CLOUD COMPUTING

JOBSHEET 12

CI/CD

Oleh:

Rajendra Rakha Arya Prabaswara

(1941720080)

TI – 3H

PROGRAM STUDI D4 TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2021

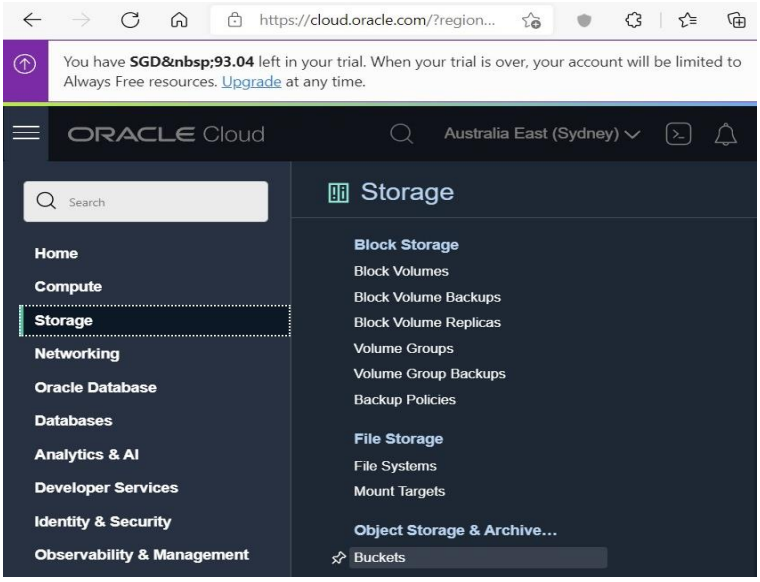
Praktikum

1. Hosting Static Website Menggunakan CI/CD Pipeline

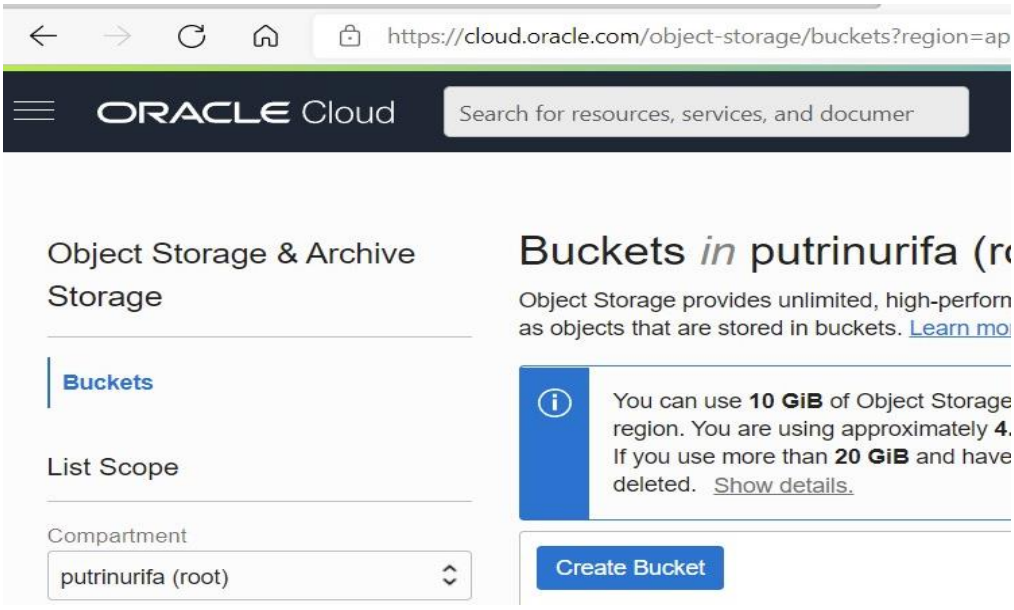
   Pada praktikum yang pertama akan dilakukan hosting static website di layanan Object Storage dari Oracle menggunakan GitHub Actions.

   1.1 **Menyiapkan Bucket Object Storage**

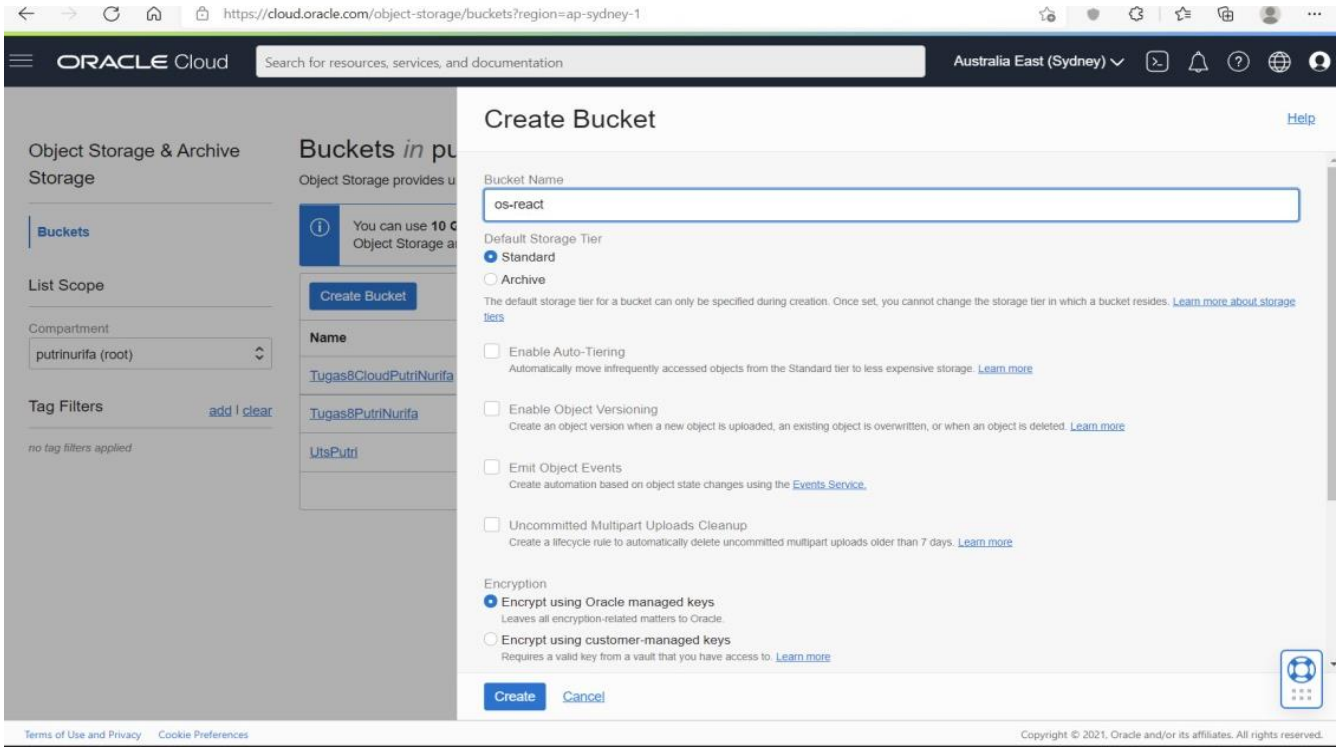
   a. Masuk ke akun https://cloud.oracle.com dan navigasi ke menu Storage -> Buckets



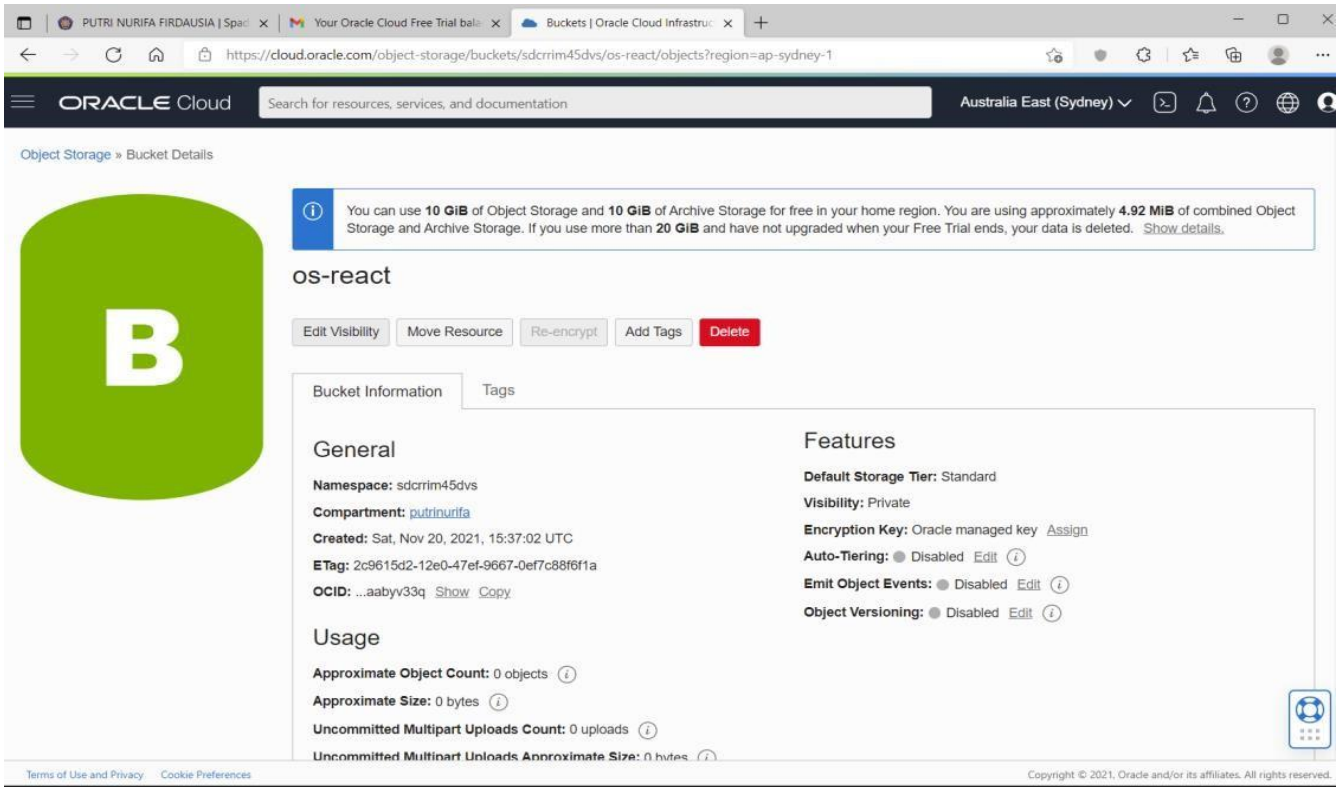
   b. Tekan tombol Create Bucket,



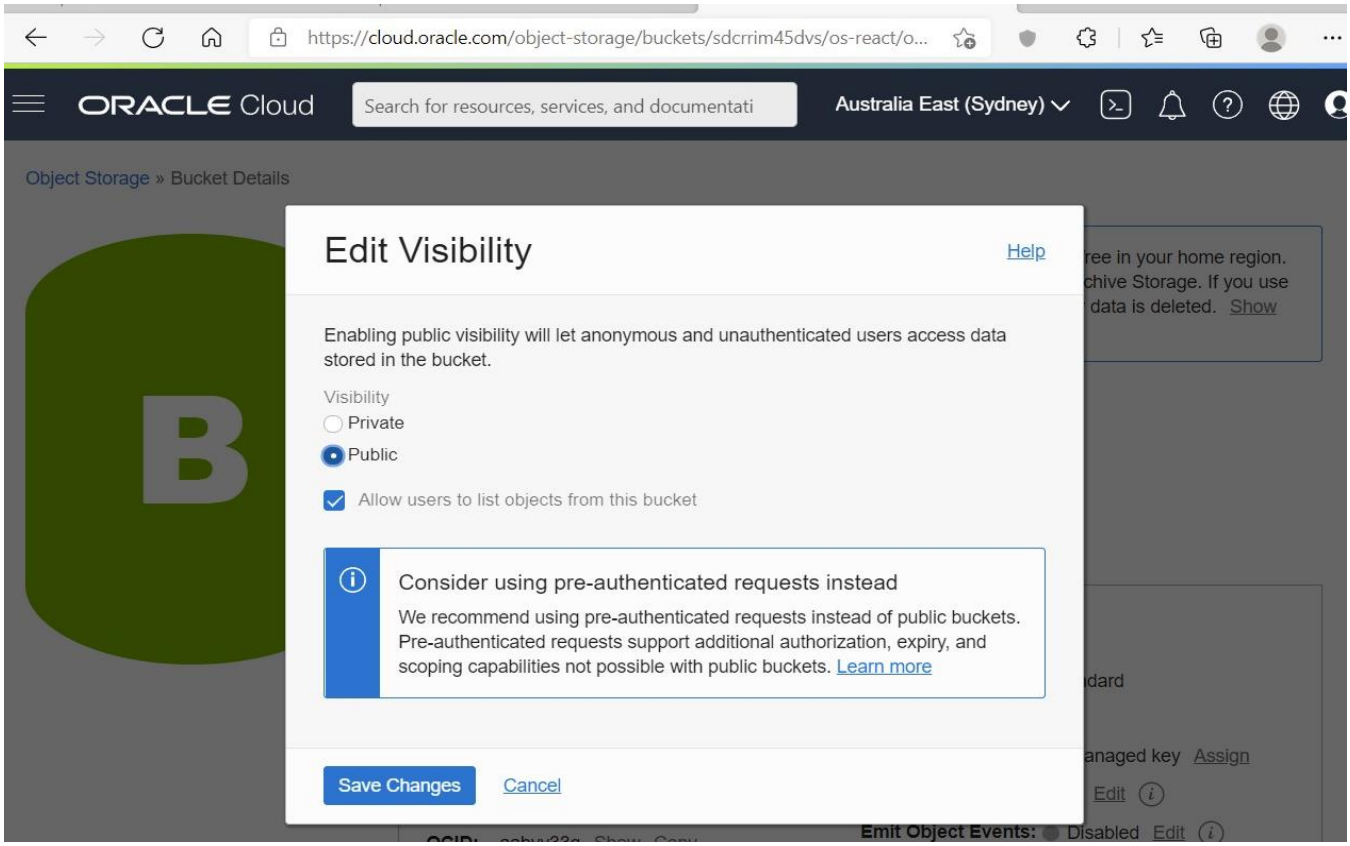
dan lengkapi Bucket Name dengan os-react. Biarkan nilai lainnya dengan nilai default kemudian tekan tombol Create.



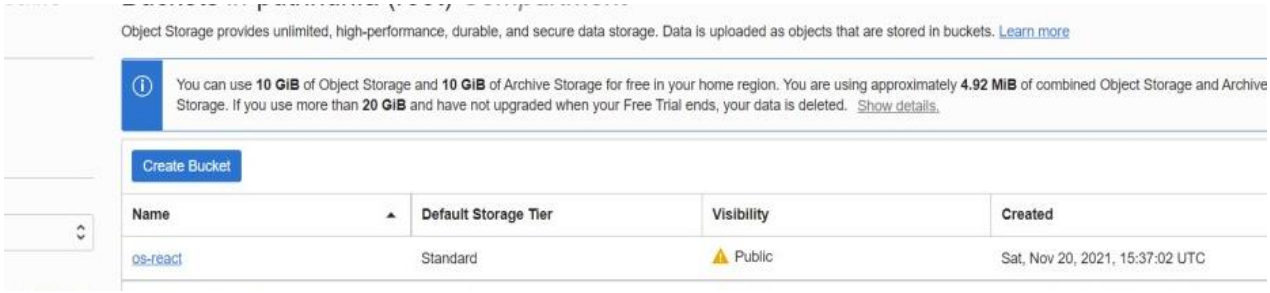
c. Ubah visibility bucket dengan menekan tombol Edit Visibility,



dan ubah menjadi public. Tekan tombol Save Changes untuk menyimpan perubahan.



d. Pada dashboard bucket, perhatikan nilai Namespace. Nilai ini nanti akan digunakan untuk pengaturan deployment.



1.2 **Menyiapkan Project**

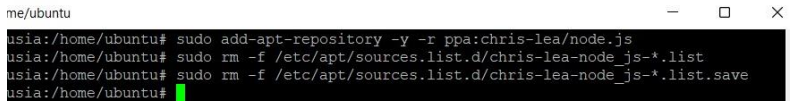
a. Pada praktikum ini, akan digunakan React sehingga harus diinstall terlebih dahulu Node. Untuk langkah instalasi Node silahkan gunakan tautan berikut sebagai referensi https://nodejs.org/en/download/ atau https://nodejs.org/en/download/package-manager/

Manual Installation

Langkah 1

Remove the old PPA if it exists

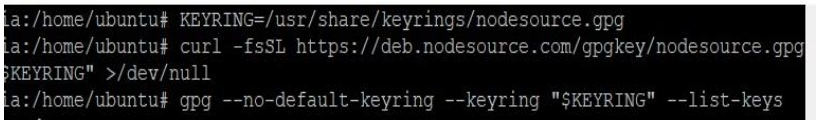
This step is only required if you previously used Chris Lea's Node.js PPA.



Langkah 2

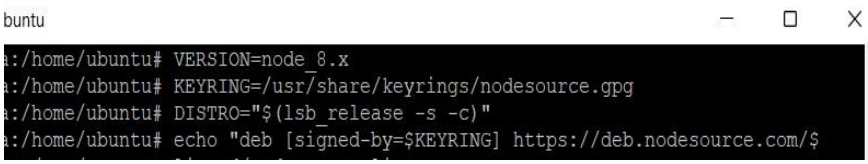
Add the NodeSource package signing key

The key ID is 9FD3B784BC1C6FC31A8A0A1C1655A0AB68576280.



Langkah 3

Add the desired NodeSource repository



Langkah 4

Update package lists and install Node.js

```
a:~$ curl -fsSL https://deb.nodesource.com/setup_17.x |

> sudo -E bash -

## Installing the NodeSource Node.js 17.x repo...


## Populating apt-get cache...

+ apt-get update
Hit:1 https://download.docker.com/linux/ubuntu focal InRelease
Get:2 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Hit:3 http://ap-sydney-1-ad-1.clouds.archive.ubuntu.com/ubuntu focal InRelease
Get:4 http://ap-sydney-1-ad-1.clouds.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:5 http://ap-sydney-1-ad-1.clouds.archive.ubuntu.com/ubuntu focal-backports InRelease [101 kB]
Get:6 http://ap-sydney-1-ad-1.clouds.archive.ubuntu.com/ubuntu focal-updates/main i386 Packages [563 kB
]
Get:7 http://ap-sydney-1-ad-1.clouds.archive.ubuntu.com/ubuntu focal-updates/main amd64 Packages [1344
kB]
Get:8 http://ap-sydney-1-ad-1.clouds.archive.ubuntu.com/ubuntu focal-updates/universe i386 Packages [64
7 kB]
Get:9 http://ap-sydney-1-ad-1.clouds.archive.ubuntu.com/ubuntu focal-updates/universe amd64 Packages [8
76 kB]
Fetched 3759 kB in 6s (587 kB/s)
Reading package lists... Done

a:~$ sudo apt-get install -y nodejs
```

```
a:~$ sudo apt install nodejs

Reading package lists... Done
Building dependency tree
Reading state information... Done
nodejs is already the newest version (17.1.0-deb-1nodesource1).
0 upgraded, 0 newly installed, 0 to remove and 16 not upgraded.
```

```
a:/home/ubuntu# node -v

a:/home/ubuntu#
```

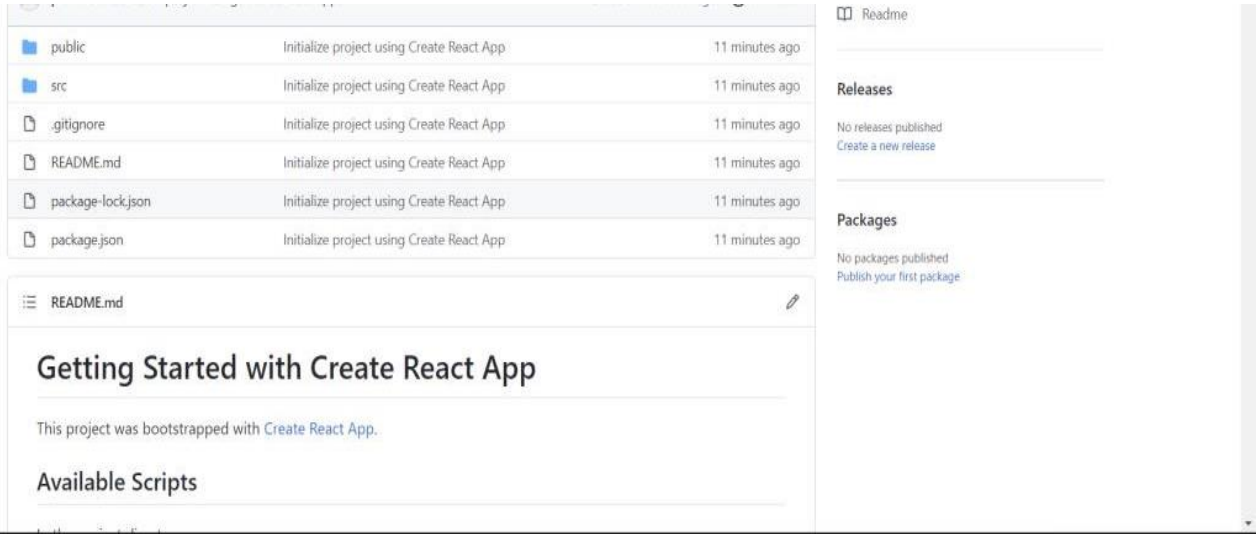
```
Select Command Prompt

Microsoft Windows [Version 10.0.22000.318]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Lenovo>node -v
v14.15.4

C:\Users\Lenovo>
```

Setelah Node terinstall, silahkan lakukan Langkah generate project dengan menggunakan perintah berikut.

```
npx create-react-app os-react
```

```
>npx create-react-app os-react

Installing packages. This might take a couple of minutes.
Installing react, react-dom, and react-scripts with cra-template...


added 1901 packages, and audited 1902 packages in 2m

153 packages are looking for funding
  run `npm fund` for details

27 vulnerabilities (16 moderate, 9 high, 2 critical)

To address all issues, run:
  npm audit fix

Run `npm audit` for details.

Initialized a git repository.

Installing template dependencies using npm...

added 57 packages, and audited 1959 packages in 8s

154 packages are looking for funding
  run `npm fund` for details

27 vulnerabilities (16 moderate, 9 high, 2 critical)
```

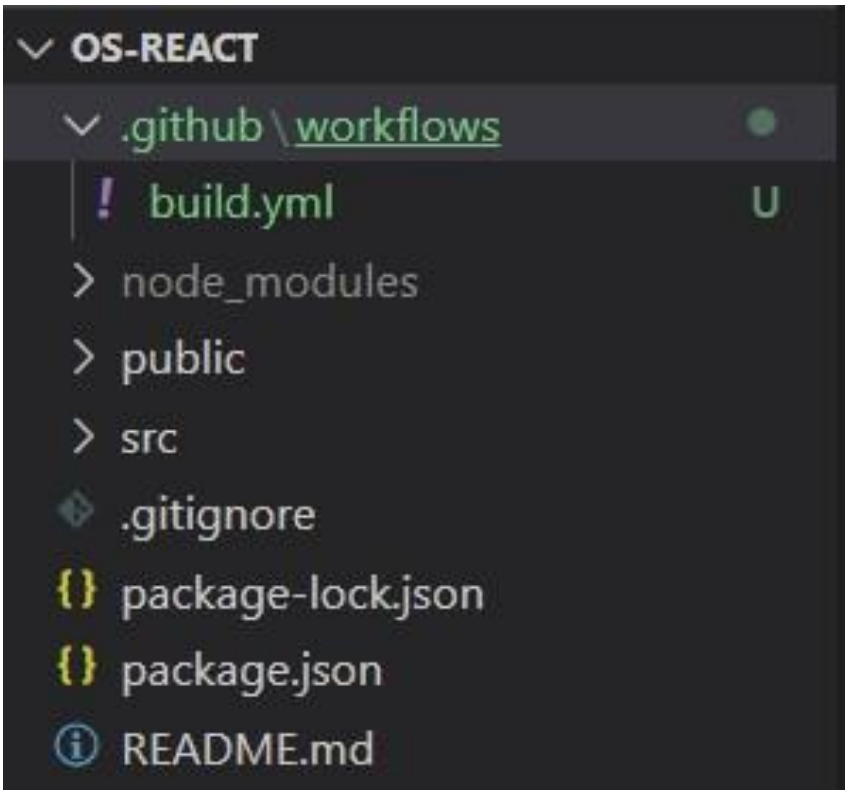
b. Buatlah repository pada GitHub kemudian push project tersebut ke dalam repository.



1.3 Menambahkan GitHub Workflow

a. Pipeline CI/CD pada GitHub dibuat dengan format yaml dan diletakkan pada lokasi .github/workflows. Buatlah file yaml pada lokasi tersebut, sebagai ilustrasi silahkan perhatikan filename berikut

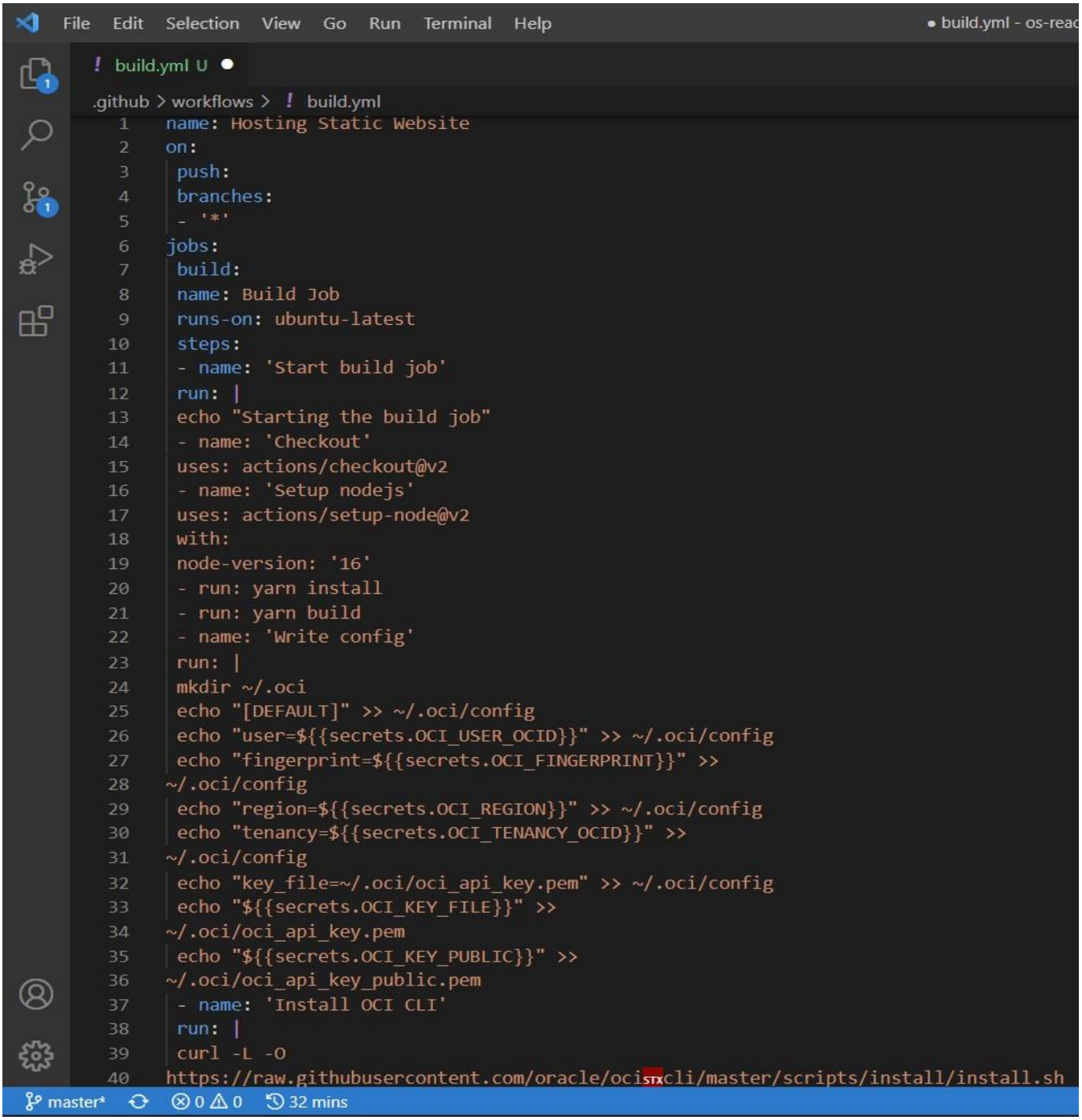
```
.github/workflows/build.yml
```



b. Salin konfigurasi pipeline berikut.

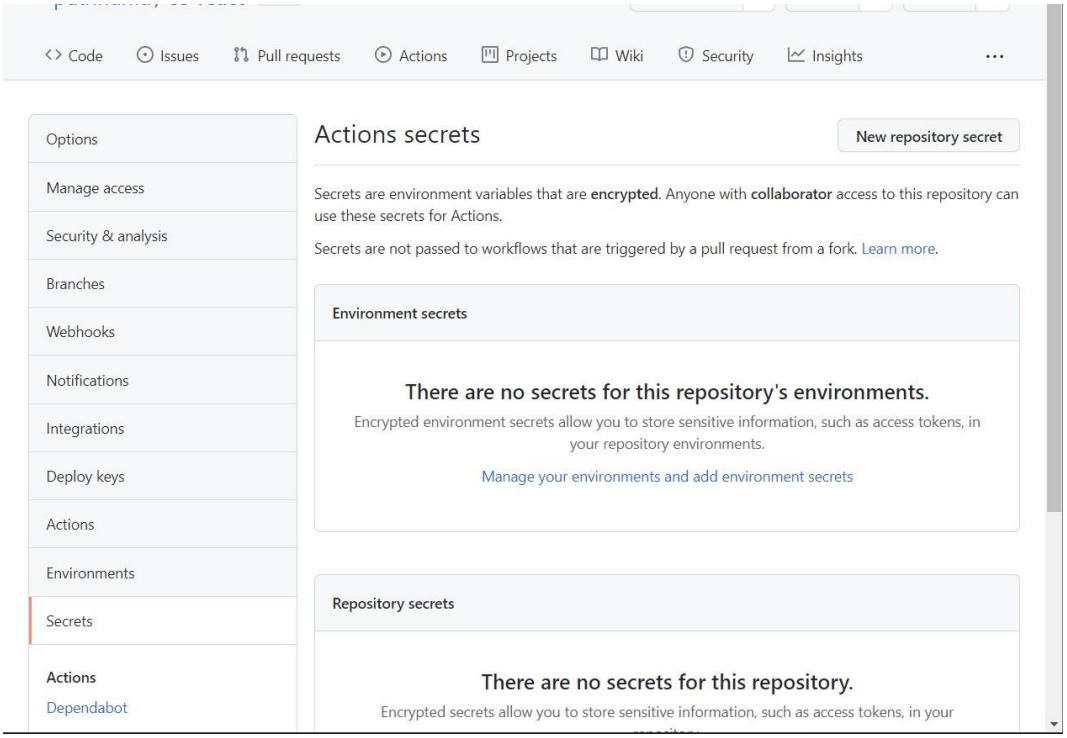
```
name: Hosting Static Website  on:
push:  branches:
-         '*'  jobs:  build:
name: Build Job  runs-on: ubuntu-
latest  steps:
-         name: 'Start build job'  run: |  echo
"Starting the build job"
-         name: 'Checkout' uses:
actions/checkout@v2
-         name: 'Setup nodejs' uses:
actions/setup-node@v2 with: node-version:
'16'
-         run: yarn install
-         run: yarn build  - name: 'Write config'
run: |  mkdir ~/.oci  echo "[DEFAULT]" >>
~/.oci/config
echo "user=${{secrets.OCI_USER_OCID}}" >> ~/.oci/config  echo
"fingerprint=${{secrets.OCI_FINGERPRINT}}" >> ~/.oci/config  echo
"region=${{secrets.OCI_REGION}}" >> ~/.oci/config  echo
"tenancy=${{secrets.OCI_TENANCY_OCID}}" >> ~/.oci/config  echo
"key_file=~/.oci/oci_api_key.pem" >> ~/.oci/config  echo
"${{secrets.OCI_KEY_FILE}}" >> ~/.oci/oci_api_key.pem  echo
"${{secrets.OCI_KEY_PUBLIC}}" >> ~/.oci/oci_api_key_public.pem
-         name: 'Install OCI CLI'  run: |
```

```
curl -L -O
https://raw.githubusercontent.com/oracle/ocicli/master/scripts/install/install.sh  chmod
+x install.sh
./install.sh --accept-all-defaults
echo "/home/runner/bin" >> $GITHUB_PATH exec -l $SHELL
- name: 'Fix Config File Permissions'  run: |
oci setup repair-file-permissions --file /home/runner/.oci/config oci setup repair-file-
permissions --file /home/runner/.oci/oci_api_key.pem - name:
'Deploy Into Object Storage'  run: | oci os object bulk-delete -bn
${{secrets.OCI_BUCKET}} --prefix static
--force  oci os object put -bn ${{secrets.OCI_BUCKET}} --file ./build/manifest.json -
-content-type application/json --force  oci os object bulk-upload -bn
${{secrets.OCI_BUCKET}} - -src-dir ./build -content-type text/html --include *.html
-- overwrite oci os object bulk-upload -bn ${{secrets.OCI_BUCKET}} - -src-dir
./build --content-type image/jpeg -include *.jpg -- overwrite  oci os object bulk-upload
-bn ${{secrets.OCI_BUCKET}} - -src-dir ./build -content-type text/javascript --
include *.js - -overwrite  oci os object bulk-upload -bn ${{secrets.OCI_BUCKET}} - -
src-dir ./build -content-type text/css --include *.css -- overwrite oci os object bulk-
upload -bn ${{secrets.OCI_BUCKET}} - -src-dir ./build --content-type text/plain -
exclude *.js -- exclude *.html --exclude *.jpg --exclude *.css --exclude
./build/manifest.json --overwrite
```

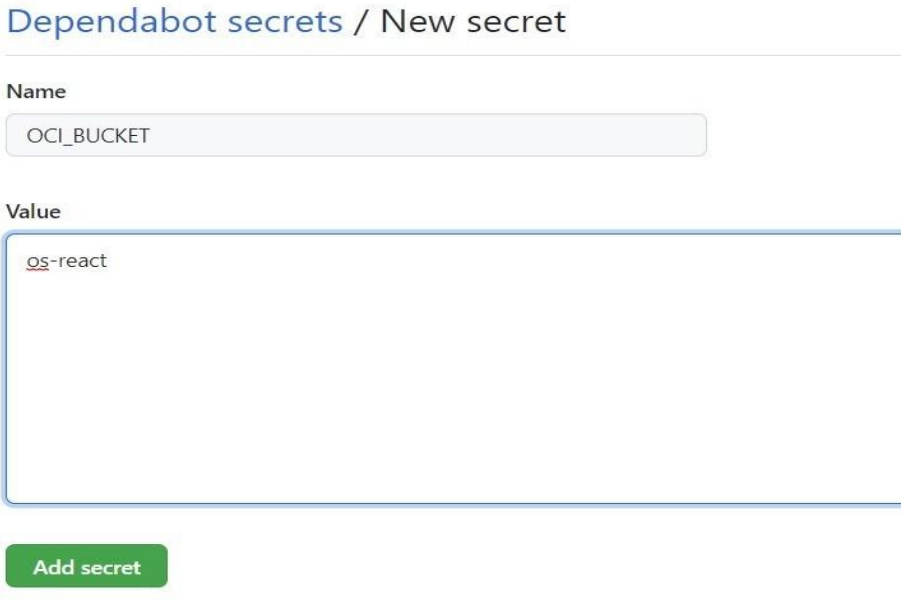


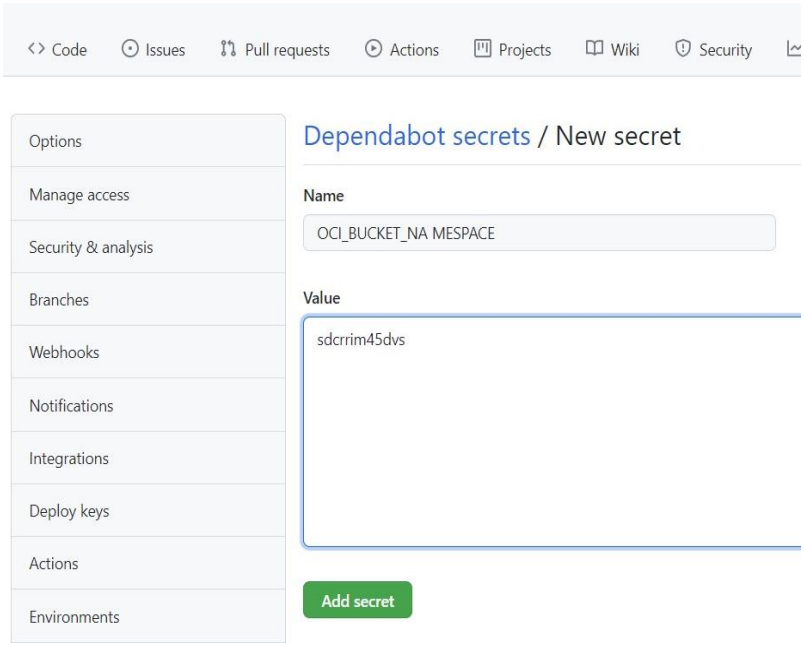
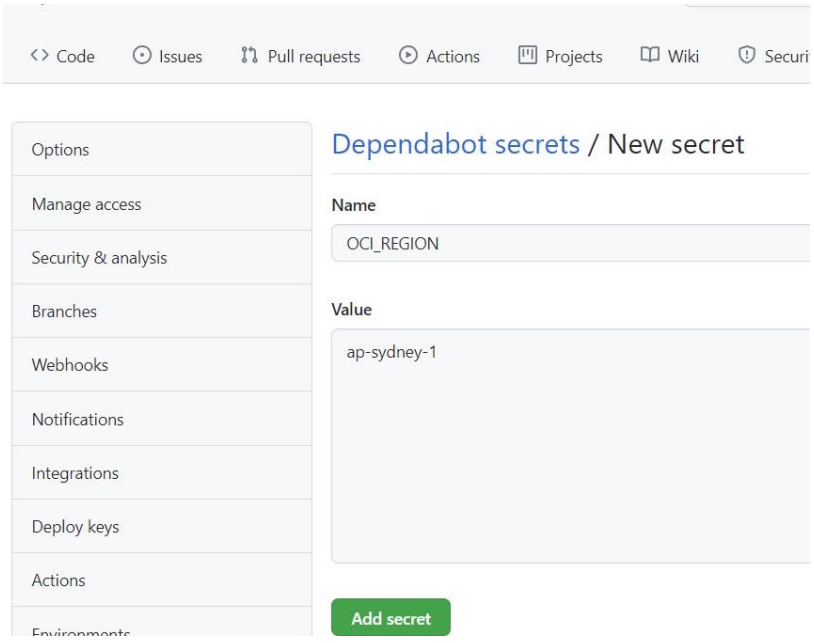
c. Pada pengaturan pipeline, terdapat nilai secrets yang diatur melalui pengaturan pada repository GitHub.

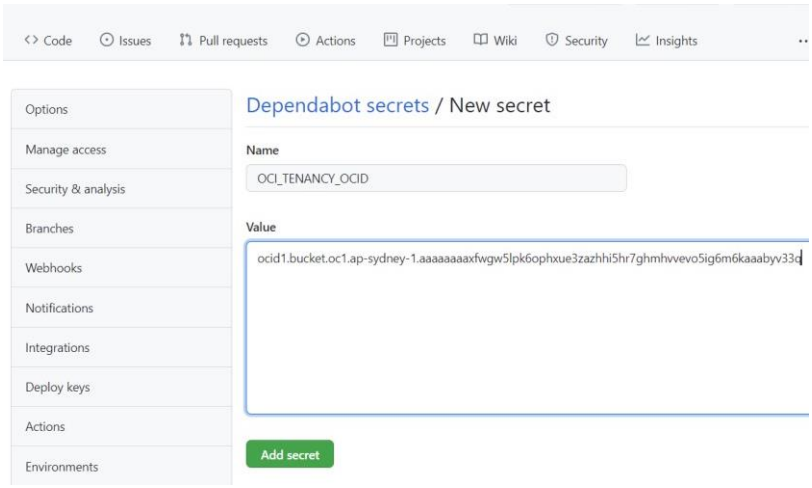
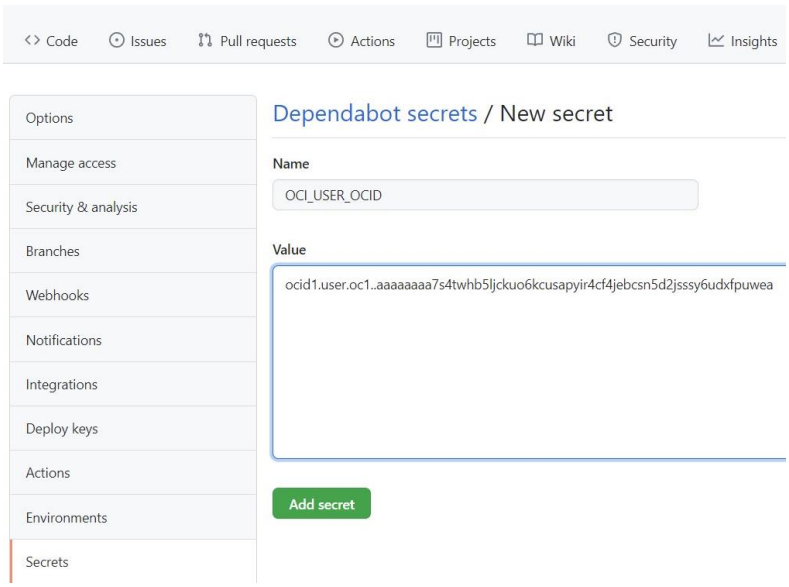
d. Buka alamat repository GitHub, dan masuk ke menu Settings -> Secrets

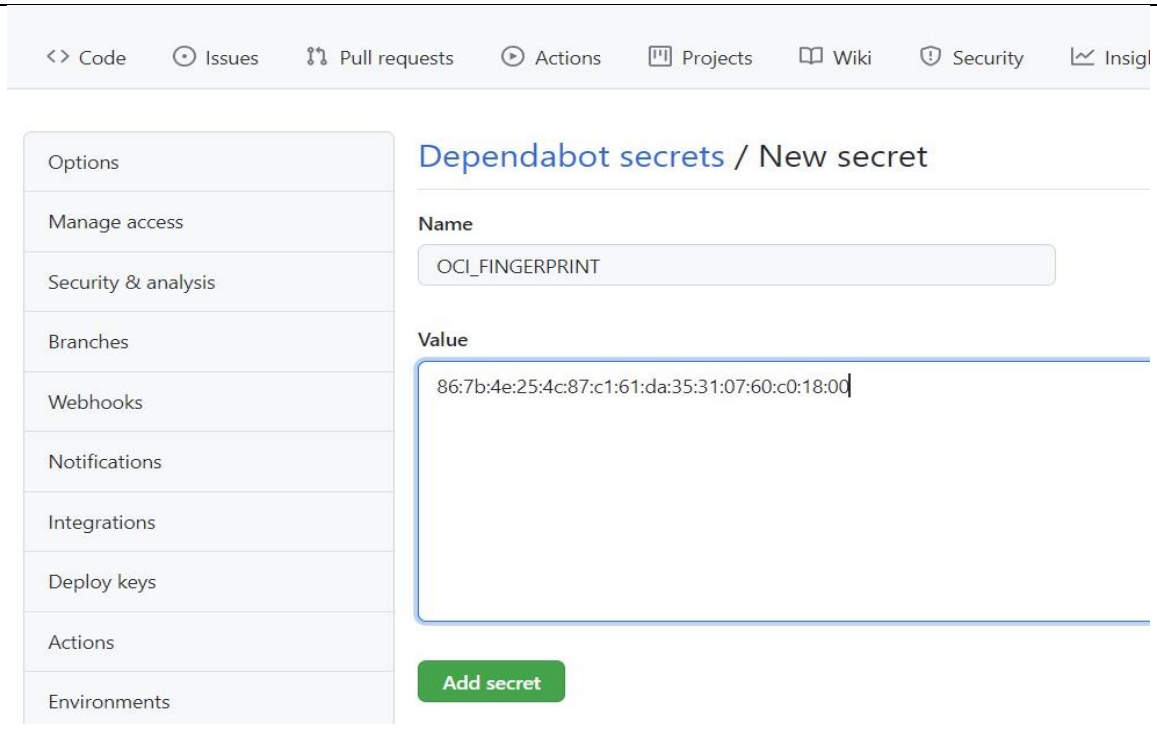


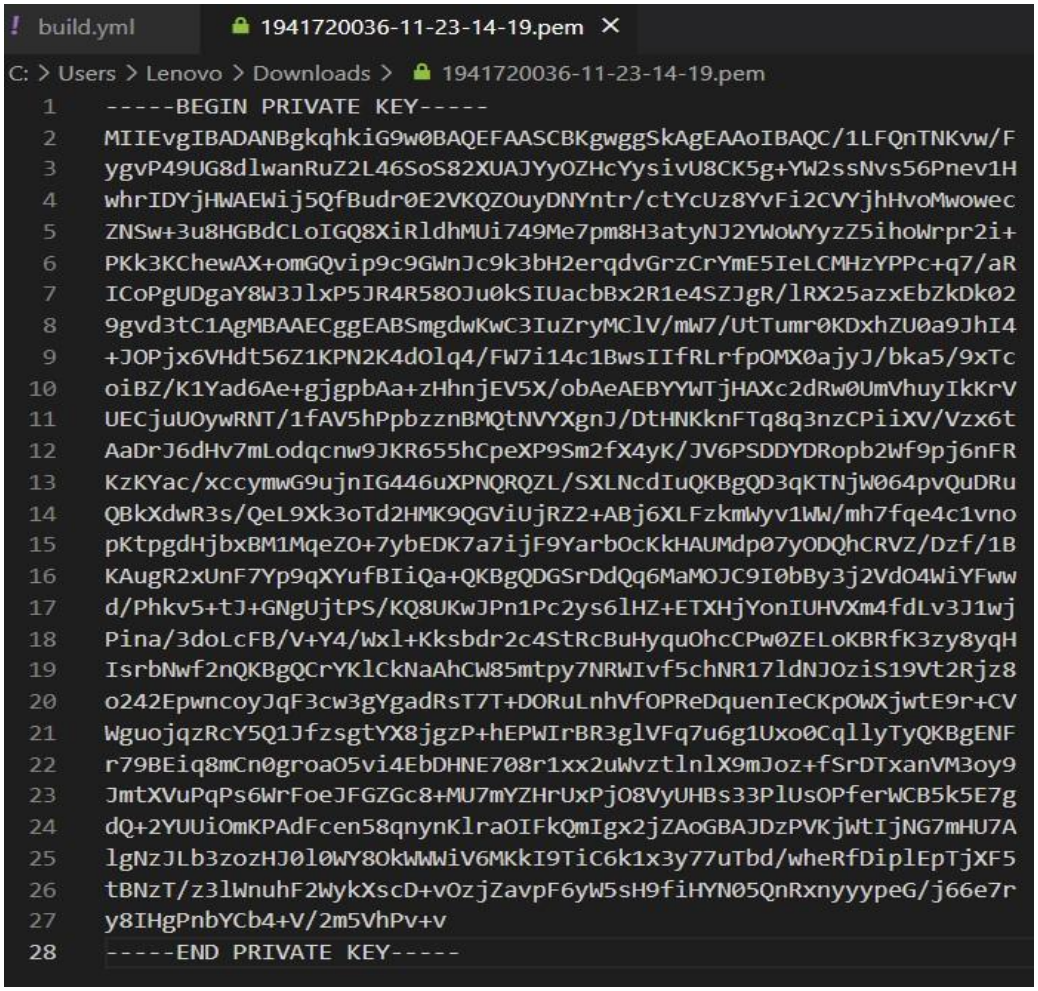
e. Sebagai panduan, silahkan gunakan table berikut untuk pengisian secrets

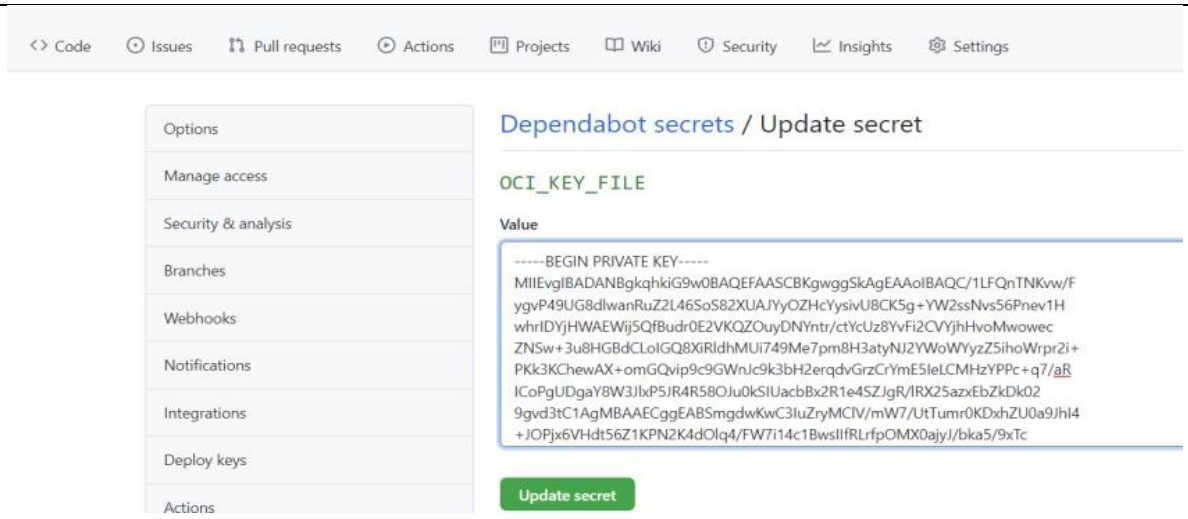
Nama Key	**Keterangan**
OCI_BUCKET	Nama bucket, pada praktikum ini digunakan nama os-react 

OCI_BUCKET_NA MESPACE	Namespace pada bucket, nilai ini dapat dilihat pada Bucket Information. Contoh: axlecckwcju3 
OCI_REGION	Nilai region dari akun oracle, Contoh: ap-melbourne-1 
OCI_TENANCY_OC ID	Nilai tenancy dapat dilihat pada menu Profile -> Tenancy. Contoh: ocid1.tenancy.oc1..aaaaaaaa26ywtdlgaxl3p3fn2sknly3wdb553 tmzgel4z222hk4kplcmcuvq

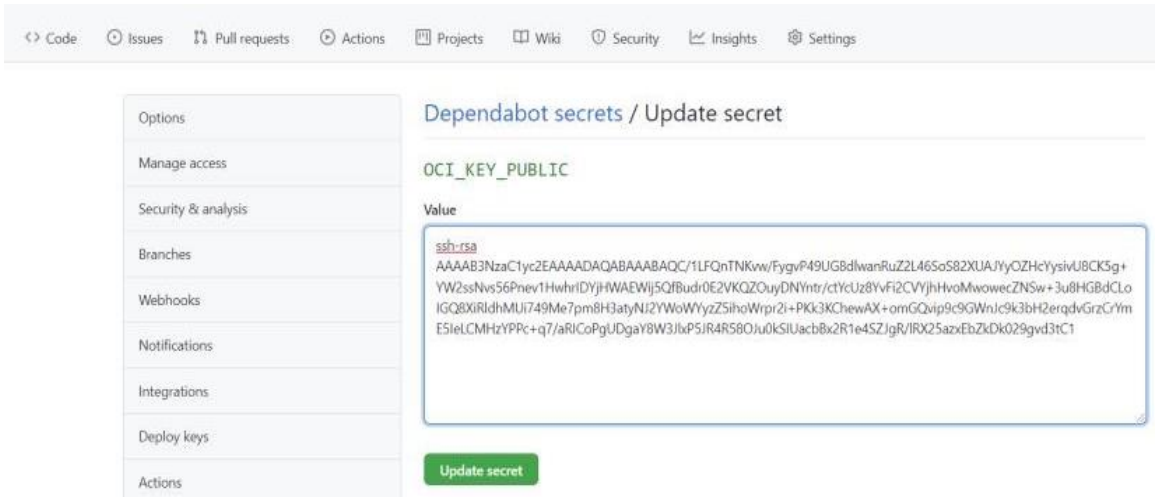
	
OCI_USER_OCID	Nilai dapat dilihat pada User Information pada menu Identity & Security -> Users 
OCI_FINGERPRINT OCI_KEY_FILE OCI_KEY_PUBLIC	Untuk mendapatkan nilai ini, pada User Information, pilih API Keys kemudian tekan tombol Add API Key. Pilih Generate API Key Pair, dan unduh private key dan public key. Pada berkas public key, diperlukan konversi format dengan menggunakan perintah ssh-keygen. Masukkan perintah berikut: ssh-keygen -I -m PKCS8 -f

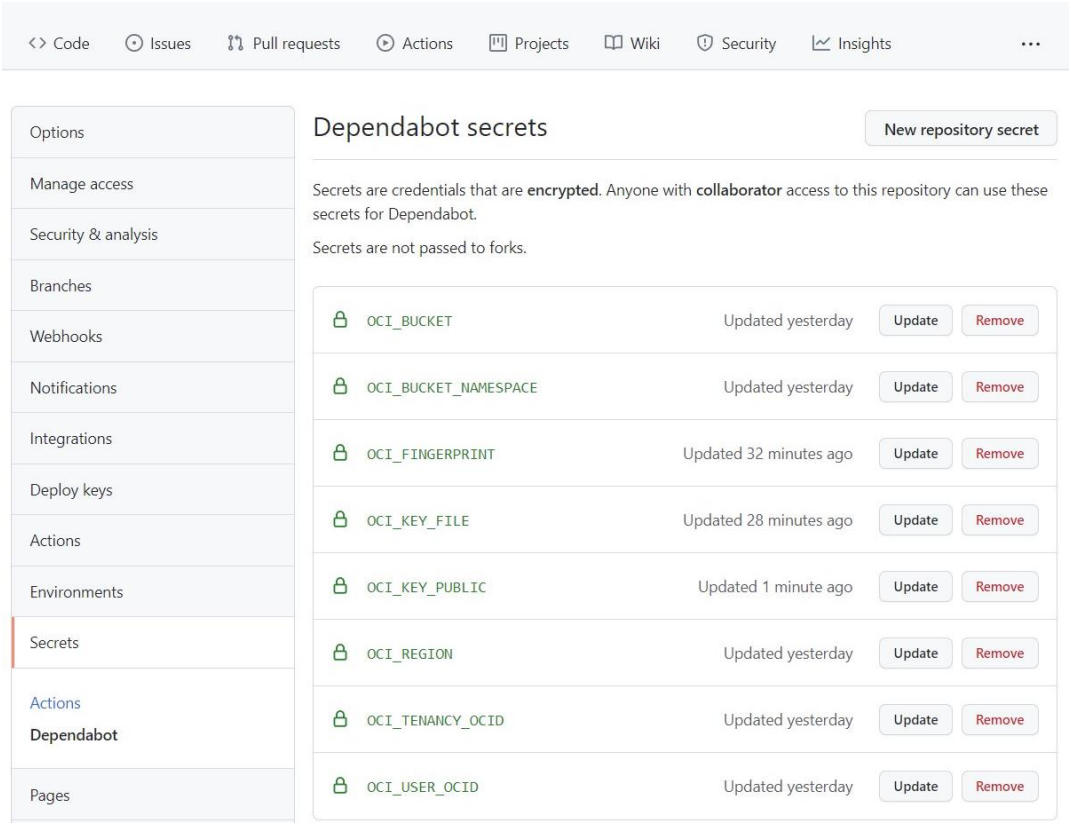

Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insigl
Options
Manage access
Security & analysis
Branches
Webhooks
Notifications
Integrations
Deploy keys
Actions
Environments
Dependabot secrets / New secret
Name
OCI_FINGERPRINT
Value
86:7b:4e:25:4c:87:c1:61:da:35:31:07:60:c0:18:00
Add secret


build.yml
1941720036-11-23-14-19.pem
C: > Users > Lenovo > Downloads > 1941720036-11-23-14-19.pem
-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQC/1LFQnTNKvw/F
ygvP49UG8dlwanRuZ2L46SoS82XUAJYyOZHcYysivU8CK5g+YW2ssNvs56Pnev1H
whrIDYjHWAEWij5QfBudr0E2VKQZOuyDNYntr/ctYcUz8YvFi2CVYjhHvoMwowec
ZNSw+3u8HGBdCLoIGQ8XiRldhMUi749Me7pm8H3atyNJ2YWoWYyzZ5ihoWrpr2i+
PKk3KChewAX+omGQvip9c9GWnJc9k3bH2erqdvGrzCrYmE5IeLCMHzYPPc+q7/aR
ICoPgUDgaY8W3JlxP5JR4R58OJu0kSIUacbBx2R1e4SZJgR/lRX25azxEbZkDk02
9gvd3tC1AgMBAAECggEABSmgdwKwC3IuZryMClV/mW7/UtTumr0KDxhZU0a9JhI4
+JOPjx6VHdt56Z1KPN2K4dOlq4/FW7i14c1BwsIIfRLrfpOMX0ajyJ/bka5/9xTc
oiBZ/K1Yad6Ae+gjgpbAa+zHhnjEV5X/obAeAEBYYWTjHAXc2dRw0UmVhuyIkKrV
UECjuUOywRNT/1fAV5hPpbzznBMQtNVYXgnJ/DtHNKknFTq8q3nzCPiiXV/Vzx6t
AaDrJ6dHv7mLodqcnw9JKR655hCpeXP9Sm2fX4yK/JV6PSDDYDRopb2Wf9pj6nFR
KzKYac/xccymwG9ujnIG446uXPNQRQZL/SXLNcdIuQKBgQD3qKTNjW064pvQuDRu
QBkXdwR3s/QeL9Xk3oTd2HMK9QGViUjRZ2+ABj6XLFzkmWyv1WW/mh7fqe4c1vno
pKtpgdHjbxBM1MqeZO+7ybEDK7a7ijF9YarbOcKkHAUMdp07yODQhCRVZ/Dzf/1B
KAugR2xUnF7Yp9qXYufBIiQa+QKBgQDGSrDdQq6MaMOJC9I0bBy3j2VdO4WiYFww
d/Phkv5+tJ+GNgUjtPS/KQ8UKwJPn1Pc2ys6lHZ+ETXHjYonIUHVXm4fdLv3J1wj
Pina/3doLcFB/V+Y4/Wxl+Kksbdr2c4StRcBuHyquOhcCPw0ZELoKBRfK3zy8yqH
IsrbNwf2nQKBgQCrYKlCkNaAhCW85mtpy7NRWIvf5chNR17ldNJOziS19Vt2Rjz8
o242EpwncoyJqF3cw3gYgadRsT7T+DORuLnhVfOPReDquenIeCKpOWXjwtE9r+CV
WguojqzRcY5Q1JfzsgtYX8jgzP+hEPWIrBR3glVFq7u6g1Uxo0CqllyTyQKBgENF
r79BEiq8mCn0groaO5vi4EbDHNE708r1xx2uWvztlnlX9mJoz+fSrDTxanVM3oy9
JmtXVuPqPs6WrFoeJFGZGc8+MU7mYZHrUxPjO8VyUHBs33PlUsOPferWCB5k5E7g
dQ+2YUUiOmKPAdFcen58qnynKlraOIFkQmIgx2jZAoGBAJDzPVKjWtIjNG7mHU7A
lgNzJLb3zozHJ0l0WY8OkWWWiV6MKkI9TiC6k1x3y77uTbd/wheRfDiplEpTjXF5
tBNzT/z3lWnuhF2WykXscD+vOzjZavpF6yW5sH9fiHYN05QnRxnyyypeG/j66e7r
y8IHgPnbYCb4+V/2m5VhPv+v
-----END PRIVATE KEY-----

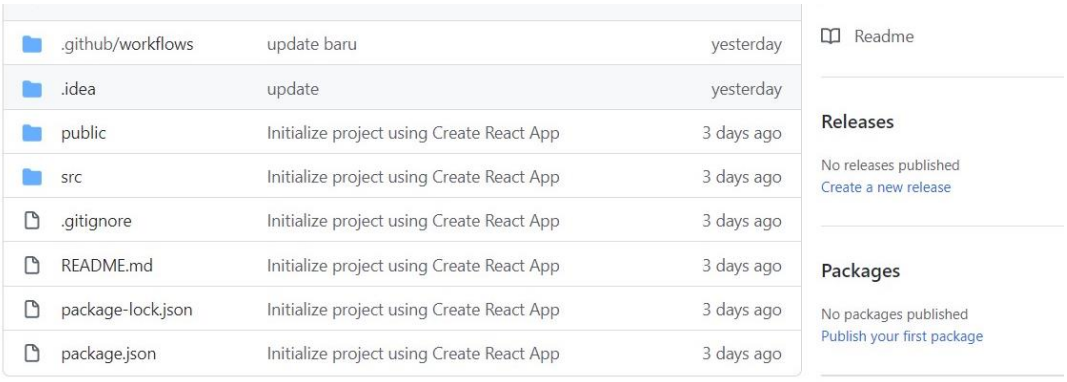


Akan muncul file public key

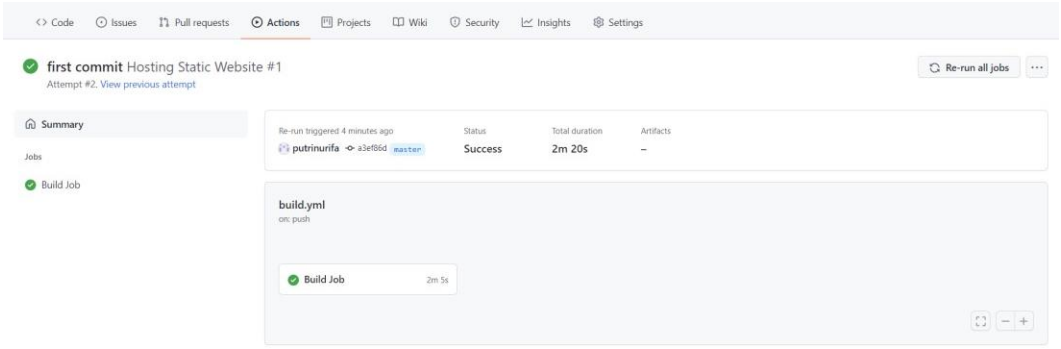




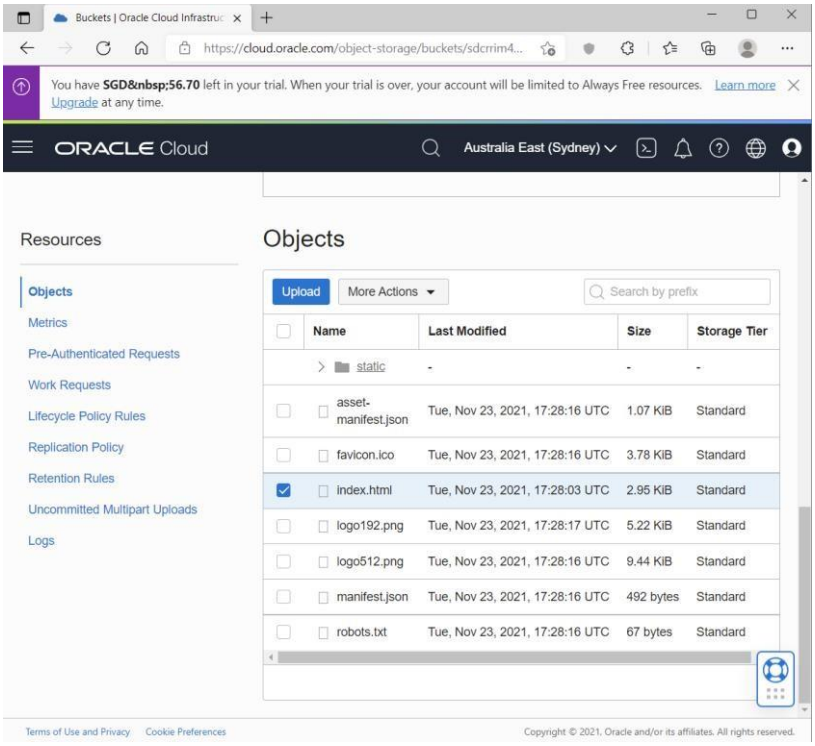
f. Lengkapi nilai semua secrets yang dibutuhkan. Kemudian push perubahan ke repository GitHub.



g. Perhatikan tab Actions pada halaman repository. Cek hasil proses deployment apakah terdapat kesalahan. Jika terjadi kesalahan, koreksi kembali nilai secrets yang dimasukkan.



h. Jika proses deployment sudah mendapatkan tanda centang hijau, silahkan kembali ke dashboard bucket object storage. Centang berkas index.html, kemudian tekan menu titik tiga dan pilih View Object Details. Perhatikan URL Path dengan pola sebagai berikut https://objectstorage..oraclecloud.com/n//b//o/index.html.

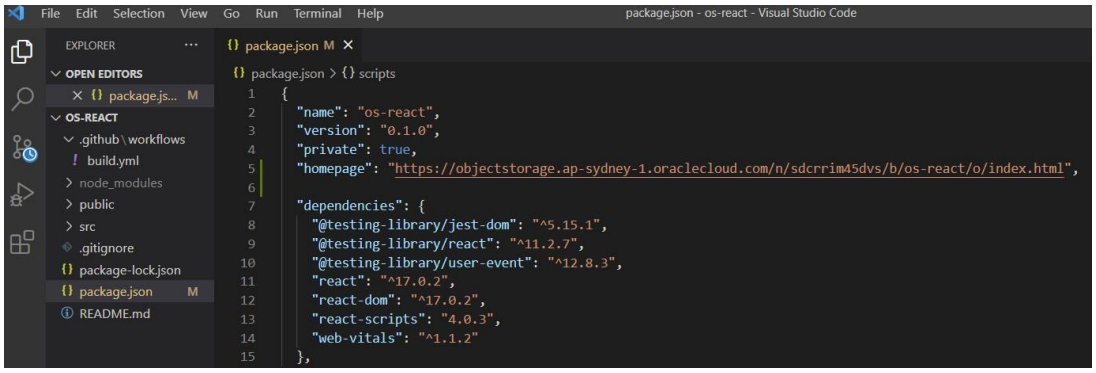


Hasil URL: https://objectstorage.ap-sydney-1.oraclecloud.com/n/sdcrrim45dvs/b/osreact/o/index.html

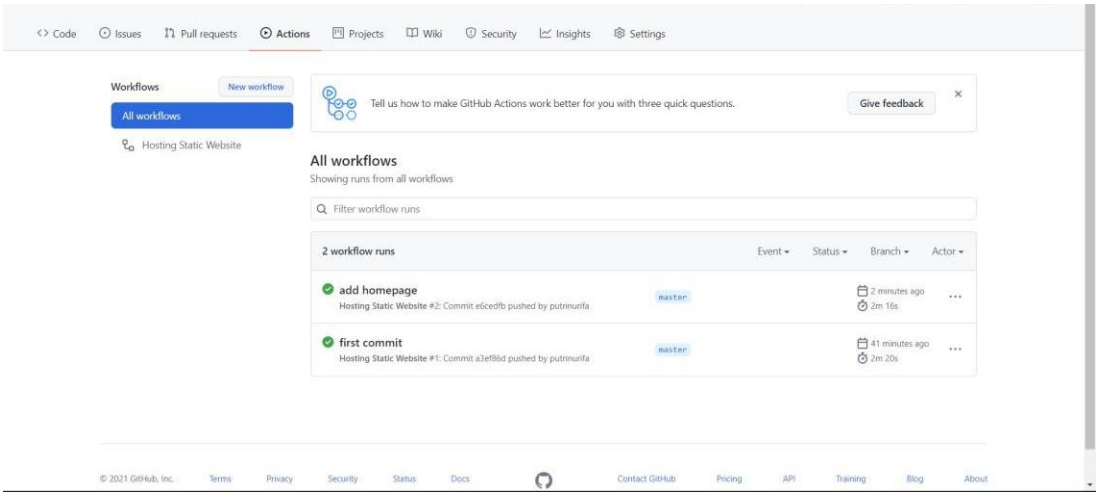
i. Salin URL Path tanpa nilai index.html, kemudian buka kembali project React. Cari berkas package.json dan tambahkan key “homepage” dengan

nilai URL Path pada Object Details. Ilustrasi berkas hasil akhir package.json dapat dilihat sebagai berikut.

```
{   "name": "hello-react",
    "version": "0.1.0",
    "private": true,  "homepage":
    "https://objectstorage.region.oraclecloud.com/n/namespace/b/bucket/o/",



}
```



Hasil perubahan setelah menambahkan key “homepage” pada package.json

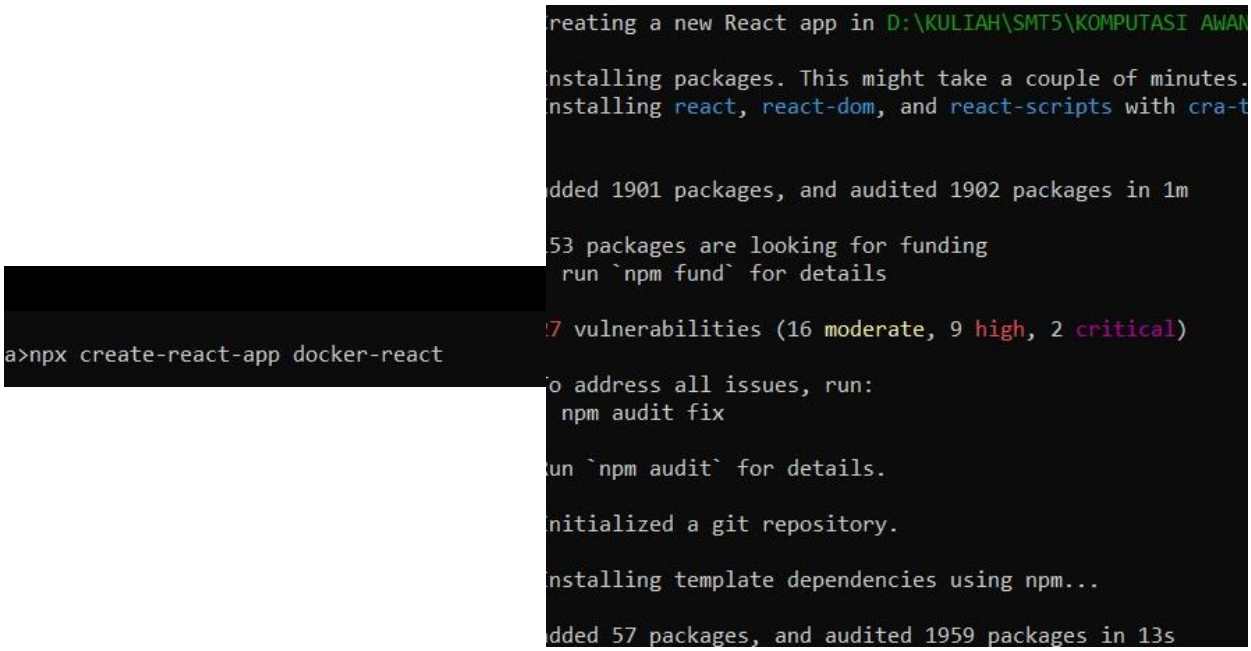


2. Deploy Simple Website Menggunakan Docker CI/CD Pipeline

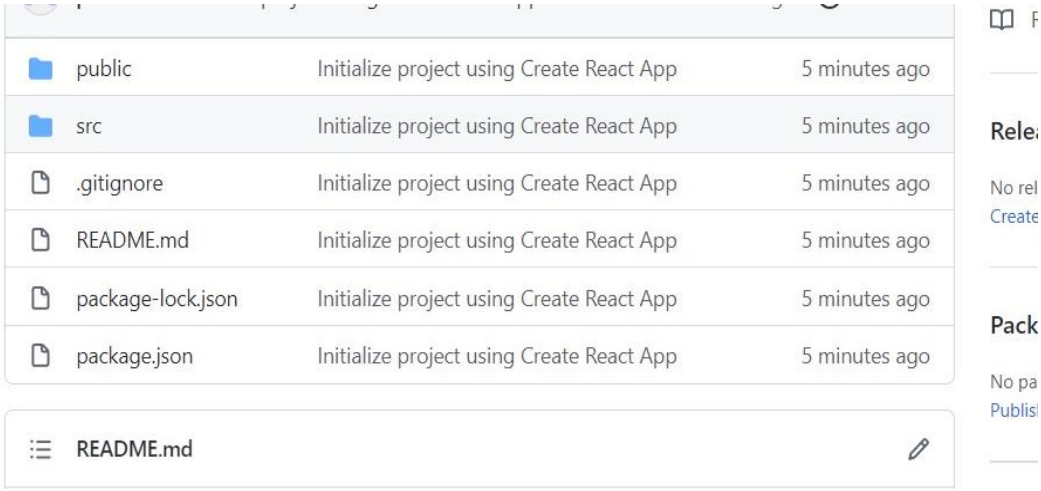
   Pada praktikum ini akan dilakukan proses deploy pada OCI Compute dengan menggunakan docker. Pada Langkah di bawah mengasumsikan telah dibuat VCN dengan akses pada port HTTP (80). Selain itu diasumsikan telah mempunyai akun docker hub.

2.1 **Menyiapkan Project Repository**

a. Buat baru project React dengan memasukkan perintah

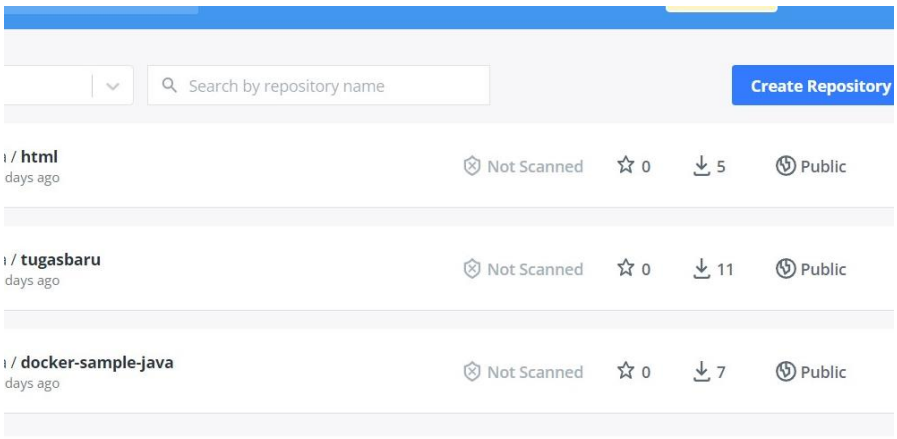


b. Buatlah project baru pada GitHub, dan push project React tersebut.

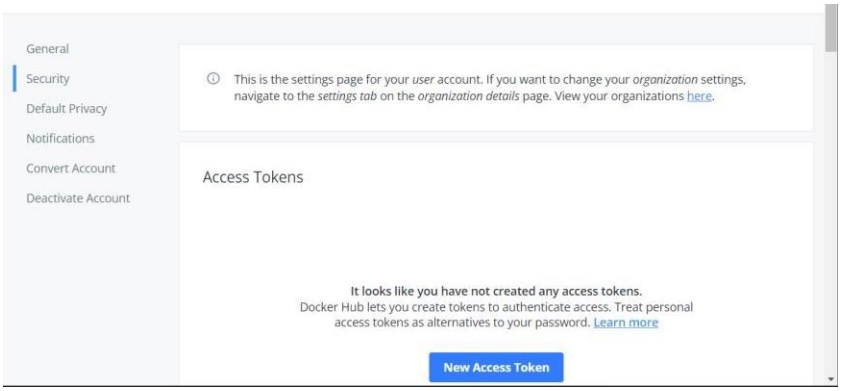


2.2 **Menyiapkan Akses Docker Hub**

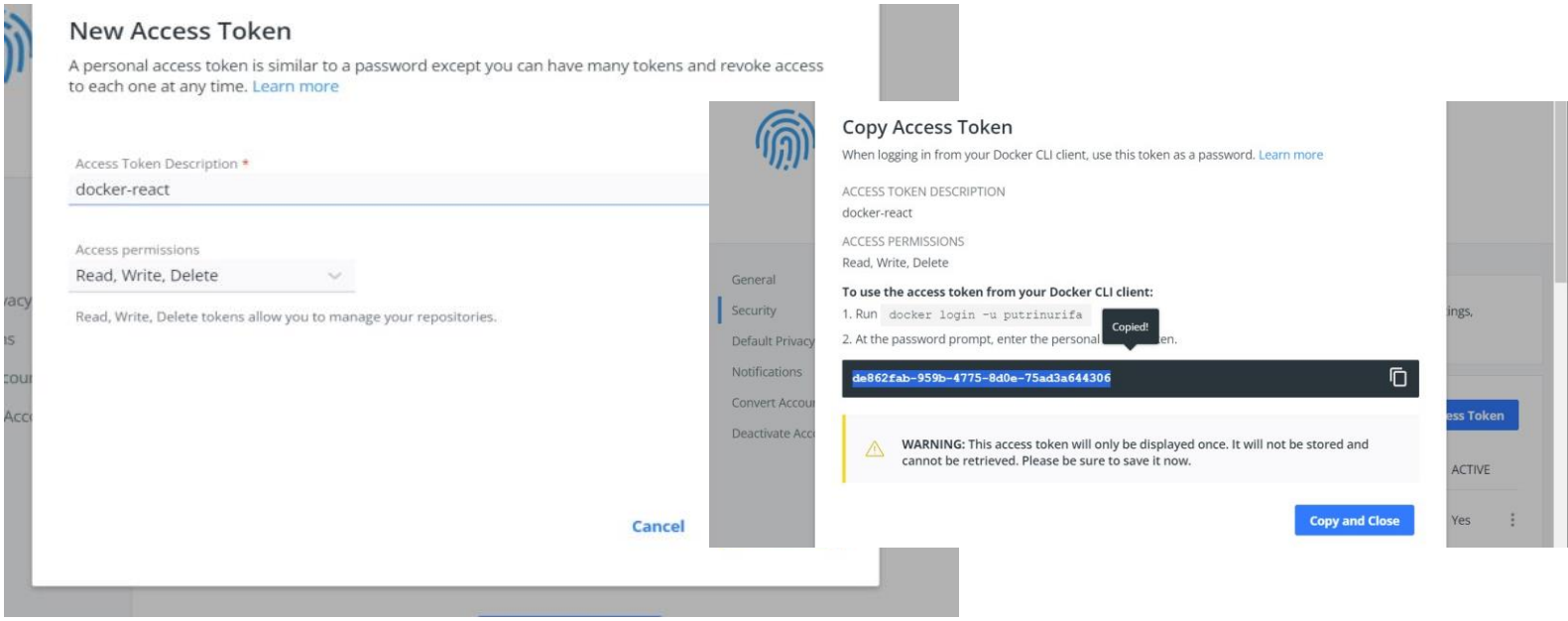
a. Silahkan login pada akun Docker pada https://hub.docker.com



b. Untuk memberikan akses push ke dalam registry, perlu dibuat Access Token. Silahkan masuk pada halaman https://hub.docker.com/settings/security kemudian tekan tombol New Access Token.



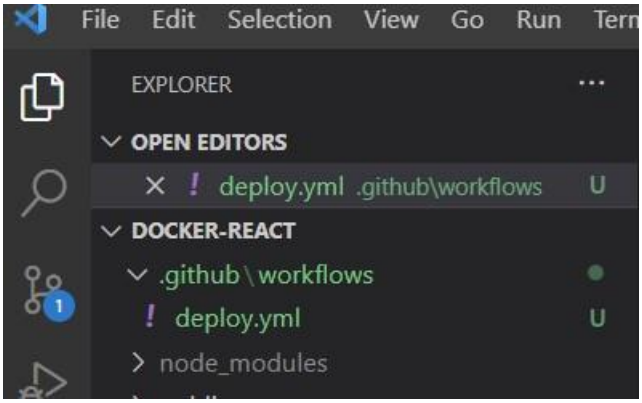
c. Masukkan GitHub Actions pada nama token, simpan nilai token yang telah didapatkan.

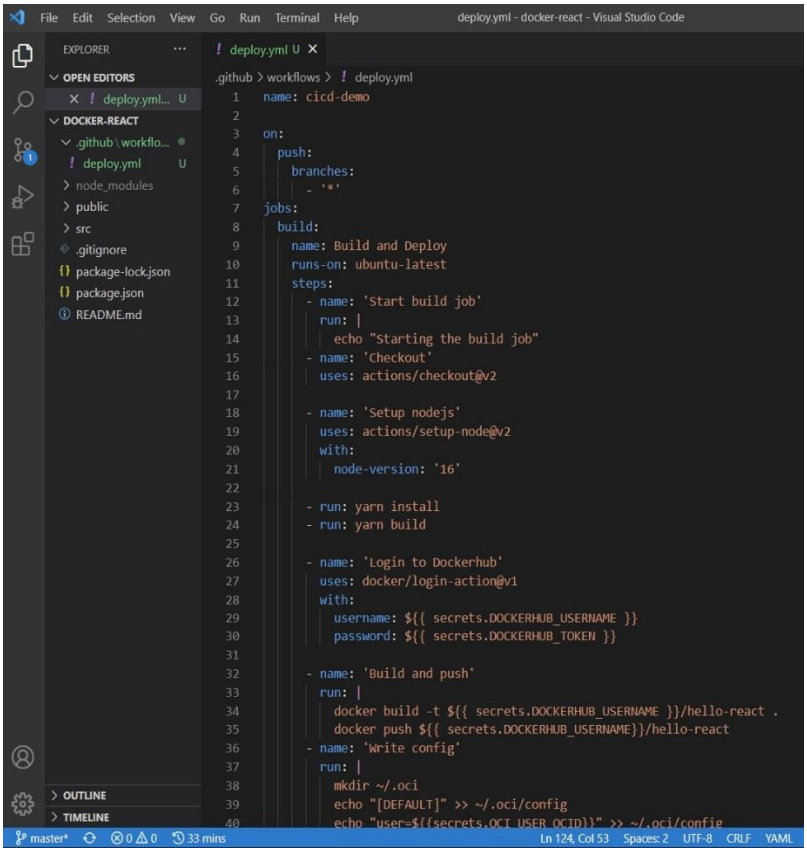


2.3 **Menambahkan GitHub Workflow**

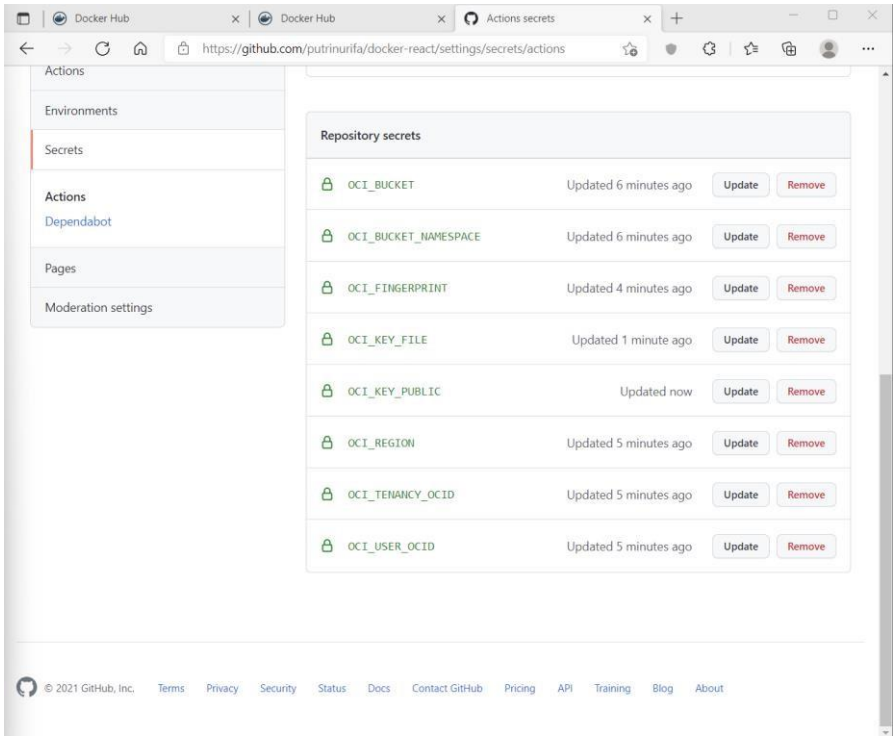
a. Buat sebuah file dengan berkas yaml, pada lokasi .github/workflows/deploy.yml.

b. Unduh konfigurasi berkas pada tautan https://github.com/dhanifudin/helloreactdocker/raw/master/.github/workflows/deploy.yml dan simpan pada lokasi tersebut.

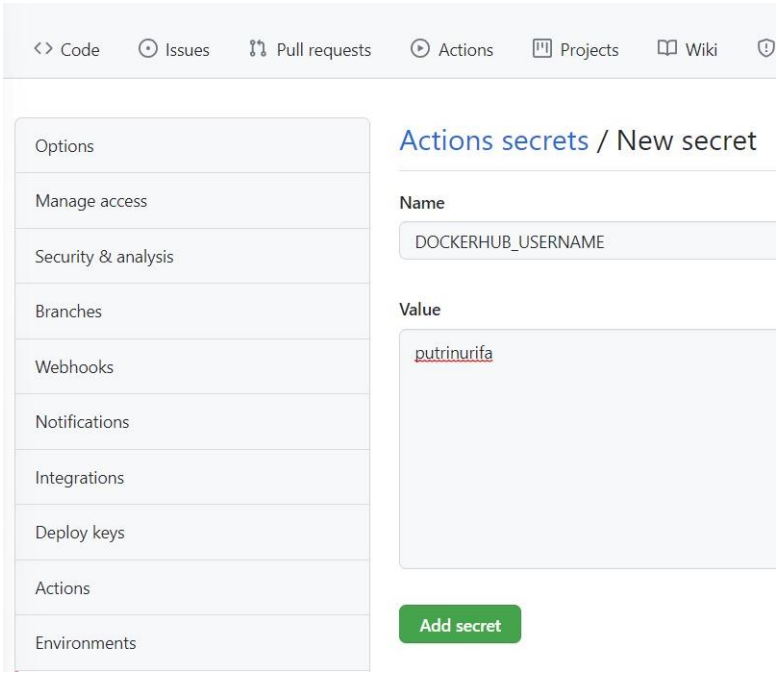
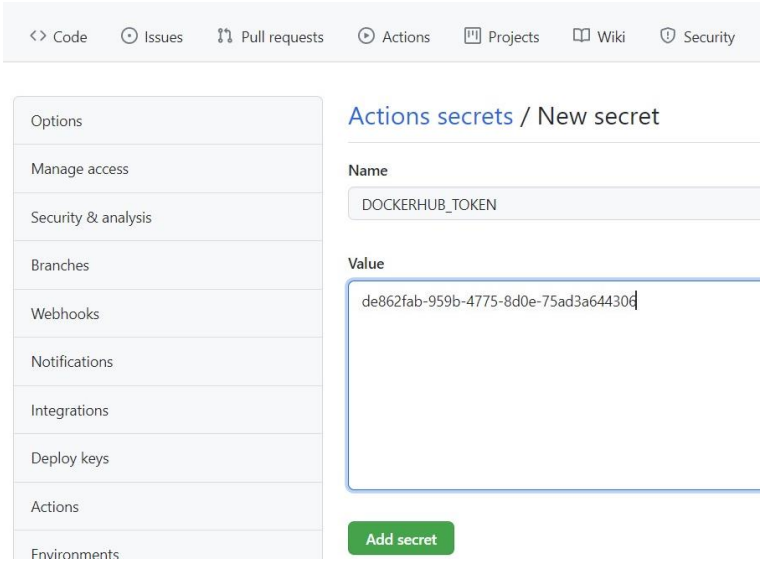


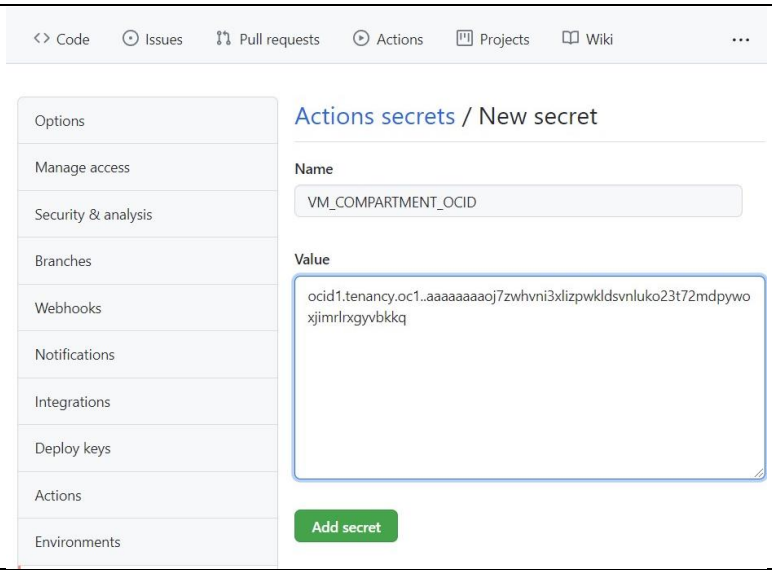
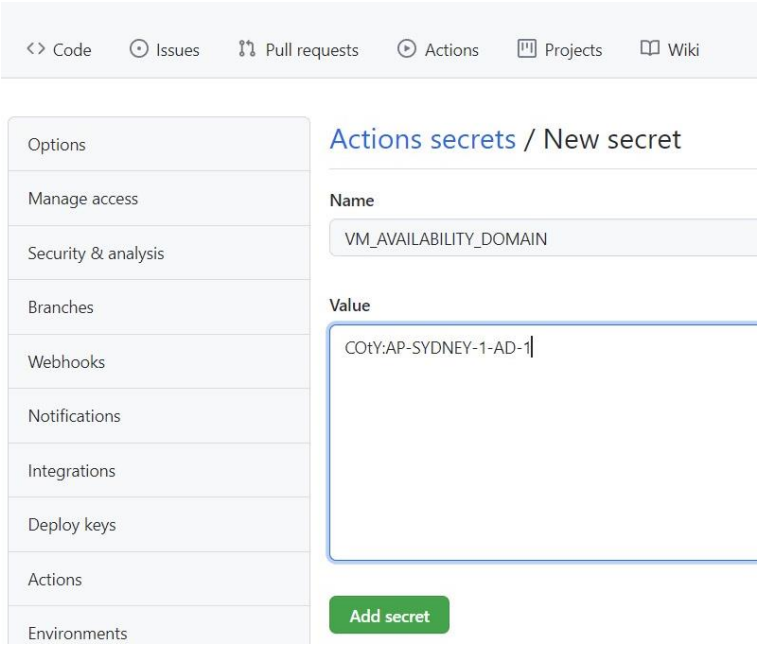


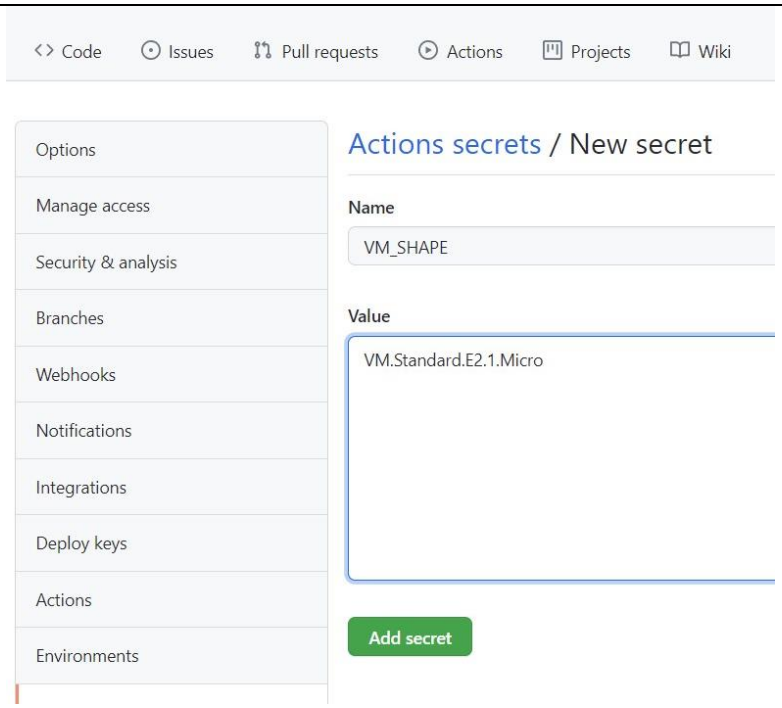
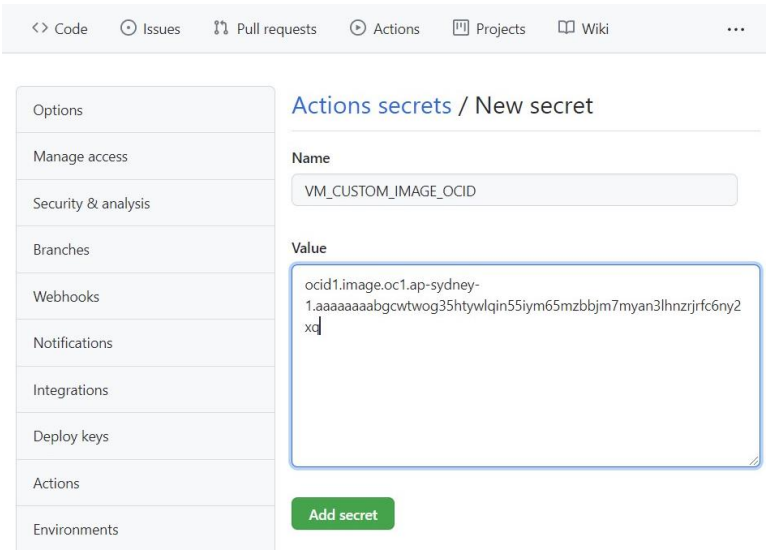
c. Tambahkan nilai-nilai secrets yang terdapat pada GitHub Workflow tersebut. Untuk nilai yang diawali prefix OCI_ dapat digunakan table sebelumnya sebagai referensi.

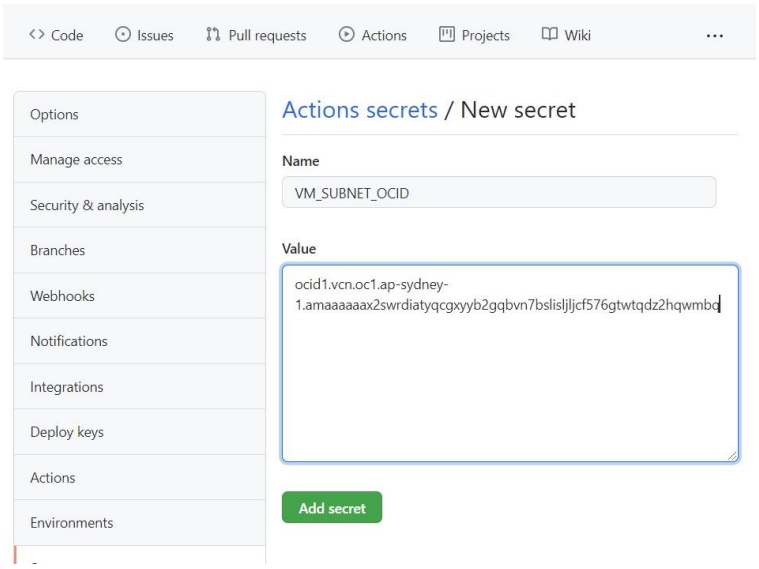


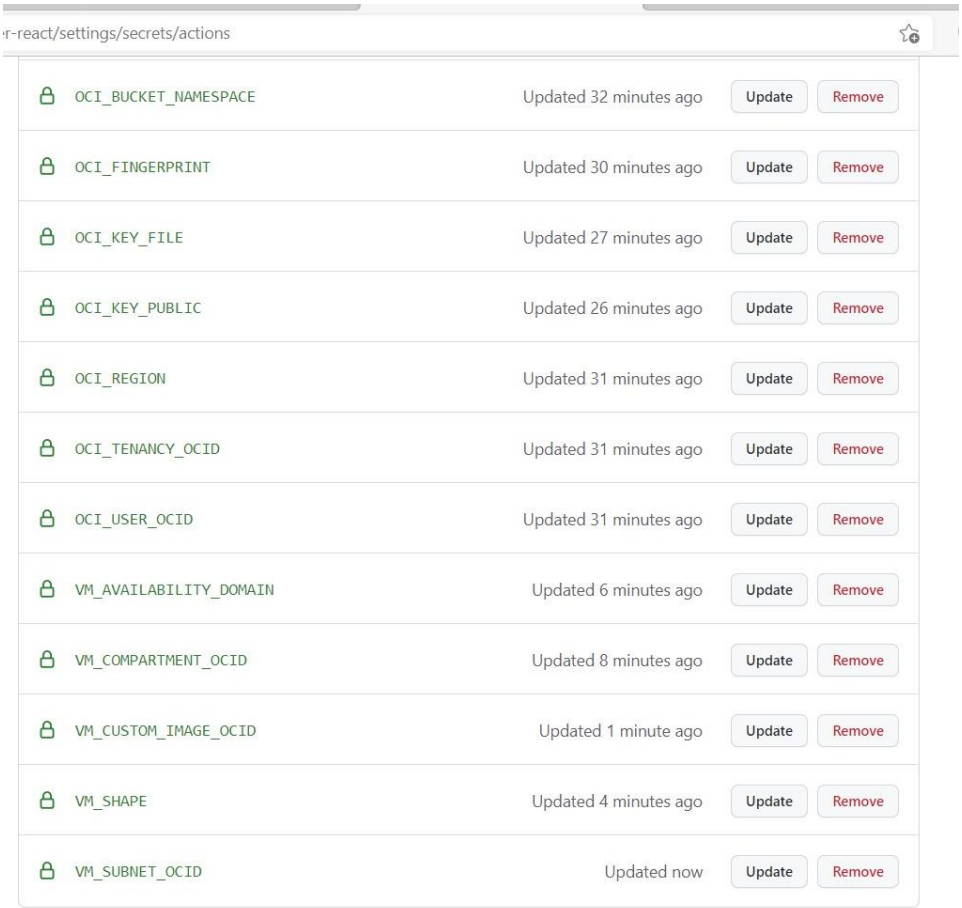
d. Perhatikan table berikut sebagai referensi untuk pengisian nilai secrets

Nama Key	**Keterangan**
DOCKERHUB_USERNAME	Username akun docker hub 
DOCKERHUB_TOKEN	Access token akun docker hub 

VM_COMPARTMENT_OCID	Nilai ini sama dengan tenancy id
	
VM_AVAILABILITY_DOMAIN	Pilih menu Compute -> Instances dan perhatikan nilai Availability domain yang berada di pojok kiri bawah. Contoh: lNJg:AP-MELBOURNE-1-AD-1 
VM_SHAPE	Pada praktikum ini digunakan shape VM.Standard.E2.1.Micro

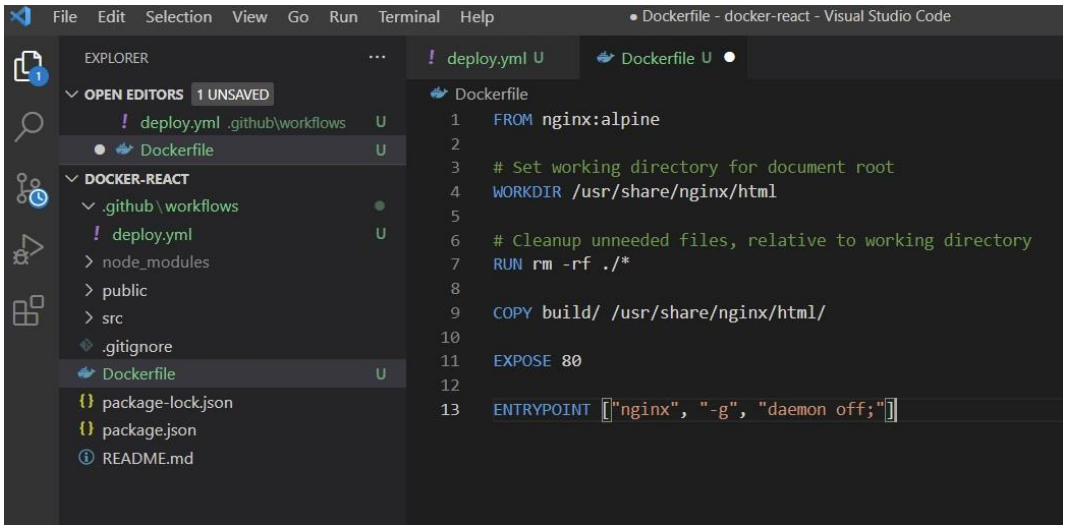
	
VM_CUSTOM_IMAGE_OCID	Buka https://docs.oracle.com/en-us/iaas/images/ Contoh: ocid1.image.oc1.apmelbourne 1.aaaaaaaalpnwbccpcvukmqfus2rn5jcexymxjjqbiybzshnhcy Fojitermyq 

VM_SUBNET_OCID	Pilih menu Networking -> Virtual Cloud Network masuk ke vcn yang telah dibuat. Informasi id dapat dilihat dalam Subnet Information pada subnet yang tersedia. 



2.4 **Menyiapkan Dockerfile**

a. Buatlah berkas dengan nama Dockerfile. Gunakan tautan berikut https://github.com/dhanifudin/hello-react-docker/blob/master/Dockerfile sebagai template.

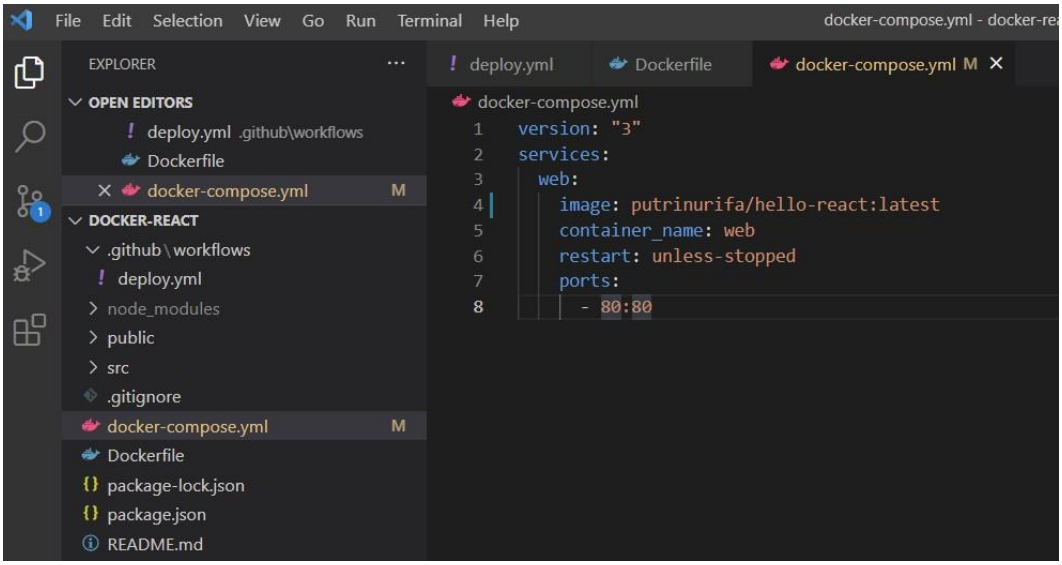


b. Pastikan penamaan berkas sudah sesuai.

2.5 **Menyiapkan Docker Compose**

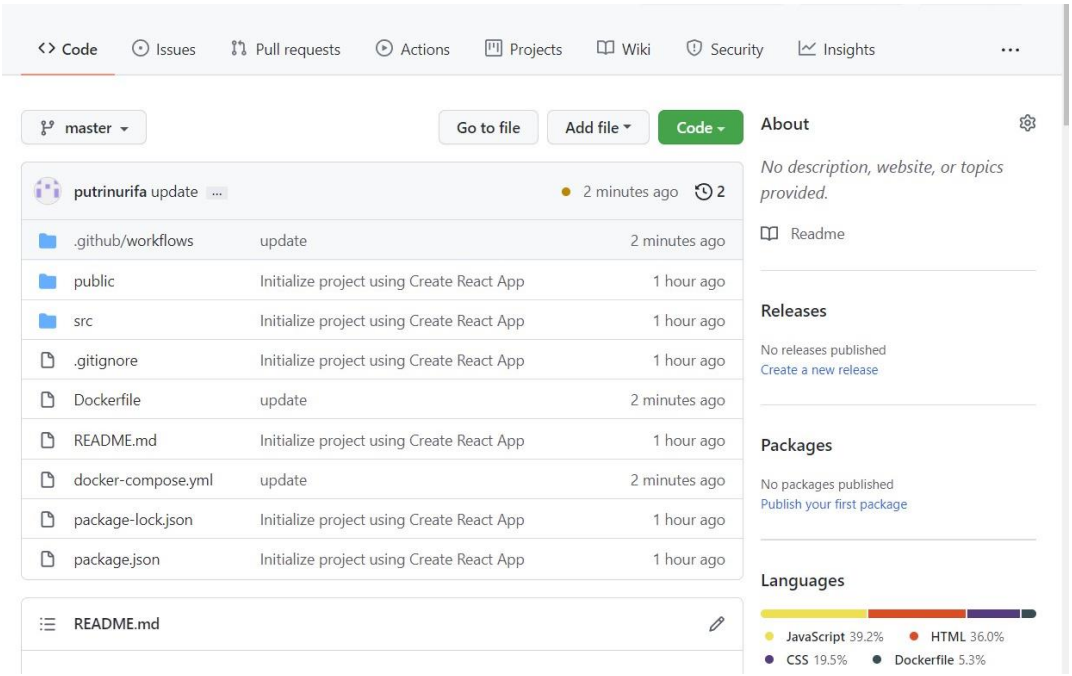
a. Buatlah berkas dengan nama docker-compose.yml

b. Gunakan tautan https://github.com/dhanifudin/hello-react-docker/blob/master/dockercompose.yml sebagai referensi untuk membuat docker-compose.yml. Jangan lupa untuk mengubah nilai image sesuai dengan username docker hub.

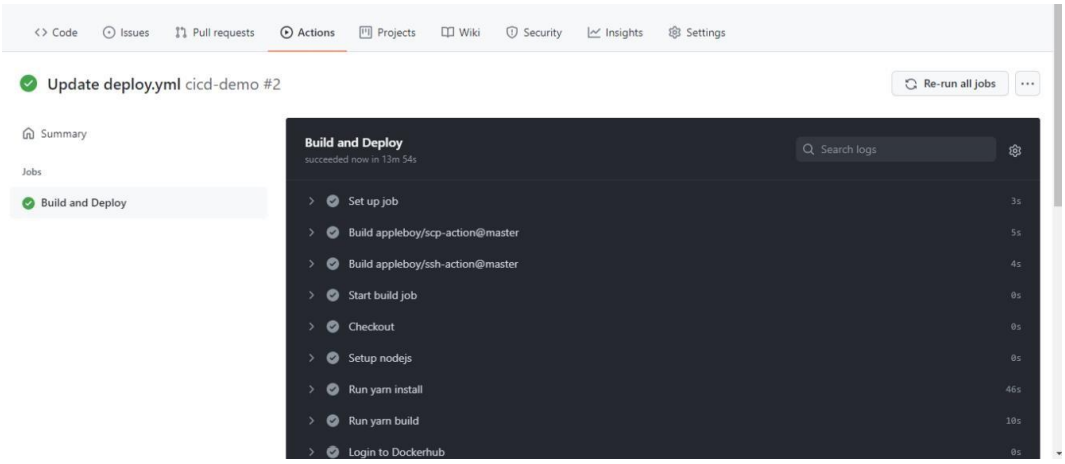


2.6 **Melakukan Deployment**

a. Setelah berkas deployment dan nilai secrets telah selesai diatur, lakukan push ke repository.



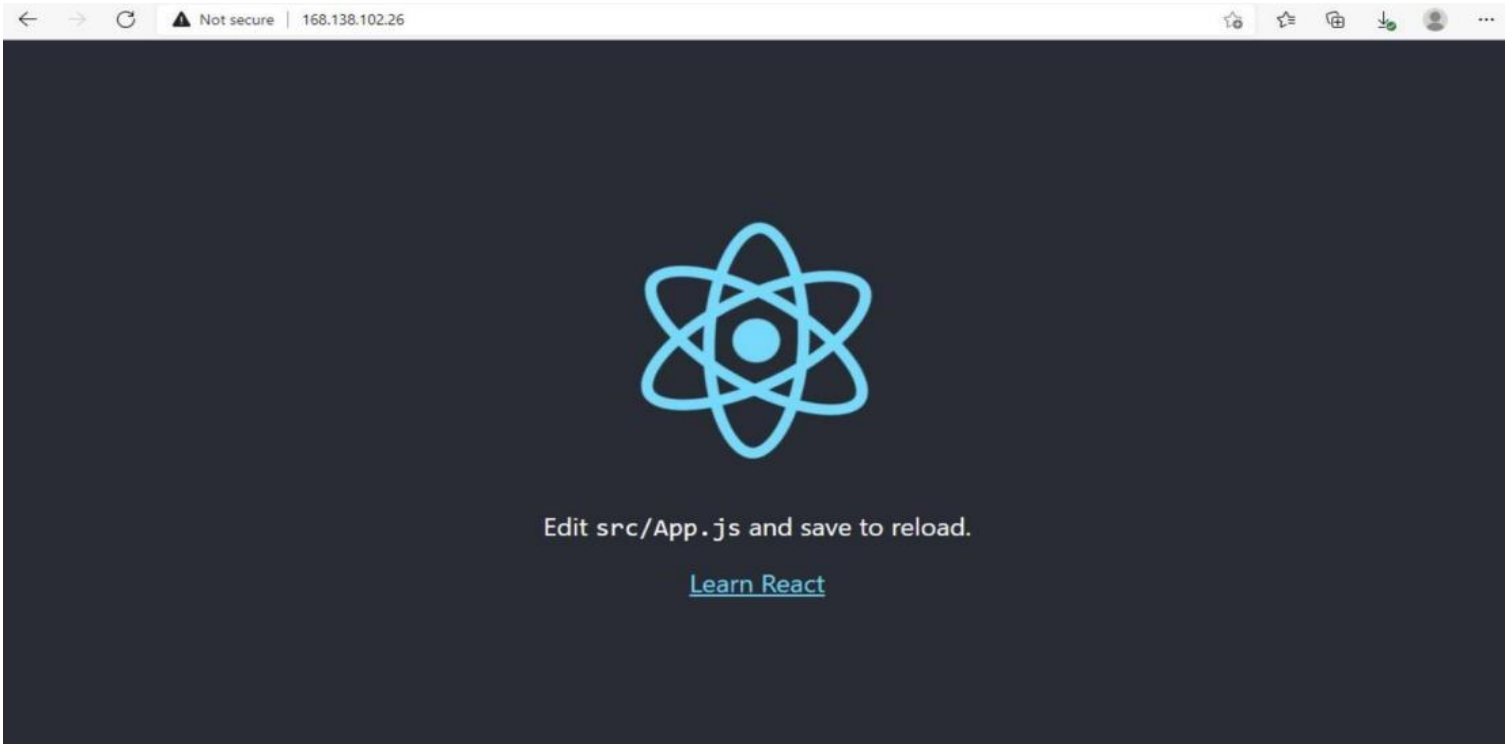
b. Amati log deployment pada tab Actions, dan pastikan proses menghasilkan tanda centang hijau. Jika terjadi error, periksa kembali konfigurasi dan nilai secrets yang dimasukkan.



c. Setelah proses deployment berhasil, untuk mendapatkan nilai IP public yang dapat diakses silahkan masuk ke dashboard pada halaman Compute Instances.

d. Pada hasil akhir, jika tidak terdapat kesalahan akan didapatkan halaman website seperti pada gambar berikut.

LINK WEBSITE : 168.138.102.26



Referensi

[1] J. Nickoloff, Docker in action. Shelter Island, NY: Manning Publications, Co, 2016.

[2] S. Gallagher, Mastering Docker: rethink what's possible with Docker - become an expert in the innovative containerization tool to unlock new opportunities in the way you use and deploy software. Birmingham Mumbai: Packt Publishing, 2015.

[3] J. Krochmalski, Docker and Kubernetes for Java developers: scale, deploy, and monitor multicontainer applications, First published. Birmingham Mumbai: Packt Publishing, 2017