

CT-2017-067

M.H. HISHAN PASINDU UMayANGA


## TASK-1

- a) In the git hub setup user name and email .

```
FCT-LB02-49+FCT@FCT-LB02-49 MINGW64 ~/Desktop/CT-2017-067
$ git config --global user.name "Hishan067"

FCT-LB02-49+FCT@FCT-LB02-49 MINGW64 ~/Desktop/CT-2017-067
$ git config --global user.email "Hishanumayanga@gmail.com"
```

Fork the given repository

 kasun-gamage-www / ds-exam

Q Type to search

ode Issues Pull requests Actions Projects Security Insights


### Create a new fork

A *fork* is a copy of a repository. Forking a repository allows you to freely experiment with changes without affecting the original project. [View existing forks.](#)


*Required fields are marked with an asterisk (\*).*

Owner \*

Repository name \*

 hishan-umayanga

/ ds-exam

 ds-exam is available.


By default, forks are named the same as their upstream repository. You can customize the name to distinguish it further.

Description (optional)

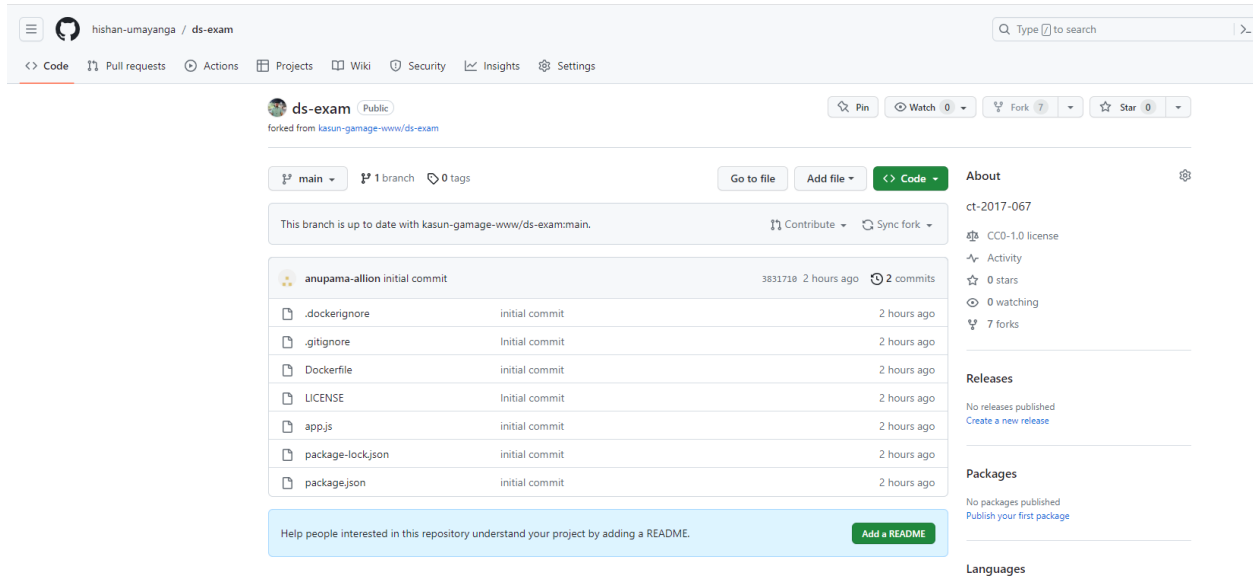
ct-2017-067

☒ Copy the `main` branch only

Contribute back to kasun-gamage-www/ds-exam by adding your own branch. [Learn more.](#)

 You are creating a fork in your personal account.

Create fork



Clone your forked repository to your local machine:

```
FCT-LB02-49+FCT@FCT-LB02-49 MINGW64 ~/Desktop/CT-2017-067
$ git clone https://github.com/kasun-gamage-www/ds-exam.git
Cloning into 'ds-exam'...
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 11 (delta 0), reused 7 (delta 0), pack-reused 0
Receiving objects: 100% (11/11), 11.34 KiB | 11.34 MiB/s, done.

FCT-LB02-49+FCT@FCT-LB02-49 MINGW64 ~/Desktop/CT-2017-067
$ ls
Capture.JPG  ct-2017-067.docx  '~$-2017-067.docx'
```

- b)
- c) created a deploy.yml file

Summary

Jobs

Build Image

Run details

Usage

Workflow file

```
1 name: Deploy to ECR
2
3 on:
4
5 push:
6   branches: [ main ]
7 pull_request:
8   branches: [ main ]
9
10 jobs:
11
12   build:
13
14     name: Build Image
15     runs-on: ubuntu-latest
16
17     steps:
18
19       - name: Check out code
20         uses: actions/checkout@v2
21
22       - name: Configure AWS credentials
23         uses: aws-actions/configure-aws-credentials@v1
24         with:
25           aws-access-key-id: ${ secrets.AWS_ACCESS_KEY_ID }
26           aws-secret-access-key: ${ secrets.AWS_SECRET_ACCESS_KEY }
27           aws-region: eu-west-2
28
29       - name: Login to Amazon ECR
30         id: login-ecr
31         uses: aws-actions/amazon-ecr-login@v1
32         with:
33           mask-password: 'true'
34
35       - name: Build, tag, and push image to Amazon ECR
36         env:
37           ECR_REGISTRY: ${ steps.login-ecr.outputs.registry }
38           ECR_REPOSITORY: ds-exam-067
39           IMAGE_TAG: ecs-example-app
40         run: |
41           docker build -t $ECR_REGISTRY/$ECR_REPOSITORY:$IMAGE_TAG .
42           docker push $ECR_REGISTRY/$ECR_REPOSITORY:$IMAGE_TAG
43
```

Added the AWS Access key & Secret access key to the repository settings:

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Webhooks

Environments

Codespaces

Pages

Security

Code security and analysis

Deploy keys

Secrets and variables

Actions

Codespaces

Dependabot

Integrations

Actions secrets and variables

Secrets and variables allow you to manage reusable configuration data. Secrets are encrypted and are used for sensitive data. Variables are shown as plain text and are used for non-sensitive data.

Anyone with collaborator access to this repository can use these secrets and variables for actions. They are not passed to workflows that are triggered by a pull request from a fork.

Secrets

Variables

New repository secret

Environment secrets

Manage environments

There are no secrets for this repository's environments.

Repository secrets

AWS\_ACCESS\_KEY\_ID

Updated 3 minutes ago

AWS\_SECRET\_ACCESS\_KEY

Updated 2 minutes ago

Activate Wi  
Go to Settings 1

hishan-umayanga / ds-exam

<> Code Pull requests Actions Projects Wiki Security Insights Settings

← Deploy to ECR

✓ Update deploy.yml #3

Summary

Jobs

✓ Build Image

Run details

Usage

Workflow file

### Build Image

succeeded 3 minutes ago in 1m 57s

- > ✓ Set up job
- > ✓ Check out code
- > ✓ Configure AWS credentials
- > ✓ Login to Amazon ECR
- > ✓ Build, tag, and push image to Amazon ECR
- > ✓ Post Login to Amazon ECR
- > ✓ Post Configure AWS credentials
- > ✓ Post Check out code
- > ✓ Complete job

Amazon Elastic Container Registry

Private registry  
Public registry  
Repositories  
Summary  
Images  
Permissions  
Lifecycle Policy  
Repository tags

ECR public gallery  
Amazon ECS

Amazon ECR > Repositories > ds-exam-067

ds-exam-067

View push commands Edit

Images (1)

Search artifacts

<input type="checkbox"/>	Image tag	Artifact type	Pushed at	Size (MB)	Image URI	Digest	Scan status	Vulnerabilities
<input type="checkbox"/>	ecs-example-app	Image	September 02, 2023, 14:52:48 (UTC+05:5)	350.92	Copy URI	sha256:545a9ae224394320b947a79c2b35a...	-	-

Login to amazon ECR using aws CLI:

```
C:\Users\FCT>aws configure
AWS Access Key ID [None]: AKIAJ3SBGA4HOW6564R
AWS Secret Access Key [None]: 9UyWQ/Whesd0a7rR5PNBqSLViJ84UBUHOKuRjNxp
Default region name [None]: eu-west-2
Default output format [None]:
```

Create iam user account:

Hishan-067 [Info](#)

## Summary


## ARN

 arn:aws:iam::502213455928:user/Hishan-067

## Created

September 02, 2023, 13:56 (UTC+05:30)

## Console access

 Enabled without MFA

## Last console sign-in

 Never

## Access key 1

[Create access key](#)Retrieve access keys [Info](#)

## Access key

If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

## Access key

 AKIAXJ3SBGA4HOW6564R

## Secret access key

 \*\*\*\*\* [Show](#)

## Summary

## ARN

 arn:aws:iam::502213455928:user/Hishan-067

## Created

September 02, 2023, 13:56 (UTC+05:30)

## Console access

 Enabled without MFA

## Last console sign-in

 Never

## Access key 1

AKIAXJ3SBGA4HOW6564R - Active

 Used today. Created today.

## Access key 2

[Create access key](#)[Permissions](#) [Groups](#) [Tags](#) [Security credentials](#) [Access Advisor](#)

## Permissions policies (5)













Permissions are defined by policies attached to the user directly or through groups.

[Remove](#)[Add permissions](#) ▼

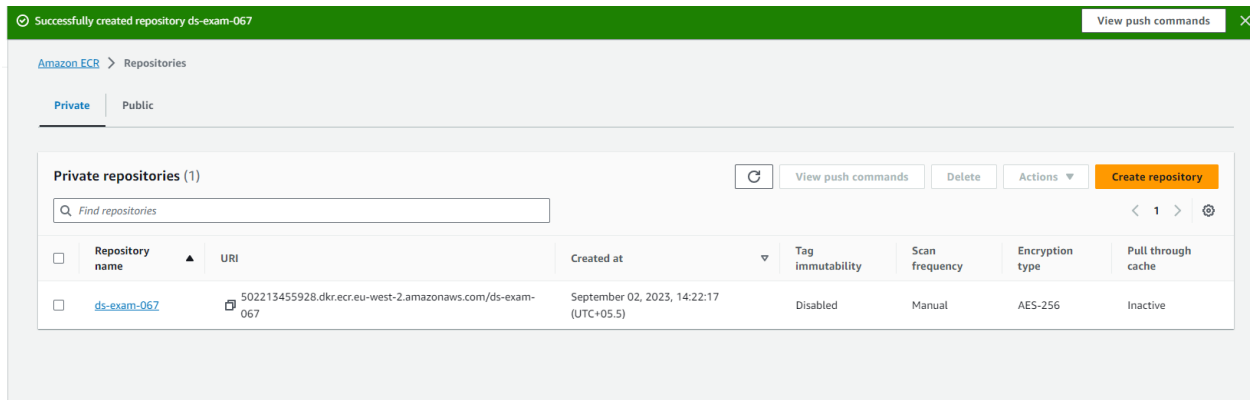
Filter by Type

All types ▼

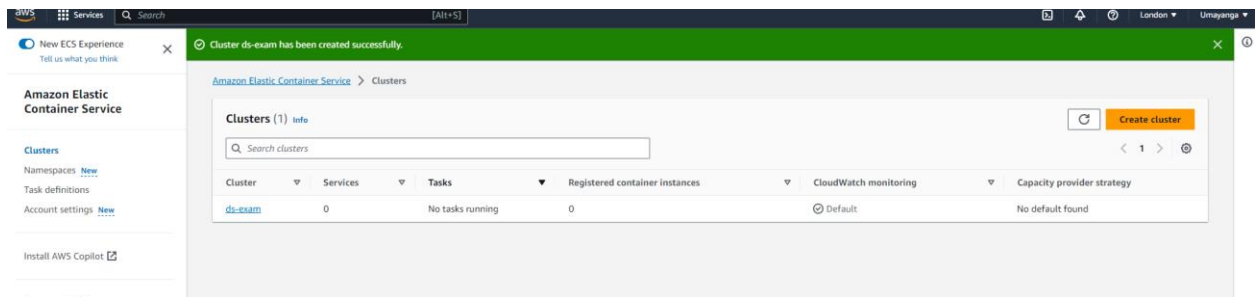
&lt; 1 &gt; ⓘ

<input type="checkbox"/>	Policy name 	Type	Attached via 
<input type="checkbox"/>	  <a href="#">AmazonEC2ContainerRegistryFullAccess</a>	AWS managed	Directly
<input type="checkbox"/>	  <a href="#">AmazonECS_FullAccess</a>	AWS managed	Directly
<input type="checkbox"/>	  <a href="#">AWSAppRunnerServicePolicyForECRAccess</a>	AWS managed	Directly
<input type="checkbox"/>	  <a href="#">EC2InstanceProfileForImageBuilderECRContainerBuilds</a>	AWS managed	Directly
<input type="checkbox"/>	  <a href="#">SecretsManagerReadWrite</a>	AWS managed	Directly

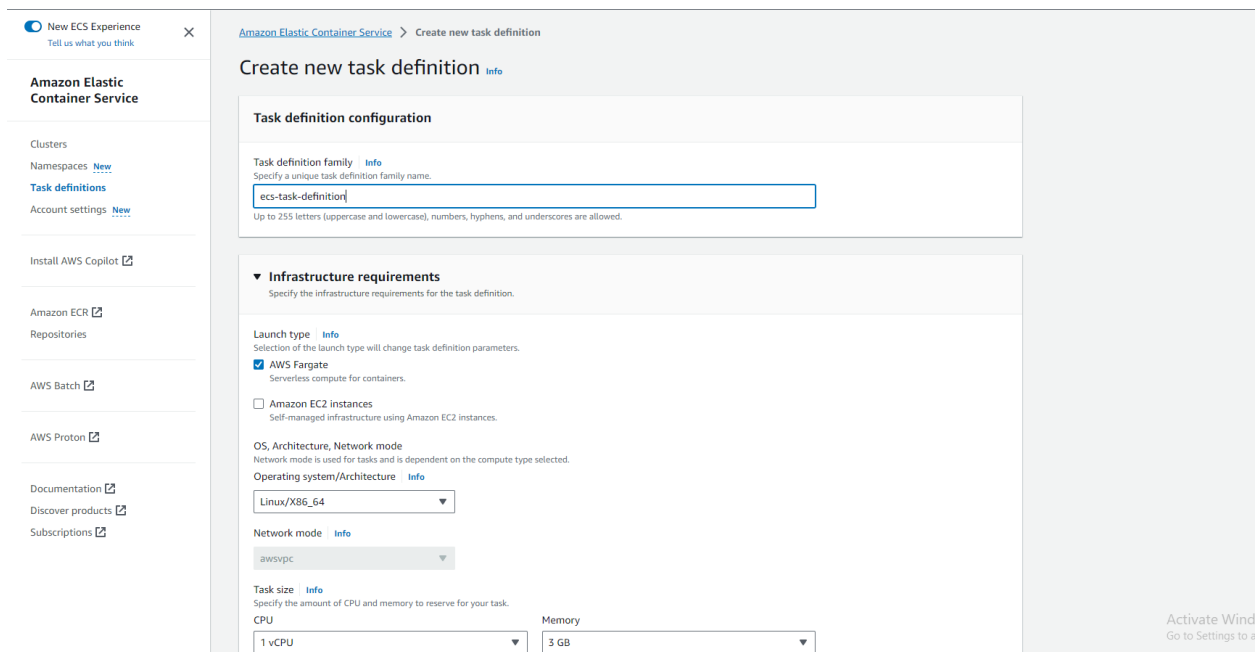
Created amazon ECR:



## Created Amazon Elastic Container Service:



## Created ECS task definition –



New ECS Experience  
Tell us what you think

Amazon Elastic Container Service

Clusters  
Namespaces [New](#)  
**Task definitions**  
Account settings [New](#)

Install AWS Copilot [↗](#)

Amazon ECR [↗](#)  
Repositories

AWS Batch [↗](#)

AWS Proton [↗](#)

Documentation [↗](#)  
Discover products [↗](#)  
Subscriptions [↗](#)

Task definition successfully created  
ecs-task-definition:1 has been successfully created. You can use this task definition to deploy a service or run a task.

Deploy ▼

Amazon Elastic Container Service > Task definitions > ecs-task-definition > Revision 1 > Containers

ecs-task-definition:1

Deploy ▼ Actions ▼ Create new revision ▼

Overview [Info](#)

ARN arn:aws:ecs:eu-west-2:502213455928:task-definition/ecs-task-definition:1	Status ACTIVE	Time created 2023-09-02T09:52:45.632Z	App environment FARGATE
Task role -	Task execution role ecsTaskExecutionRole <a href="#">↗</a>	Operating system/Architecture Linux/X86_64	Network mode awsvpc

Containers | JSON | Storage | Tags

Task size

Task CPU 1 vCPU	Task memory 3 GB
--------------------	---------------------

Containers [Info](#)

Container name	Image	Private registry	Essential	CPU	Memory hard/soft limit	GPU
ds-exam-067	502213455928.dkr.e...	-	Yes	0	-/-	

Activate Windows  
Go to Settings to activate Windows.

## Task verification-

New ECS Experience  
Tell us what you think

Amazon Elastic Container Service

Clusters  
Namespaces [New](#)  
**Task definitions**  
Account settings [New](#)

Install AWS Copilot [↗](#)

Amazon ECR [↗](#)  
Repositories

AWS Batch [↗](#)

AWS Proton [↗](#)

Documentation [↗](#)  
Discover products [↗](#)  
Subscriptions [↗](#)

Task definition successfully created  
ecs-task-definition:1 has been successfully created. You can use this task definition to deploy a service or run a task.

Deploy ▼

Amazon Elastic Container Service > Task definitions > ecs-task-definition

ecs-task-definition (1) [Info](#)

Filter task definition revisions by value Active task definitions

<input type="checkbox"/> Task definition: revision	Status
<input type="checkbox"/> ecs-task-definition:1	ACTIVE

New ECS Experience  
Tell us what you think

Amazon Elastic Container Service

Clusters  
Namespaces [New](#)  
Task definitions  
Account settings [New](#)

Install AWS Copilot [↗](#)

Amazon ECR [↗](#)  
Repositories

AWS Batch [↗](#)

AWS Proton [↗](#)

Documentation [↗](#)  
Discover products [↗](#)  
Subscriptions [↗](#)

Task definition successfully created  
ecs-task-definition:1 has been successfully created. You can use this task definition to deploy a service or run a task.

Deploy ▼

Amazon Elastic Container Service > Clusters > ds-exam > Create service

Create [Info](#)

Environment [AWS Fargate](#)

Existing cluster  
Select an existing cluster. To create a new cluster, go to [Clusters](#).

ds-exam

▼ Compute configuration (advanced)

Compute options [Info](#)  
To ensure task distribution across your compute types, use appropriate compute options.

☒ Capacity provider strategy  
Specify a launch strategy to distribute your tasks across one or more capacity providers.

☐ Launch type  
Launch tasks directly without the use of a capacity provider strategy.

Capacity provider strategy [Info](#)  
Select either your cluster default capacity provider strategy or select the custom option to configure a different strategy.

☐ Use cluster default

☒ Use custom (Advanced)

Capacity provider	Base	Weight
FARGATE ▼	0	1

Add more

New ECS Experience  
Tell us what you think

Amazon Elastic Container Service

Clusters  
Namespaces New  
Task definitions  
Account settings New

Install AWS Copilot [↗](#)

Amazon ECR [↗](#)  
Repositories

AWS Batch [↗](#)

AWS Proton [↗](#)

Documentation [↗](#)  
Discover products [↗](#)  
Subscriptions [↗](#)

Deployment configuration

Application type Info  
Specify what type of application you want to run.

☒ Service  
Launch a group of tasks handling a long-running computing work that can be stopped and restarted. For example, a web application.

☐ Task  
Launch a standalone task that runs and terminates. For example, a batch job.

Task definition  
Select an existing task definition. To create a new task definition, go to [Task definitions](#) [↗](#).

☐ Specify the revision manually  
Manually input the revision instead of choosing from the 100 most recent revisions for the selected task definition family.

Family  
ecs-task-definition

Revision  
1 (LATEST)

Service name  
Assign a unique name for this service.

demo-service

Service type Info  
Specify the service type that the service scheduler will follow.

☒ Replica  
Place and maintain a desired number of tasks across your cluster.

☐ Daemon  
Place and maintain one copy of your task on each container instance.

Desired tasks  
Specify the number of tasks to launch.

1

Deployment options

New ECS Experience  
Tell us what you think

Amazon Elastic Container Service

Clusters  
Namespaces New  
Task definitions  
Account settings New

Install AWS Copilot [↗](#)

Amazon ECR [↗](#)  
Repositories

AWS Batch [↗](#)

AWS Proton [↗](#)

Documentation [↗](#)  
Discover products [↗](#)  
Subscriptions [↗](#)

Task definition successfully created  
ecs-task-definition:1 has been successfully created. You can use this task definition to deploy a service or run a task.

Deploy [↗](#)

demo-service deployment is in progress. It takes a few minutes.

[View in CloudFormation](#) [↗](#)

Amazon Elastic Container Service > Clusters > ds-exam > Services

ds-exam

[↗](#) Update cluster Delete cluster

Cluster overview

ARN ds-exam	Status Active	CloudWatch monitoring Default	Registered container instances -
Services Draining -	Active 1	Tasks Pending -	Running 1

Services (1) Info

Filter services by value

All launch types

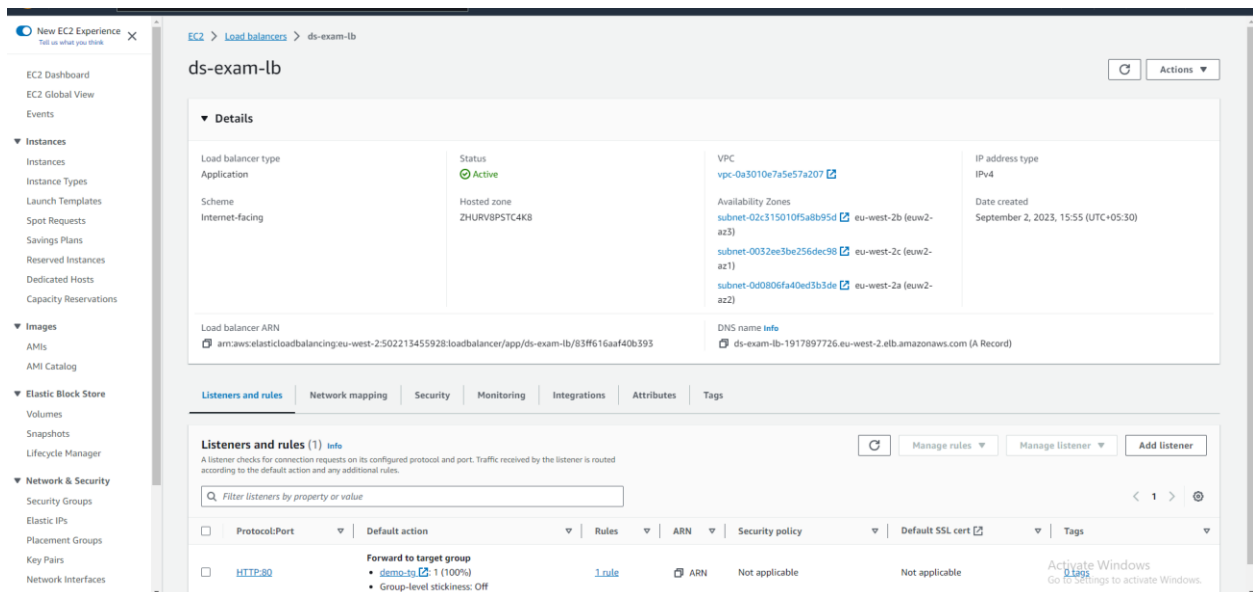
All service types

Manage tags Update Delete service Create

Service name	Status	ARN	Service type	Deployments and tasks	Last deploy...	Task defini...	Revision
demo-service	Active	arn:aws:ecs...	REPLICA	1/1 Tasks ru...	In progress	ecs-task-defi...	1

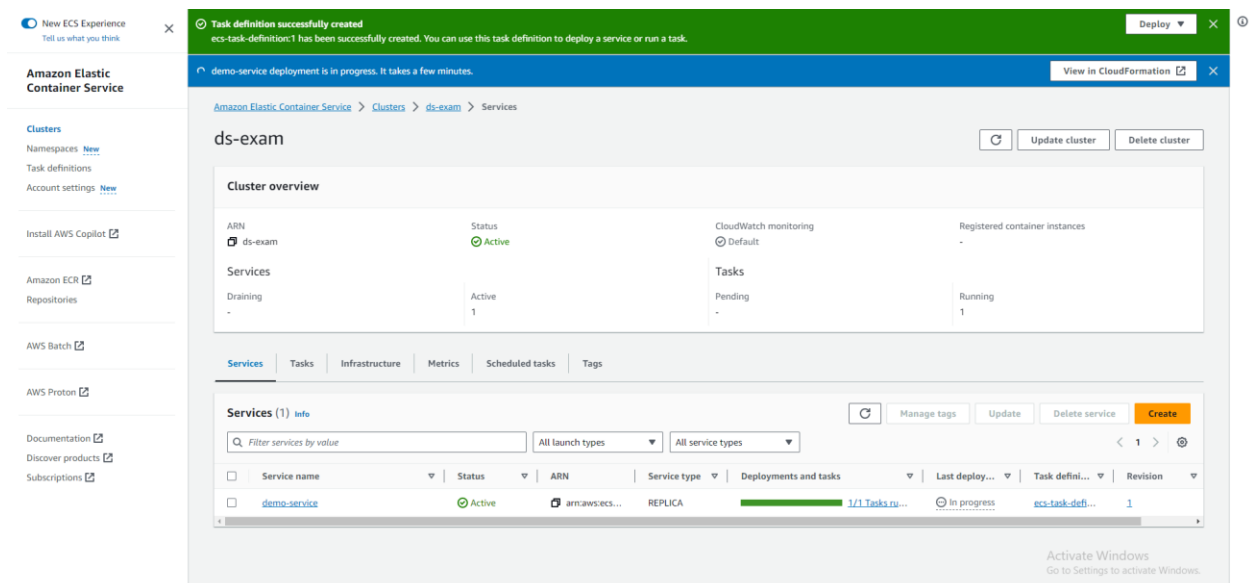
Activate Windows  
Go to Settings to activate Windows.





Verification:-

c)



g) commit verification file to github

```

FCT-LB02-49+FCT@FCT-LB02-49 MINGW64 ~/Desktop/CT-2017-067/ds-exam (main)
$ git commit -m "verification file"
[main 0bdefd3] verification file
1 file changed, 1 insertion(+)
create mode 100644 verification.txt

FCT-LB02-49+FCT@FCT-LB02-49 MINGW64 ~/Desktop/CT-2017-067/ds-exam (main)
$ git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 4 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 345 bytes | 345.00 KiB/s, done.
Total 3 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/hishan-umayanga/ds-exam.git
60d488b..0bdefd3  main -> main

FCT-LB02-49+FCT@FCT-LB02-49 MINGW64 ~/Desktop/CT-2017-067/ds-exam (main)
$

```

Github final-

hishan-umayanga / ds-exam

Q Type to search

+

<> Code

Pull requests

Actions

Projects

Wiki

Security

Insights

Settings

ds-exam

Public

forked from kasun-gamage-www/ds-exam

Pin

Watch 0

Fork 11

Star 0

main

1 branch

0 tags

Go to file

Add file

Code

This branch is 4 commits ahead of kasun-gamage-www:main.

Contribute

Sync fork

Hishan067 and Hishan067 verification file

0bdefd3 2 minutes ago 6 commits

.github/workflows	Update deploy.yml	1 hour ago
.dockerignore	initial commit	4 hours ago
.gitignore	initial commit	4 hours ago
Dockerfile	initial commit	4 hours ago
LICENSE	initial commit	4 hours ago
app.js	initial commit	4 hours ago
package-lock.json	initial commit	4 hours ago
package.json	initial commit	4 hours ago
verification.txt	verification file	2 minutes ago

Help people interested in this repository understand your project by adding a README.

Add a README

About

ct-2017-067

CC0-1.0 license

Activity

0 stars

0 watching

11 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Languages

Dockerfile 61.2%

JavaScript 38.8%