

## Task 2

Below is a public Github repository containing a simple Node.js web application.

<https://github.com/sngsweekeat/one2onetool>

The application makes use of a built-in JSON data file, whose filename can be specified in the "DATA\_FILE" environment variable. If the environment variable is not found, then it will default to using "Questions.json".

Fork the repository to your own Github account (or any other public Git server), and use it for the rest of the exercise.

Create a CI/CD pipeline (bash script or any CI/CD tool that you are **most comfortable** with) that monitors the "release" and "staging" branches of the repository. The pipeline should be triggered on new commits and perform **at least** the following:

1. The pipeline should build and run tests on the application
2. Containerise and deploy the application on a public cloud instance
3. There are two branches, "staging" and "release":
  - a. The "staging" branch should use "Questions-test.json" as its input data file
  - b. The "release" branch should use "Questions.json" as its input data file
4. Consider how versioning can be done to differentiate the builds

If any of the tasks fails at any point, the pipeline should be stopped and an email alert should be sent with relevant information.

Github Repo: <https://github.com/hanshengng/one2onetool/tree/staging>

I have tried using 3 solutions. First, I have tried using Codepipeline, build the docker image using codebuild and deploy in ECS. Secondly, I have tried another method, without containerising it, I have deployed in Elastic Beanstalk.

Due to large number for testing, my AWS account was messed up, therefore I have to clean-up everything and did a simple one.

Therefore, I used Codepipeline and connect with Github. I have provision a EC2 instance with group tag and added it in my deployment group tag. So when the deploy trigger, EC2 will know which instance to deploy in. Due to time constraint, the application is not up in public cloud.

I have tested the built the docker image in my local machine and it works with env variables (Staging: Questions\_test.json & Release: Questions.json)

## Testing Scenario

### 1) CI/CD using AWS CodePipeline (CodeBuild, CodeDeploy) **Containerised**

AWS Services used:

- CodePipeline
- CodeBuild (Create a **docker** image and push to **ECR** from Source repository: **GitHub**)
- CodeDeploy (Deploy the **ECR** latest image to **ECS**)
- Elastic Container Service
- Elastic Container Registry

Code Configuration files:

- Buildspec-ecr.yml (Build the image and push to ECR)
- Appsec.yml (Deploy the image to Fargate container)
- Dockerfile (Build the image from source code)
- Fargate.yml (Create a Fargate Cluster with Container)

### 2) CI/CD using AWS CodePipeline (CodeDeploy, Elastic BeanStalk)

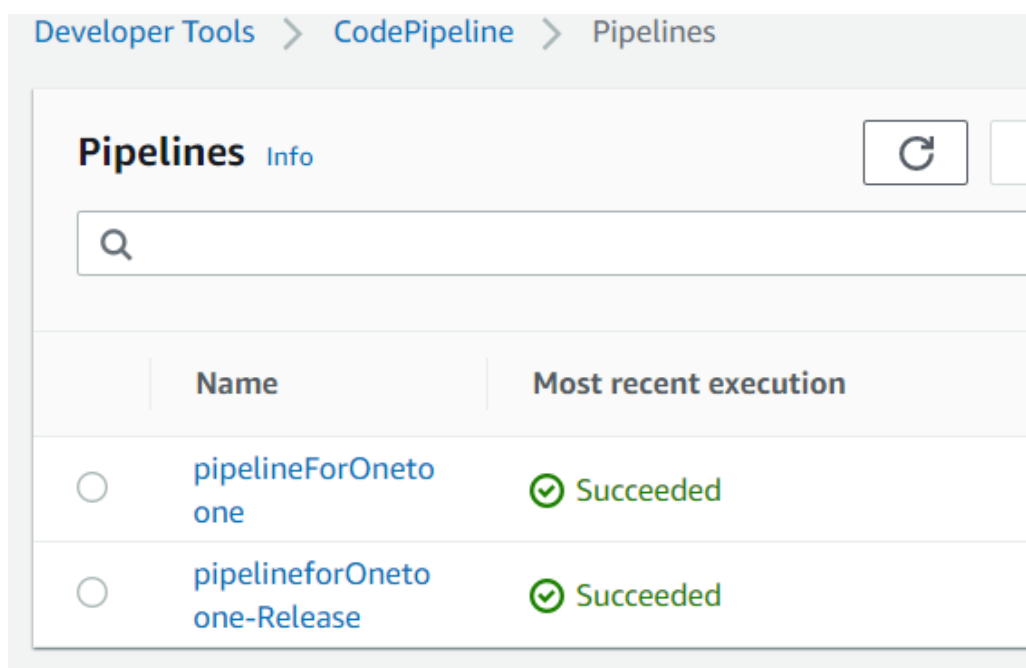
AWS Services used:

- CodeBuild (To create artifact from source code)
- CodeDeploy (Pull the artifact and deploy on Elastic BeanStalk server)
- CodePipeline
- Elastic BeanStalk

Code Configuration files:

- Buildspec-ebstalk.yml (Build the artifact and deploy to BeanStalk)

I have setup 2 CodePipeline connecting to my Github for Continuous Integration.



Once I have done a commit in my Git repository, CodePipeline will be triggered.

Developer Tools > CodePipeline > Pipelines > pipelineForOnetoone

## pipelineForOnetoone

[Notify](#) [Edit](#) [Stop execution](#) [Clone pipeline](#)

✓ **Source** Succeeded  
Pipeline execution ID: [4770d006-192b-481b-85d2-3845b812c02e](#)

Source ⓘ

[GitHub \(Version 2\)](#)

✓ **Succeeded** - 33 minutes ago  
[47fd8bbe](#)

[47fd8bbe](#) [Source: Update Dockerfile](#)

After source stage, build stage will trigger. Build stage will read buildspec.yml. Docker image will be pushed to ECR.

↓ [Disable transition](#)

✓ **Build** Succeeded  
Pipeline execution ID: [4770d006-192b-481b-85d2-3845b812c02e](#)

Build ⓘ

[AWS CodeBuild](#)

✓ **Succeeded** - 31 minutes ago  
[Details](#)

[47fd8bbe](#) [Source: Update Dockerfile](#)

Docker images will be stored in ECR.

ECR

>

Repositories

Private

Public

Private repositories (1)

View push commands

Delete

Edit

Create repository

Find repositories

<

1

>

|  | Repository name | URI   | Created at               | Tag immutability | Scan on push | Encryption type |
|--|-----------------|---|--------------------------|------------------|--------------|-----------------|
|  | one2onerepo     | 669189415304.dkr.ecr.ap-southeast-1.amazonaws.com/one2onerepo | Apr 13, 2021 10:57:12 AM | Disabled         | Disabled     | AES-256         |

ECR

>

Repositories

>

one2onerepo

one2onerepo

Images (34)

Find images

|  | Image tag  | Pushed at                 | Size (MB) | Image URI           | Digest                               |
|--|--|---------------------------|-----------|---------------------|--------------------------------------|
|  | 21cd10d7288a10a3657e75a584e6971d04bec23f, latest | Apr 14, 2021 01:44:37 AM  | 401.09    | <div>Copy URI</div> | <div>sha256:1dae8b6197d0bc...</div>  |
|  | decef331811be1dd272717c734a2cbca3df044e6         | Apr 14, 2021 01:39:59 AM  | 401.08    | <div>Copy URI</div> | <div>sha256:d1fcadc546eca64...</div> |
|  | 7217c0d2a2ed5338aa06c56bed4e9a3efd87655c         | Apr 14, 2021 01:32:34 AM  | 401.08    | <div>Copy URI</div> | <div>sha256:d7161d80f671b9...</div>  |
|  | 47fd8bbecf3060c76a324c168e98c16868ca0b68         | Apr 14, 2021 12:56:33 ... | 401.08    | <div>Copy URI</div> | <div>sha256:3a4f7ad5a9cc881...</div> |

Build stage show succeeded using buildspec.yml.

Build logs

Phase details

Reports

Environment variables


Build details

Resource utilization

| Name             | Status    | Context | Duration | Start time                      | End time                        |
|------------------|-----------|---------|----------|---------------------------------|---------------------------------|
| SUBMITTED        | Succeeded | -       | <1 sec   | Apr 14, 2021 1:30 AM (UTC+8:00) | Apr 14, 2021 1:30 AM (UTC+8:00) |
| QUEUED           | Succeeded | -       | 1 sec    | Apr 14, 2021 1:30 AM (UTC+8:00) | Apr 14, 2021 1:30 AM (UTC+8:00) |
| PROVISIONING     | Succeeded | -       | 26 secs  | Apr 14, 2021 1:30 AM (UTC+8:00) | Apr 14, 2021 1:31 AM (UTC+8:00) |
| DOWNLOAD_SOURCE  | Succeeded | -       | <1 sec   | Apr 14, 2021 1:31 AM (UTC+8:00) | Apr 14, 2021 1:31 AM (UTC+8:00) |
| INSTALL          | Succeeded | -       | <1 sec   | Apr 14, 2021 1:31 AM (UTC+8:00) | Apr 14, 2021 1:31 AM (UTC+8:00) |
| PRE_BUILD        | Succeeded | -       | 33 secs  | Apr 14, 2021 1:31 AM (UTC+8:00) | Apr 14, 2021 1:31 AM (UTC+8:00) |
| BUILD            | Succeeded | -       | 44 secs  | Apr 14, 2021 1:31 AM (UTC+8:00) | Apr 14, 2021 1:32 AM (UTC+8:00) |
| POST_BUILD       | Succeeded | -       | 6 secs   | Apr 14, 2021 1:32 AM (UTC+8:00) | Apr 14, 2021 1:32 AM (UTC+8:00) |
| UPLOAD_ARTIFACTS | Succeeded | -       | <1 sec   | Apr 14, 2021 1:32 AM (UTC+8:00) | Apr 14, 2021 1:32 AM (UTC+8:00) |
| FINALIZING       | Succeeded | -       | 2 secs   | Apr 14, 2021 1:32 AM (UTC+8:00) | Apr 14, 2021 1:32 AM (UTC+8:00) |
| COMPLETED        | Succeeded | -       | -        | Apr 14, 2021 1:32 AM (UTC+8:00) | -                               |

Deployment completed.


↓

 **Deploy** Succeeded

Pipeline execution ID: [4770d006-192b-481b-85d2-3845b812c02e](#)

Deploy


[Amazon ECS](#)

 Succeeded - 28 minutes ago

[Details](#)

DeployBeanstalk

[AWS Elastic Beanstalk](#)

 Succeeded - 29 minutes ago

[47fd8bbe](#) [Source: Update Dockerfile](#)


Cloudformation screenshot


CloudFormation > Stacks > GovTechFargateStack

**Stacks (2)**

Filter by stack name

Active View nested

GovTechFargateStack  
2021-04-13 20:21:15 UTC+0800  
 CREATE\_COMPLETE

fireeye-dod-s3-integration  
2020-10-28 18:55:29 UTC+0800  
 CREATE\_COMPLETE



**GovTechFargateStack**

Delete Update Stack actions Create stack

Stack info Events Resources Outputs Parameters Template Change sets

**Events (44)**

Search events


| Timestamp                    | Logical ID          | Status  | Status reason |
|------------------------------|---------------------|---|---------------|
| 2021-04-13 20:24:14 UTC+0800 | GovTechFargateStack |  CREATE_COMPLETE | -             |
| 2021-04-13 20:24:13 UTC+0800 | AutoScalingPolicy   |  CREATE_COMPLETE | -             |


CloudFormation > Stacks > GovTechFargateStack

**Stacks (2)**

Filter by stack name





Active View nested

GovTechFargateStack  
2021-04-13 20:21:15 UTC+0800  
 CREATE\_COMPLETE

fireeye-dod-s3-integration  
2020-10-28 18:55:29 UTC+0800  
 CREATE\_COMPLETE


**Resources (14)**

Search resources

| Logical ID             | Physical ID   | Type  | Status  |
|------------------------|---|---|---|
| AutoScalingTarget      | service/govtechserviceCluster/govtechservice ecs:service:DesiredCount ecs | AWS::ApplicationAutoScaling::ScalableTarget |  CREATE_COMPLETE |
| Cluster                | <a href="#">govtechserviceCluster</a>                                     | AWS::ECS::Cluster                           |  CREATE_COMPLETE |
| ContainerSecurityGroup | <a href="#">sg-04b3ea4200b9d9b24</a>                                      | AWS::EC2::SecurityGroup                     |  CREATE_COMPLETE |
| ExecutionRole          | <a href="#">govtechserviceExecutionRole</a>                               | AWS::IAM::Role                              |  CREATE_COMPLETE |

In local machine

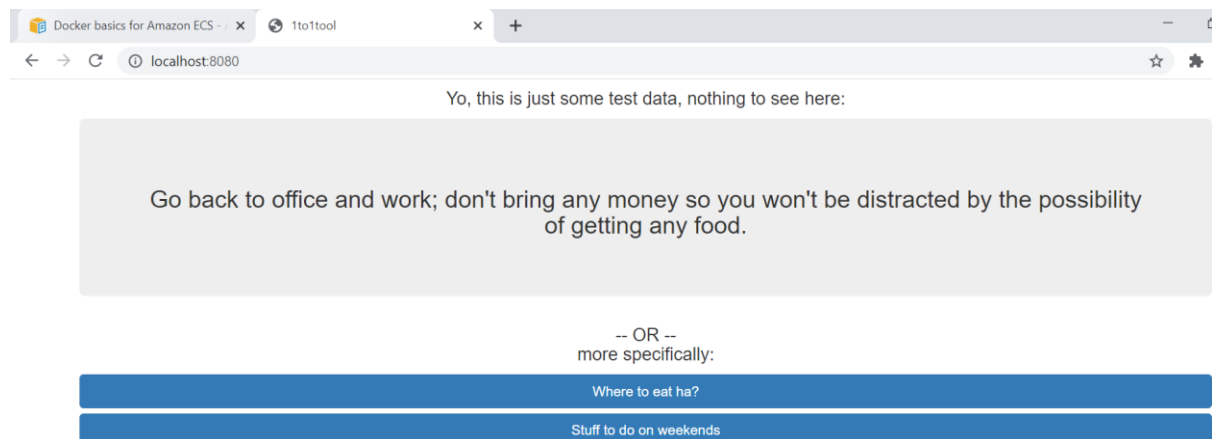
```
docker run -p 8080:3000 --env DATA_FILE="Questions-test.json" ffda960e823e
```

 MINGW64:/c/Users/hansh/Documents/devops/one2onetool2/one2onetool

```
hansh@LAPTOP-D8N26140 MINGW64 ~/Documents/devops/one2onetool2/one2onetool (staging)
$ ll
total 226
-rw-r--r-- 1 hansh 197609    188 Apr 13 17:08 Dockerfile
-rw-r--r-- 1 hansh 197609   1089 Apr 13 17:08 LICENSE
-rw-r--r-- 1 hansh 197609    13 Apr 13 17:08 README.md
-rw-r--r-- 1 hansh 197609  1174 Apr 14 02:31 appspec-fargate.yml
-rw-r--r-- 1 hansh 197609   345 Apr 14 02:31 appspec.yml
-rw-r--r-- 1 hansh 197609   442 Apr 14 02:31 builds-spec-ebstalk.yml
-rw-r--r-- 1 hansh 197609  1494 Apr 13 20:10 builds-spec.yml
-rw-r--r-- 1 hansh 197609 10189 Apr 14 02:31 cloudformation-fargate.yml
drwxr-xr-x 1 hansh 197609    0 Apr 13 17:08 controllers/
drwxr-xr-x 1 hansh 197609    0 Apr 13 17:08 data/
drwxr-xr-x 1 hansh 197609    0 Apr 13 17:08 helpers/
-rw-r--r-- 1 hansh 197609   128 Apr 14 02:31 image-definitions.json
-rw-r--r-- 1 hansh 197609   714 Apr 13 17:08 index.js
drwxr-xr-x 1 hansh 197609    0 Apr 13 17:08 models/
-rw-r--r-- 1 hansh 197609 196499 Apr 13 17:08 package-lock.json
-rw-r--r-- 1 hansh 197609   859 Apr 13 17:08 package.json
drwxr-xr-x 1 hansh 197609    0 Apr 13 17:08 routes/
drwxr-xr-x 1 hansh 197609    0 Apr 14 02:31 scripts/
drwxr-xr-x 1 hansh 197609    0 Apr 13 17:08 test/
drwxr-xr-x 1 hansh 197609    0 Apr 13 17:08 views/

hansh@LAPTOP-D8N26140 MINGW64 ~/Documents/devops/one2onetool2/one2onetool (staging)
$ docker run -p 8080:3000 --env DATA_FILE="Questions-test.json" ffda960e823e
Example app listening on port 3000
```

## Staging



Yo, this is just some test data, nothing to see here:

Go back to office and work; don't bring any money so you won't be distracted by the possibility of getting any food.

-- OR --  
more specifically:

Where to eat ha?

Stuff to do on weekends

```
docker run -p 8080:3000 --env DATA_FILE="Questions.json" ffda960e823e
```

## Release

Docker basics for Amazon ECS x 1to1tool x +

localhost:8080/?category=1

It's important to develop your staff. Try asking:

What work are you doing here that you feel is most in line with your long term goals?

-- OR --  
more specifically:

Organization Matters

Personal Development

Work Satisfaction

Establishing Rapport

Coaching

Receiving Feedback