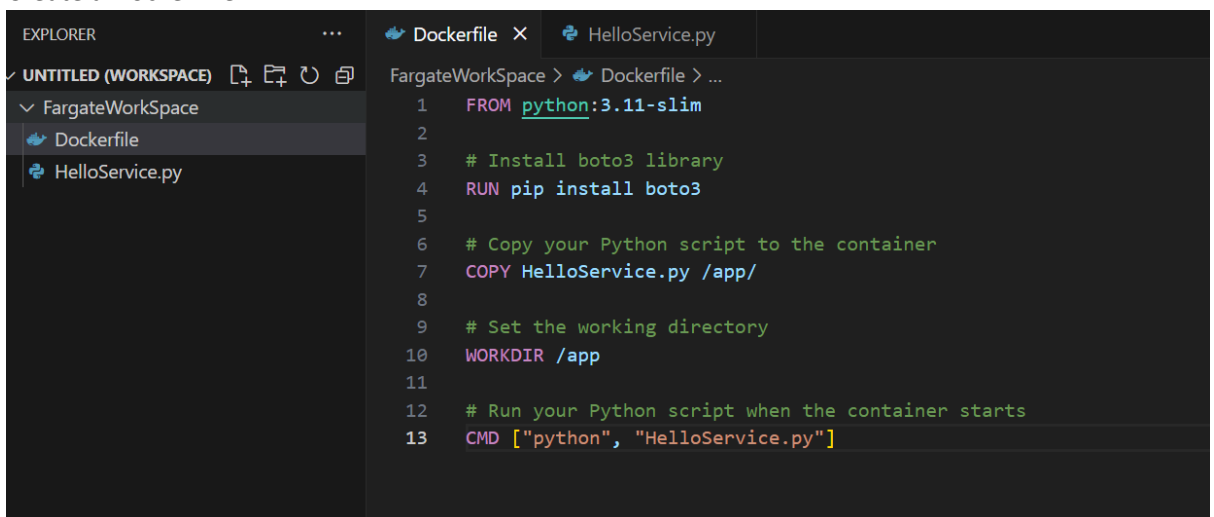


1. Create a python file which will work as dummy service



```
FargateWorkSpace > HelloService.py > ...
4  import sys
5  import time
6  import datetime
7  import csv
8  import boto3
9  from io import StringIO
10
11  print("Hello World1")
12  current_time = datetime.datetime.fromtimestamp(time.time())
13  print(f"Current time: {current_time}")
14  i = 0
15  while i<5:
16      current_time = datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")
17      print(f"Hello, World! Current time: {current_time}")
18      i = i+1
19      time.sleep(10)
20  current_time = datetime.datetime.fromtimestamp(time.time())
21  print(f"Current time+: {current_time}")
22
```

2. Create a DockerFile



```
EXPLORER
├── FargateWorkSpace
│   ├── Dockerfile
│   └── HelloService.py
└── ...

FargateWorkSpace > Dockerfile > ...
1  FROM python:3.11-slim
2
3  # Install boto3 library
4  RUN pip install boto3
5
6  # Copy your Python script to the container
7  COPY HelloService.py /app/
8
9  # Set the working directory
10 WORKDIR /app
11
12 # Run your Python script when the container starts
13 CMD ["python", "HelloService.py"]
```

3. Create an ECR repository, and access key for access key and password
Use the following commands for docker build and push to ECR repository

```
docker build -t helloservicefargate .
```

```
docker run helloservicefargate
```

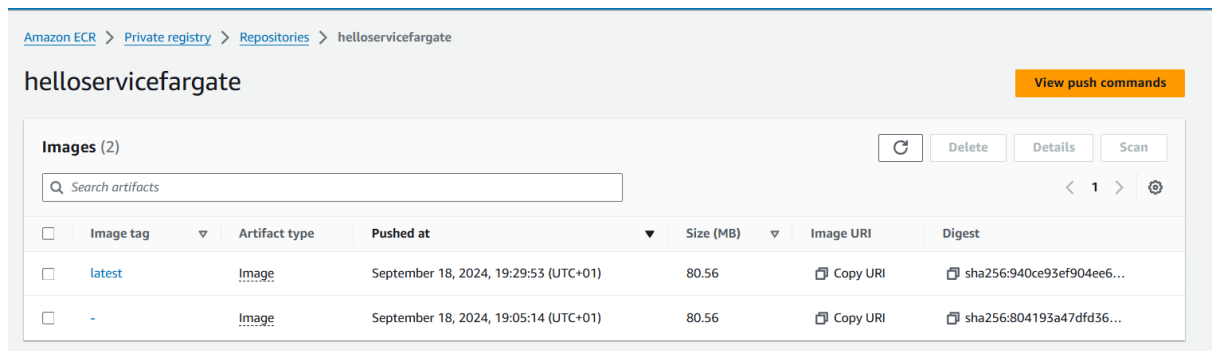
```
aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin <ecr repository>
```

```
docker build -t helloservicefargate .
```

```
docker tag helloservicefargate:latest <ECR repository URI>
```

```
docker push <ECR repository URI>/helloservicefargate:latest
```

-
-
-
4. The image is uploaded in ECR repository which will be used to get non-production environment services



-
-
-
-
5. Creation of ECS Cluster with Fargate capacity
6. Create a ECS Task definition with ECR image created

CPU: .25 vCPU | Memory: .5 GB

▼ Task roles - conditional

Task role | Info
A task IAM role allows containers in the task to make API requests to AWS services. You can create a task IAM role from the [IAM console](#).

ecsTaskExecutionRole

Task execution role | Info
A task execution IAM role is used by the container agent to make AWS API requests on your behalf. If you don't already have a task execution IAM role created, we can create one for you.

ecsTaskExecutionRole

▼ Task placement - optional

Task placement constraints are not supported for AWS Fargate launch type.

▼ Container - 1 | Info

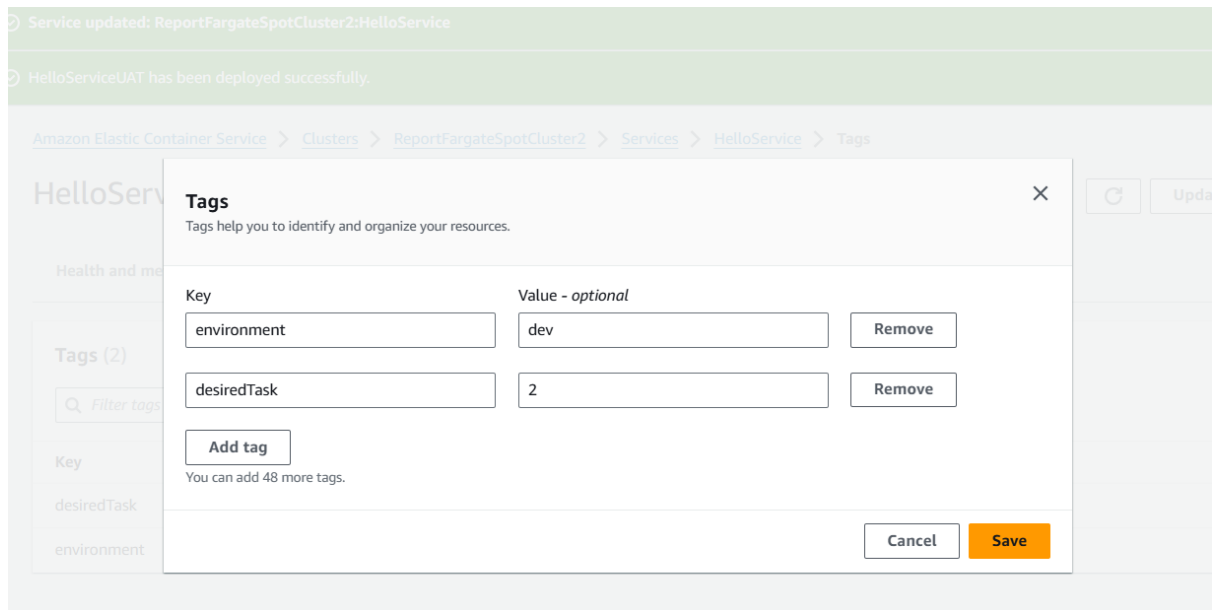
Essential container | Remove

Container details
Specify a name, container image, and whether the container should be marked as essential. Each task definition must have at least one essential container.

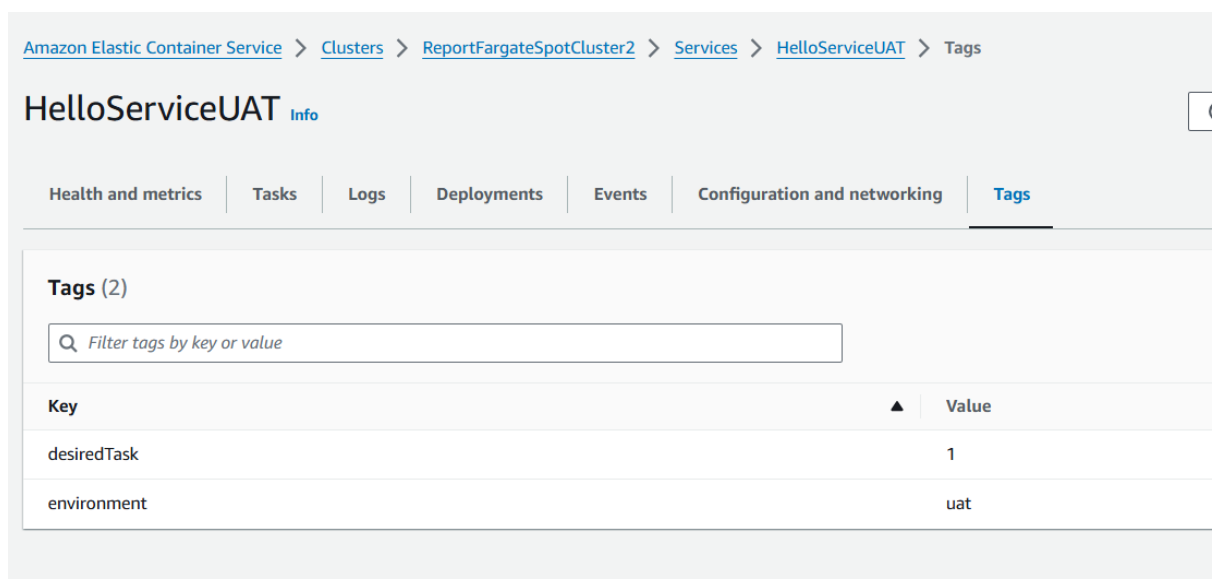
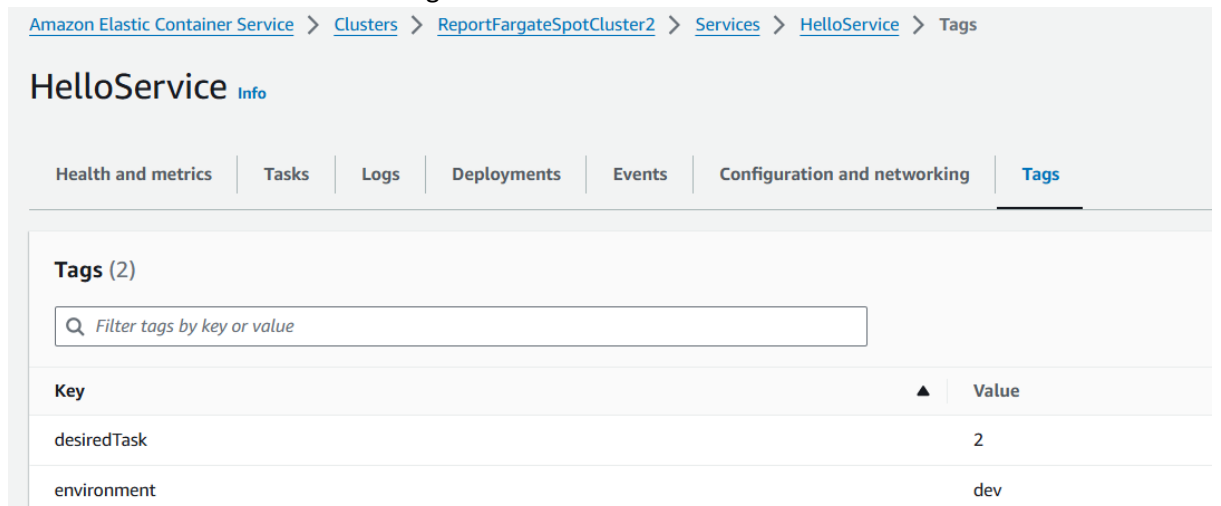
Name	Image URI	Essential container
HelloServiceFargate	820242904343.dkr.ecr.us-east-1.amazonaws.com/hello-service-fargate:latest	Yes

Un to 255 letters (uppercase and lowercase). | Un to 255 letters (uppercase and lowercase), numbers, hyphens, underscores, colons, periods.

-
-
-
-
-
7. Creation of ECS service Dev with tags



8. Creation of ECS Service UAT with tags



9. Write the code which will be invoked from EventBridge

10. Add EventBridge schedule for Start

Occurrence [Info](#)

You can define an one-time or recurrent schedule.

☐ One-time schedule

☒ Recurring schedule

Time zone

The time zone for the schedule.

(UTC+01:00) Europe/London

Schedule type

Choose the schedule type that best meets your needs.

☒ Cron-based schedule

A schedule set using a cron expression that runs at a specific time, such as 8:00 a.m. PST on the first Monday of every month.

☐ Rate-based schedule

A schedule that runs at a regular rate, such as every 10 minutes.

Cron expression [Info](#)

Define the cron expression for the schedule

Copy

Clear

cron (

00

08

?

*

1-5

2024

)

Minutes

Hours

Day of month

Month

Day of the week

Year

Next 10 trigger dates

Date and time are displayed in your current time zone in UTC format, e.g. "Wed, Nov 9, 2022 09:00 (UTC - 08:00)" for Pacific time

Sun, 22 Sep 2024 08:00:00 (UTC+01:00)

Mon, 23 Sep 2024 08:00:00 (UTC+01:00)

Tue, 24 Sep 2024 08:00:00 (UTC+01:00)

Wed, 25 Sep 2024 08:00:00 (UTC+01:00)

Thu, 26 Sep 2024 08:00:00 (UTC+01:00)

Sun, 29 Sep 2024 08:00:00 (UTC+01:00)

Mon, 30 Sep 2024 08:00:00 (UTC+01:00)

Tue, 01 Oct 2024 08:00:00 (UTC+01:00)

Wed, 02 Oct 2024 08:00:00 (UTC+01:00)

11. Add EventBridge schedule for Stop

Occurrence

Info

You can define an one-time or recurrent schedule.

☐ One-time schedule

☒ Recurring schedule

Time zone

The time zone for the schedule.

(UTC+01:00) Europe/London

Schedule type

Choose the schedule type that best meets your needs.

☒ Cron-based schedule

A schedule set using a cron expression that runs at a specific time, such as 8:00 a.m. PST on the first Monday of every month.

☐ Rate-based schedule

A schedule that runs at a regular rate, such as every 10 minutes.

Cron expression

Info

Define the cron expression for the schedule

Copy

Clear

cron (

30

17

?

*

1-5

2024

)

Minutes

Hours

Day of month

Month

Day of the week

Year

Next 10 trigger dates

Date and time are displayed in your current time zone in UTC format, e.g. "Wed, Nov 9, 2022 09:00 (UTC - 08:00)" for Pacific time

Thu, 19 Sep 2024 17:30:00 (UTC+01:00)

Sun, 22 Sep 2024 17:30:00 (UTC+01:00)

Mon, 23 Sep 2024 17:30:00 (UTC+01:00)

Tue, 24 Sep 2024 17:30:00 (UTC+01:00)

Wed, 25 Sep 2024 17:30:00 (UTC+01:00)

Thu, 26 Sep 2024 17:30:00 (UTC+01:00)

Sun, 29 Sep 2024 17:30:00 (UTC+01:00)

Mon, 30 Sep 2024 17:30:00 (UTC+01:00)

Tue, 01 Oct 2024 17:30:00 (UTC+01:00)

Wed, 02 Oct 2024 17:30:00 (UTC+01:00)

12. Update the Payload with required values

Invoke

AWS Lambda

Universal target definition

Lambda function

ECSFargateScheduler

Create new Lambda function

Configure version/alias

Payload

The JSON that you want to provide to your Lambda function as input. For example, --payload '{ key: value }'. [Learn more](#)

```
1 {
2   "environment": {
3     "variables": {
4       "environment": "dev",
5       "exclusionList": "item1,item2,item3",
6       "clusterName": "ReportFargateSpotCluster2",
7       "action": "Start"
8     }
9   }
10 }
```

13. Attach the policy to Lambda execution role

ECSFargateScheduler-role-yj4g87a4 [Info](#) [Delete](#)

Summary [Edit](#)

Creation date
September 17, 2024, 16:26 (UTC+01:00)

ARN
[am:aws:iam::820242904343:role/service-role/ECSFargateScheduler-role-yj4g87a4](#)

Last activity
-

Maximum session duration
1 hour

[Permissions](#) [Trust relationships](#) [Tags](#) [Access Advisor](#) [Revoke sessions](#)

Permissions policies (2) [Info](#) [Refresh](#) [Simulate](#) [Remove](#) [Add permissions](#)

You can attach up to 10 managed policies.

Search

Filter by Type
All types

< 1 > [Settings](#)

<input type="checkbox"/>	Policy name Info	Type	Attached entities
<input type="checkbox"/>	AmazonEC2_FullAccess	AWS managed	4
<input type="checkbox"/>	AWSLambdaBasicExecutionRole-98a19711-ac61...	Customer managed	1

14. The lambda is executed from EventBridge

Dev service is started and UAT not started

Services | Tasks | Infrastructure | Metrics | Scheduled tasks | Tags

Services (2) Info

Filter launch type: Any launch type

Filter service type: Any service type

<input type="checkbox"/>	Service name	ARN	Status	Service t...	Deployments and tasks	Last de
<input type="checkbox"/>	HelloService	arn:aws:ecs:us-east-1:820242904...	Active	REPLICA	1/2 Tasks running	Com
<input type="checkbox"/>	HelloServiceUAT	arn:aws:ecs:us-east-1:820242904...	Active	REPLICA	0/0 Tasks running	Com

15. Task draining after stop executed from EventBridge scheduler payload

Services | Tasks | Infrastructure | Metrics | Scheduled tasks | Tags

Services (2) Info

Filter launch type: Any launch type

Filter service type: Any service type

<input type="checkbox"/>	Service name	ARN	Status	Service t...	Deployments and tasks	Last de
<input type="checkbox"/>	HelloService	arn:aws:ecs:us-east-1:820242904...	Active	REPLICA	1/0 Tasks running	Com
<input type="checkbox"/>	HelloServiceUAT	arn:aws:ecs:us-east-1:820242904...	Active	REPLICA	0/0 Tasks running	Com