

NOTE: Due to the CircleCI is out of credit, we created private repo on GitHub and a new project in CircleCI.

Part 1 Backend:

1. Set the environment variables on CircleCI

The screenshot shows the CircleCI Project Settings page for a project named 'SEPT'. The left sidebar contains navigation links: Overview, Advanced, Environment Variables (selected), SSH Keys, API Permissions, Jira Integration, and Slack Integration. The main content area is titled 'Environment Variables' and includes a description: 'Environment variables let you add sensitive data (e.g. API keys) to your jobs rather than placing them in the repository. The value of the variables cannot be read or edited in the app once they are set.' Below this, a link states: 'If you're looking to share environment variables across projects, try Contexts.' A table lists the following environment variables:

Name	Value	Action
AWS_ACCESS_KEY_ID	xxxx6TGV	✕
AWS_ACCOUNT_ID	xxxx9016	✕
AWS_DEFAULT_REGION	xxxxst-1	✕
AWS_RESOURCE_NAME_PREFIX	xxxxeam3	✕
AWS_SECRET_ACCESS_KEY	xxxxnQCs	✕
AWS_SESSION_TOKEN	xxxx1lc=	✕

Buttons for 'Add Variable' and 'Import Variables' are located at the top right of the table.

2. The config.yml on the private repo had been copied to our team repo <https://github.com/RMIT-SEPT/majorproject-1-fri-10-30-3>.
3. The screenshot of failed CI builds.

The screenshot shows the CircleCI build history for the 'SEPT' project. The left sidebar shows navigation links: Pipelines, Projects, Organization Settings, Plan, and Status (OPERATIONAL). The main content area displays a list of builds:

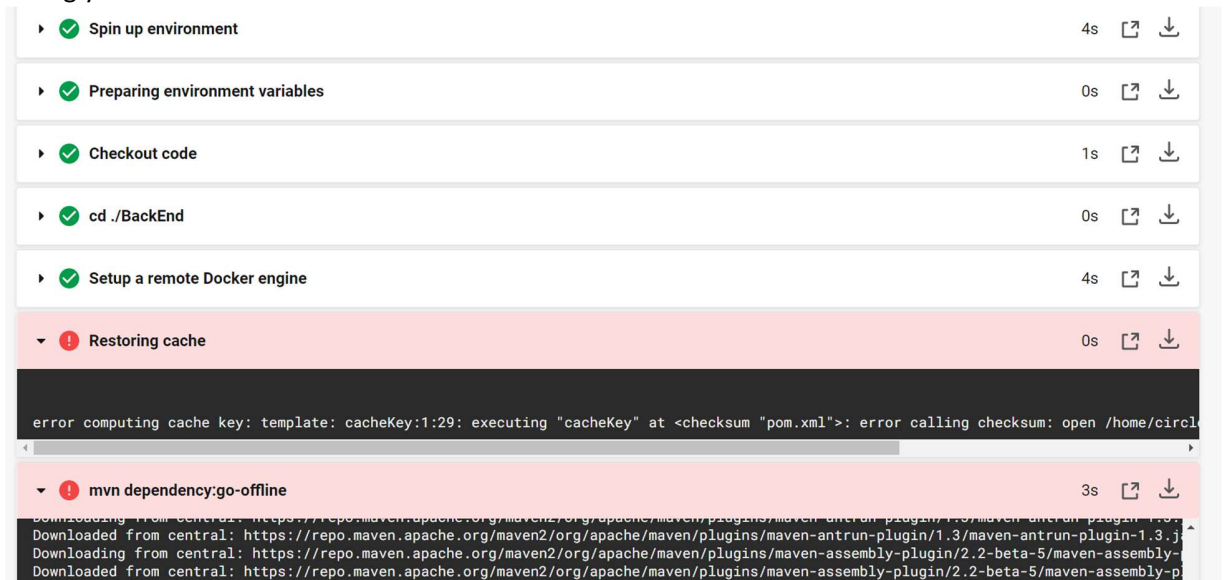
Build ID	Status	Job Name	Branch	Commit	Message	Time	Actions
SEPT #6	Failed	testing_suite	master	c1111ef	Update config.yml	1h ago 1m 44s	🔄 🗑️ ⌵ ...
SEPT #5	Failed	testing_suite	master	4604b91	Update config.yml	1h ago 1m 38s	🔄 🗑️ ⌵ ...
SEPT #4	Success	testing_suite	master	1bec661	Update config.yml	1h ago 1m 11s	🔄 🗑️ ⌵ ...
SEPT #3	Failed	Build Error	master	cd3c9a5	Merge pull request #2 from s3795574/development	2h ago 20s	🔄 🗑️ ⌵ ...

The error message for SEPT #3 is displayed below the build entry:

```
Unable to parse YAML
while parsing a block mapping
  in 'string', line 17, column 5:
    docker:
      ^
expected <block end>, but found '<block mapping start>'
  in 'string', line 19, column 6:
    environment:
      ^
```

#3 has some compile errors, they were about alignment.

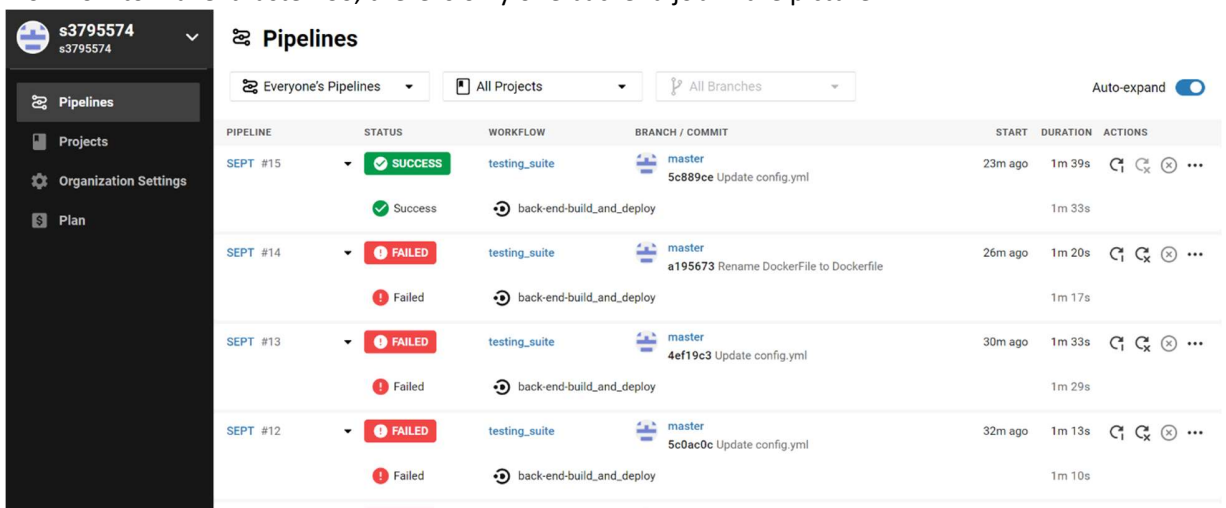
#4 is success because I comment out the jobs to test if there is any compile error.
 #5 and the builds after it, The front-end testing is passed, however, the backend job in config.yml failed.



As you can see from the picture above, it is a path error because my config.yml is at the root directory of the project. However, I have both frontend and backend folder, checksum cannot find pom.xml under root directory. (It is under /backend)

You can see I run the command `cd ./BackEnd`, it did not work because I put it in the wrong place.

- After tried 15 times, I finally pushed the image to the ECR. I commented out the frontend job in workflow to make it faster. So, there is only one backend job in the picture.



The issues were mainly about the path. Also, one of the issues is that the Dockerfile name, as you can see in the #14 on the picture, I changed it from DockerFile to Dockerfile.

- The picture below shows the detail of each steps

s3795574

s3795574

Pipelines

Projects

Organization Settings

Plan

Status

Help

OPERATIONAL

cd ./BackEnd && mvn dependency:go-offline

13s

Saving cache

0s

cd ./BackEnd && mvn package

11s

Uploading test results

0s

Uploading artifacts

0s

Install dependencies for AWS

33s

Build image

8s

Push image

11s

```
#!/bin/bash -eo pipefail
cd ./BackEnd && aws ecr get-login-password --region us-east-1 | docker login --username AWS --password-stdin 803415849016.dkr.ecr.us-east-1.amazonaws.com
docker push $FULL_IMAGE_NAME

WARNING! Your password will be stored unencrypted in /home/circleci/.docker/config.json.
Configure a credential helper to remove this warning. See
https://docs.docker.com/engine/reference/commandline/login/#credentials-store

Login Succeeded
The push refers to a repository [*****.dkr.ecr.*****.amazonaws.com/*****]

94be23df: Preparing
bb9d904b: Preparing
9e1e6ef5: Preparing
7f3d56a0: Preparing
latest: digest: sha256:e0b0983483b1c6939310d418487aa9795be3e15d3e41e52afe99719bd6da23e0 size: 1367
CircleCI received exit code 0
```

You can see the image had been built and pushed into ECR. You can see my account id in picture above. (The second line under Push image)

- The backend image had been pushed into AWS ECR.

aws

Services

Resource Groups

Resource Groups

vocstartsoft/user966515=s379...

N. Virginia

Support

Amazon Container Services

Amazon ECS

Clusters

Task definitions

Amazon EKS

Clusters

Amazon ECR

Repositories

Images

Permissions

Lifecycle Policy

Tags

ECR > Repositories > fri-10-30-team3

fri-10-30-team3

View push commands

Images (1)

Find images

Image tag

Image URI

Pushed at

Digest

Size (MB)

Scan status

Vulnerab

latest

803415849016.dkr.ecr.us-east-1.amazonaws.com/fri-10-30-team3:latest

09/19/20, 08:59:54 PM

sha256:e0b098348...

115.11

-

-

- Make it run on AWS

Clusters > default > Task: d21f6b80-cca2-476d-bc90-05b0d1d83504

Task : d21f6b80-cca2-476d-bc90-05b0d1d83504

Run more like this St

Details Tags Logs

Cluster default

Launch type FARGATE

Platform version 1.3.0

Task definition first-run-task-definition:5

Group service:custom-service

Task role None

Last status RUNNING

Desired status RUNNING

Created at 2020-09-20 09:48:00 +1000

Started at 2020-09-20 09:48:30 +1000

Network

Network mode awsvpc

ENI Id eni-051a6d4a4e9d85bb0

Subnet Id subnet-0b7e74b4c3a0f17ea

Private IP 10.0.1.115

Public IP 54.210.18.102

Mac address 0a:41:cb:d2:92:89

Containers

Last updated on September 20, 2020 9:48:58 AM (11m ago) ↺ ⌂

Name	Container Runtime I...	Statu...	Image	Image Digest	CPU ...	Hard/...	Esse...	Reso...
custom	86fad643248c5da085...	RUN...	803415849016.dkr.ecr.us-east-1.amazon...	sha256:e0b0983483b1c6939310d...	0	--/512	true	641fc...

You can see the image and its IP address. Now, you could use <http://54.210.18.102:8080/api/> to add or retrieve from the database (For milestone2 it is in-memory database).

(More API information are in another PDF file in the submission.)

POST http://localhost:8080/api/empl... GET http://54.210.18.102:8080/api/e... No Environment

Untitled Request BUILD

POST http://54.210.18.102:8080/api/employee Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```

1 {
2   "username": "worker1",
3   "fname": "test",
4   "lname": "test",
5   "password": "123"
6 }

```

Body Cookies Headers (8) Test Results Status: 201 Created Time: 1434 ms Size: 378 B Save

Pretty Raw Preview Visualize JSON

```

1 {
2   "id": 1,
3   "username": "worker1",
4   "password": "123",
5   "created_At": "2020-09-19",
6   "updated_At": null,
7   "fname": "test",
8   "lname": "test"
9 }

```

POST http://localhost:8080/api/empl... GET http://54.210.18.102:8080/api/e...

Untitled Request BUILD

GET http://54.210.18.102:8080/api/employee Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (8) Test Results Status: 200 OK Time: 577 ms Size: 375 B Save

Pretty Raw Preview Visualize JSON

```

1 {
2   "id": 1,
3   "username": "worker1",
4   "password": "123",
5   "created_At": "2020-09-19",
6   "updated_At": null,
7   "fname": "test",
8   "lname": "test"
9 }
10
11

```

In the java code, new Date() probably returns GMT time so that it shows 2020-09-19 in the picture above, however, in the Task screenshot, you can see I did this on 2020-09-20. We set the H2 database could not be accessed from remote, so visit <http://54.210.18.102:8080/h2-console/> will get an error message.

Part 2 Frontend:

1. The details of Frontend debug on CircleCI

AroneSusau Arone Tie Susau

Pipelines Projects Organization Settings Plan

Status OPERATIONAL Help

- Spin up environment 26s
- Preparing environment variables 0s
- Checkout code 1s
- Setup a remote Docker engine 2s
- Restoring cache 0s
- cd ./Frontend && npm install 0s
- Saving cache 0s
- Install dependencies for AWS 35s
- Build image 5s

error computing cache key: template: cacheKey:1:19: executing "cacheKey" at <checksum `./Frontend/package.json`>: error calling checksum: open /home/circleci/pro

cd ./Frontend && npm install

Error computing cache key: template: cacheKey:1:19: executing "cacheKey" at <checksum `./Frontend/package.json`>: error calling checksum: open /home/circleci/pro

Save cache

Error computing cache key: template: cacheKey:1:19: executing "cacheKey" at <checksum `./Frontend/package.json`>: error calling checksum: open /home/circleci/pro

Install dependencies for AWS

Build image

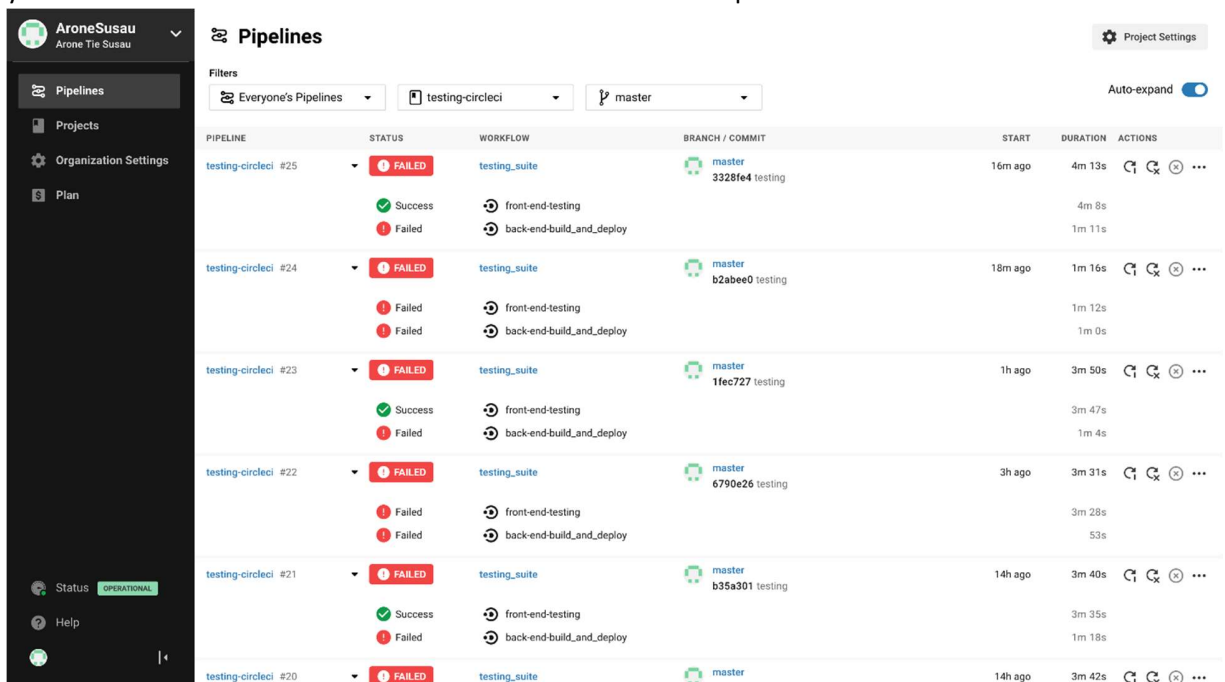
```

#!/bin/bash -eo pipefail
cd ./Frontend && docker build -t $FULL_IMAGE_NAME .

Sending build context to Docker daemon  5.15kB
Step 1/13 : FROM node:13.12.0-alpine as build
13.12.0-alpine: Pulling from library/node
3a933944: Pulling fs layer
d932288e: Pulling fs layer
2c59b937: Pulling fs layer
Digest: sha256:cd8e723fab3827ad28a181ba288cae1f8b625f256ec2c86b9e94d9bfe834766
Status: Downloaded newer image for node:13.12.0-alpine
--> 45334206c5f5

```

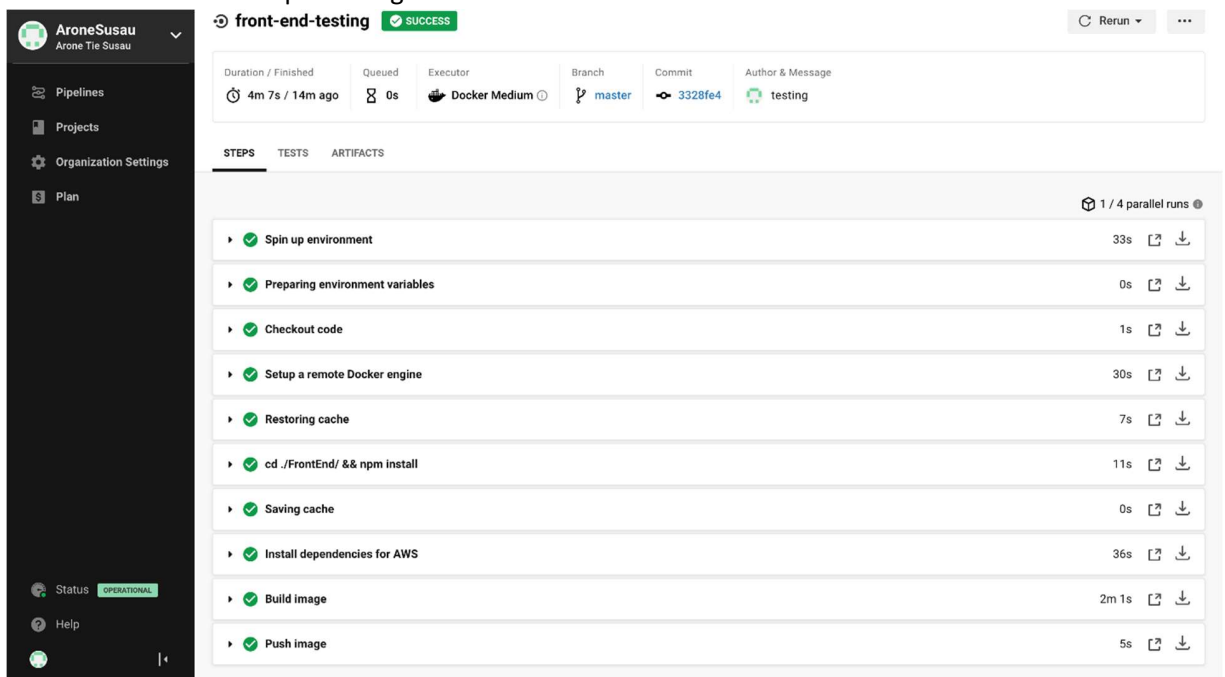
2. After 25 tries, it finally worked. NOTE: we did this separately(Frontend and Backend), so that you can see when the the backend is failed while the frontend passed.



The screenshot shows the 'Pipelines' page in the AroneSusau application. The left sidebar contains navigation links: Pipelines, Projects, Organization Settings, and Plan. The main content area displays a table of pipeline runs. The table has columns for Pipeline, Status, Workflow, Branch / Commit, Start, Duration, and Actions. The pipeline 'testing-circleci' is shown with multiple runs, all of which are marked as 'FAILED'. The workflow for each run is 'testing_suite', which includes steps for 'front-end-testing' and 'back-end-build_and_deploy'. The status of the 'front-end-testing' step is 'Success', while the 'back-end-build_and_deploy' step is 'Failed'.

PIPELINE	STATUS	WORKFLOW	BRANCH / COMMIT	START	DURATION	ACTIONS
testing-circleci #25	FAILED	testing_suite	master 3328fe4 testing	16m ago	4m 13s	🔄 📄 ⌵ ⋮
	Success	front-end-testing			4m 8s	
	Failed	back-end-build_and_deploy			1m 11s	
testing-circleci #24	FAILED	testing_suite	master b2abee0 testing	18m ago	1m 16s	🔄 📄 ⌵ ⋮
	Failed	front-end-testing			1m 12s	
	Failed	back-end-build_and_deploy			1m 0s	
testing-circleci #23	FAILED	testing_suite	master 1fec727 testing	1h ago	3m 50s	🔄 📄 ⌵ ⋮
	Success	front-end-testing			3m 47s	
	Failed	back-end-build_and_deploy			1m 4s	
testing-circleci #22	FAILED	testing_suite	master 6790e26 testing	3h ago	3m 31s	🔄 📄 ⌵ ⋮
	Failed	front-end-testing			3m 28s	
	Failed	back-end-build_and_deploy			53s	
testing-circleci #21	FAILED	testing_suite	master b35a301 testing	14h ago	3m 40s	🔄 📄 ⌵ ⋮
	Success	front-end-testing			3m 35s	
	Failed	back-end-build_and_deploy			1m 18s	
testing-circleci #20	FAILED	testing_suite	master	14h ago	3m 42s	🔄 📄 ⌵ ⋮

3. The details of build and push image to aws.



The screenshot shows the details of a specific pipeline run, 'front-end-testing', which is marked as 'SUCCESS'. The left sidebar is the same as in the previous screenshot. The main content area displays the pipeline's metadata, including Duration / Finished (4m 7s / 14m ago), Queued (0s), Executor (Docker Medium), Branch (master), Commit (3328fe4), and Author & Message (testing). Below the metadata, there are tabs for STEPS, TESTS, and ARTIFACTS. The STEPS tab is active, showing a list of steps with their durations and download links. The steps are: Spin up environment (33s), Preparing environment variables (0s), Checkout code (1s), Setup a remote Docker engine (30s), Restoring cache (7s), cd ./FrontEnd/ && npm install (11s), Saving cache (0s), Install dependencies for AWS (36s), Build image (2m 1s), and Push image (5s).

Step	Duration	Download
Spin up environment	33s	📄 ⌵
Preparing environment variables	0s	📄 ⌵
Checkout code	1s	📄 ⌵
Setup a remote Docker engine	30s	📄 ⌵
Restoring cache	7s	📄 ⌵
cd ./FrontEnd/ && npm install	11s	📄 ⌵
Saving cache	0s	📄 ⌵
Install dependencies for AWS	36s	📄 ⌵
Build image	2m 1s	📄 ⌵
Push image	5s	📄 ⌵

4. The link of frontend page is <http://34.207.143.47/>, Inside the **admin page**, you can see the employee record shown on the Backend part step 7 are listed on the web page.

Welcome Admin Foo

Create Customer

Create Employee

Create Skill

Create Schedule

Create Booking

	Customers		Employees		Skills	Schedules	Bookings	
id	userName	password	created_At	updated_At			fname	lname
1	worker1	123	2020-09-19				test	test