**It’s Called ‘Legacy Code’ for a Reason**

Nama: Aryo Bama Wiratama

NIM: 13523088

1. Kesalahan pada Dockerfile
2. Tidak menginstall univcorn dan fast-api sehingga docker tidak bisa dijalankan karena tidak adanya depedensi yang dibutuhkan

**Solusi:**

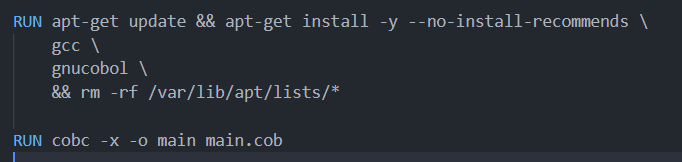
Menambakan line berikut pada Dockerfile



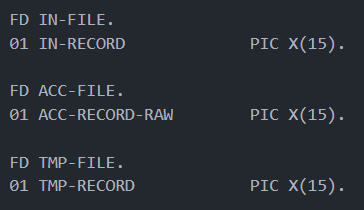
1. Main.cob belum dicompile dan dijadikan file executable sehingga service tidak bisa berjalan

**Solusi:**

Menambahkan perintah berikut pada Dockerfile



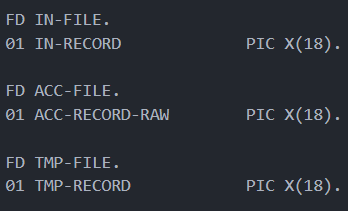
1. Kesalahan pada main.cob
2. Tidak mendefinisikan panjang variabel dengan benar



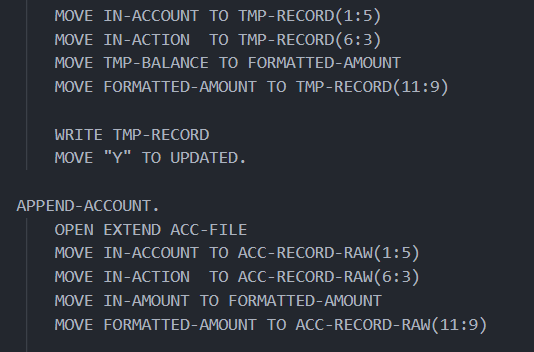
Variabel di atas digunakan untuk menyimpan informasi terkait akun bank yang berasal dari input.txt, memang terlihat baik-baik saja. Namun, setelah melihat isi input.txt, file tersebut ternyata menyimpan input yang memiliki panjang sebesar 18 per baris sehingga variabel di atas tidak cukup untuk menampung input

**Solusi:**

Menambah panjang variabel



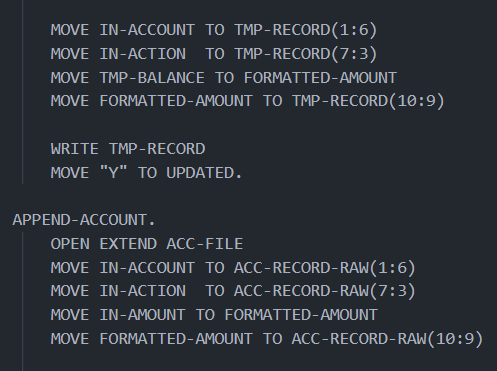
1. Kesalahan indexing



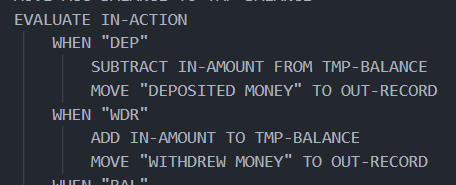
Dari kode di atas terlihat bahwa program mencoba mengambil nomor akun, action, dan balance akun. Namun, terdapat kesalahan index seharusnya nomor akun memiliki panjang 6, tetapi kode di atas mengambil nomor akun dari index 0 hingga 4 yang seharusnya dari index 0 hingga 5. Kesalahan ini juga terjadi saat mengambil action dan juga balance.

**Solusi:**

Ubah index yang sesuai



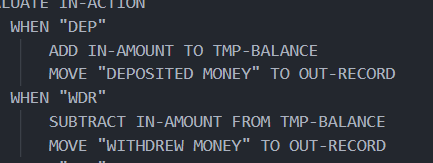
1. Operasi yang terbalik antara aksi *withdraw* dengan *deposit*



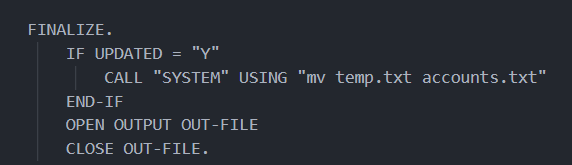
Deposit harusnya menambah saldo bukan mengurangi saldo. Begitu pula pada withdraw

**Solusi:**

Balik saja operasinya



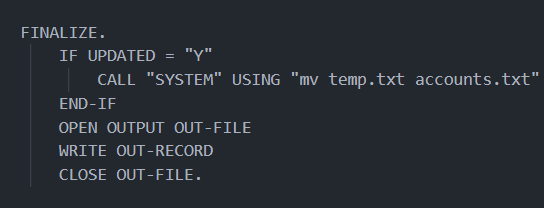
1. Lupa Write aktivitas



Saat FINALIZE seharusnya program menulis aktivitasi yang sudah dilakukan. Namun, di sini program langsung menutup sehingga aktivitas tidak tercatat

**Solusi:**

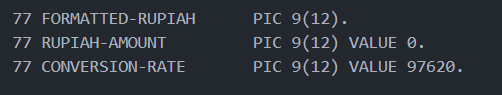
Tambahkan perintah WRITE seperti di bawah

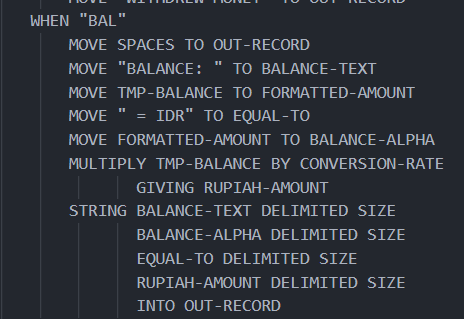


1. Bonus yang dikerjakan
2. Membuat konversi ke Indonesia Rupiah secara otomatis (asumsikan saat ini masih dalam bentuk Rai Stone dari Yap Island)

**Solusi:**

Menambahkan beberapa variabel dan operasi perkalian dan menampilkan hasil konversi, berikut pada source code



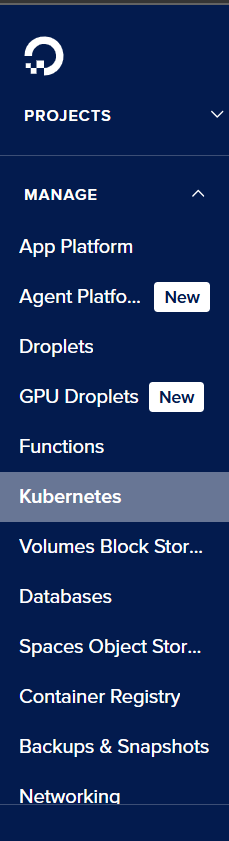


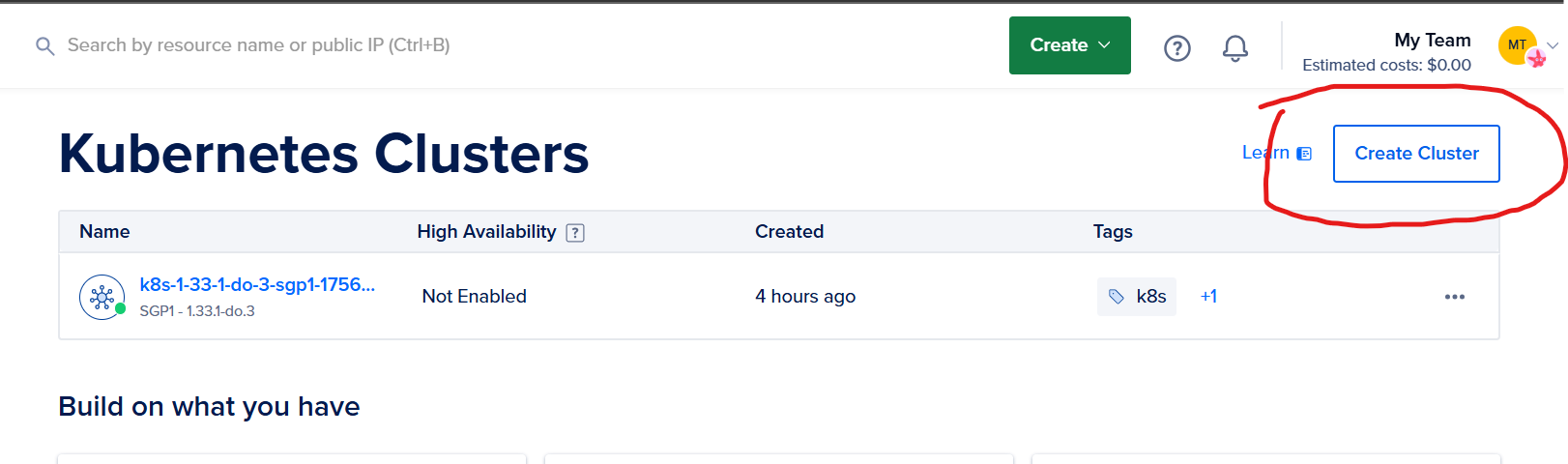
1. Deploy menggunakan kubernetes

**Solusi:**

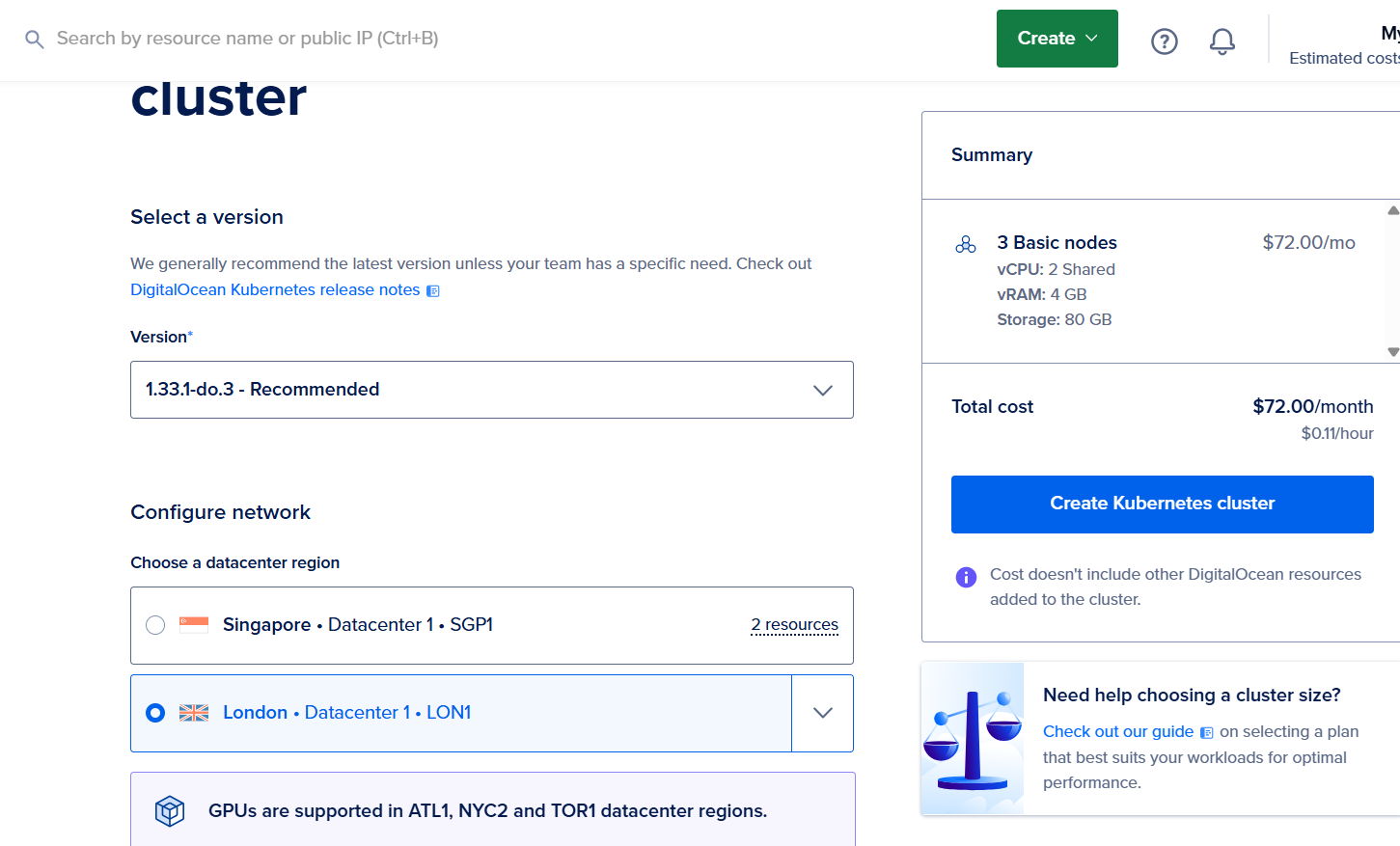
Di sini penulis menggunakan kubernetes pada digital ocean untuk kebutuhan deploy

1. Buat kubernetes cluster





Tentukan spesifikasi yang diperlukan, kemudian pilih “Create Kubernetes cluster”



1. Install kubectl di lokal

*# Untuk Linux/MacOS*

*curl -LO "https://dl.k8s.io/release/$(curl -L -s https://dl.k8s.io/release/stable.txt)/bin/linux/amd64/kubectl"*

*chmod +x kubectl*

*sudo mv kubectl /usr/local/bin/*

*# Untuk Windows, gunakan Chocolatey (pastikan Chocolatey sudah terinstall):*

*choco install kubernetes-cli*

1. Install doctl untuk cli DigitalOcean
2. Jalankan *doctl auth init* untuk masuk ke akun. Dapatkan token yang dibutuhkan pada url berikut <https://cloud.digitalocean.com/account/api/tokens>
3. Unduh konfig kubernetes dengan perintah berikut *doctl kubernetes cluster kubeconfig save <NAMA\_CLUSTER>*
4. Build dan push docker image (Pastikan sudah menginstall docker) dengan perintah berikut

*# Build image*

*docker build -t username/myapp:latest .*

*# Login Docker*

*docker login*

*# Push ke Docker Hub*

*docker push username/myapp:latest*

1. Buatlah YAML untuk deploy misal *deployment.yaml* . Kubernetes manifest dapat dilihat pada repository
2. Jalankan *kubectl apply -f deployment.yaml*
3. Jalankan perintah berikut untuk melihat status dan Public IP untuk akses web

*kubectl get deployments*

*kubectl get pods*

*kubectl get services*

Berikut url hasil deploy: <http://159.89.211.101/>