

Big Project

Migrasi Infrastruktur
On Premises Ke Cloud AWS



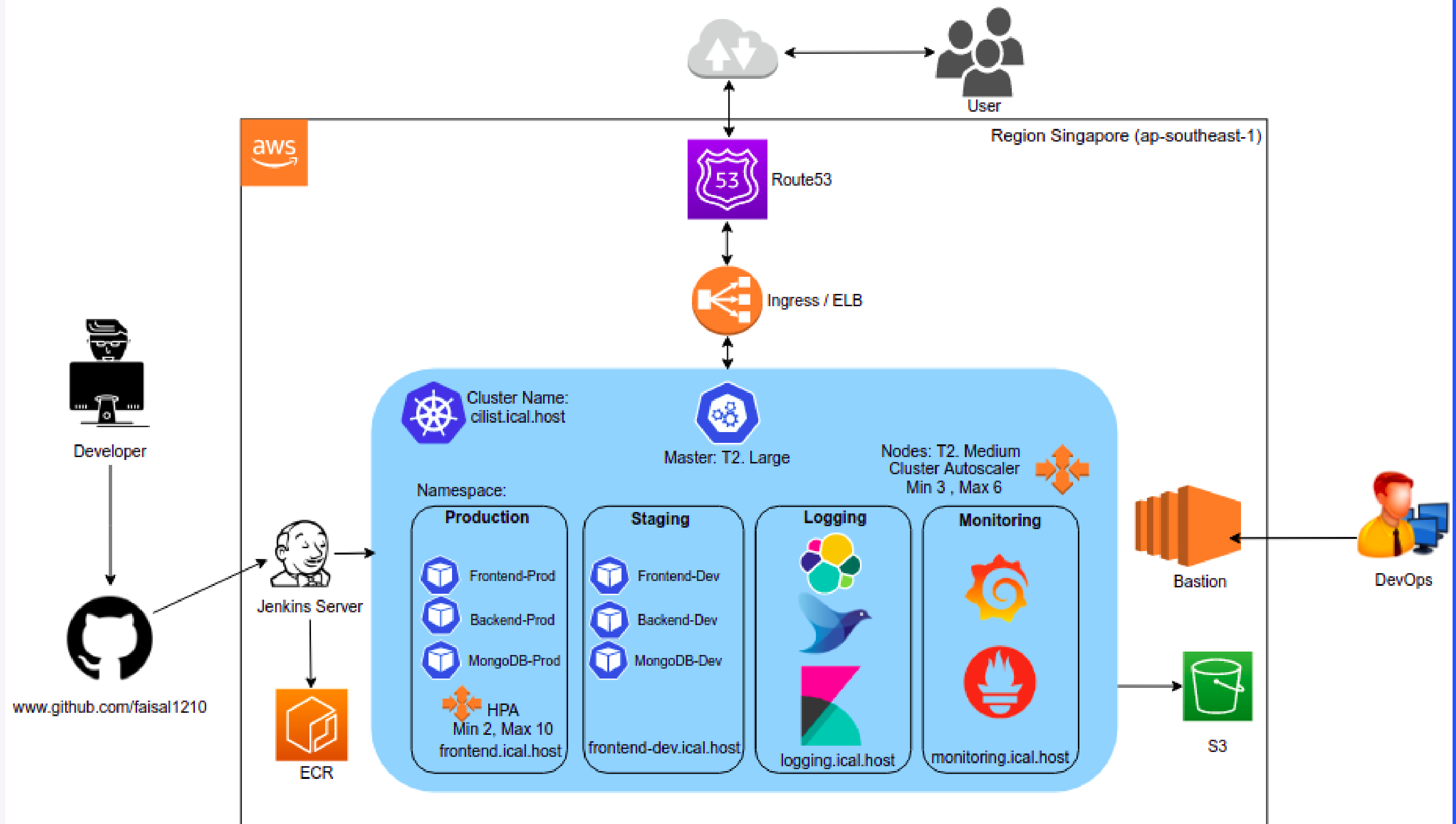
FAISAL - Sekolah DevOps Cilsy
faisal.alfatih90@gmail.com



Background

Sebuah perusahaan bernama Cilsy saat ini sedang melakukan development produk baru yang bernama Cilist (Cilsy List). Cilist ini merupakan aplikasi to-do list yang berbasis web. Aplikasi ini dibangun menggunakan MERN Stack (MongoDB, ExpressJS, ReactJS, dan NodeJS). Saat ini, frontend, backend, dan database baru disimpan di 1 vm yang sama, sehingga apabila terjadi error di vm tersebut, maka aplikasi tidak bisa digunakan sama sekali, serta seluruh sistem di Cilist belum menerapkan sistem DevOps seperti CI/CD.

Infrastructur Topology



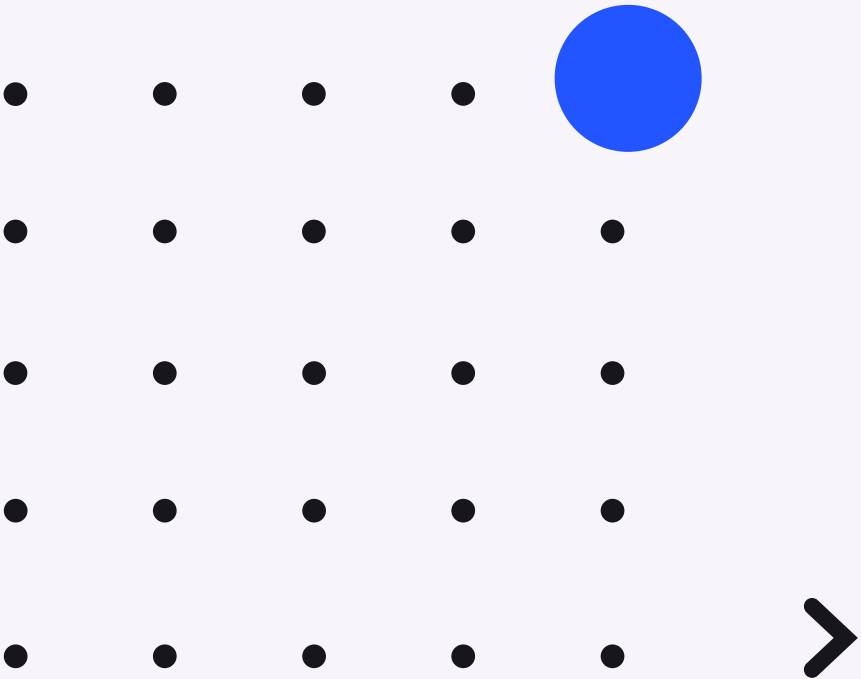
Minimum Budget

Services	Type / Specification	Estimate (per month)	Count	Total (USD)
EC2 (Bastion)	t2.micro, Memory 1GB , 1 vCPUs	10,69	1	10,69
	EBS SSD 20GB			
	Operating System Linux			
EC2 (Jenkins)	t2.medium, Memory 4GB , 2 vCPUs	42,75	1	42,75
	EBS SSD 20GB			
	Operating System Linux			
EC2 (Node K8s)	t2.medium, Memory 4GB , 2 vCPUs	42,75	3	128,25
	EBS SSD 20GB			
	Operating System Linux			
EC2 (Master K8s)	t2.large, Memory 8GB , 2 vCPUs	88,62	1	88,62
	EBS SSD 20GB			
	Operating System Linux			
S3	S3 Standard 10GB, with calculate data transfer 100GB	17,53	1	17,53
Classic Elastic Load Balancer	Process Data (100GB)	21,30	3	63,9
Elastic Container Registry	Outbound & Inbound: Internet (20GB)	5,5	4	22
Route 53	Hosted Zone	0,5	1	0,5
	Domain .com	12	1	12
		TOTAL (Per month)	\$	386,24
		TOTAL (Per 6 months)	\$	2317,44

Budgeting

Maximum Budget

Services	Type / Specification	Estimate (per month)	Count	Total (USD)
EC2 (Bastion)	t2.micro, Memory 1GB , 1 vCPUs	10,69	1	10,69
	EBS SSD 20GB			
	Operating System Linux			
EC2 (Jenkins)	t2.medium, Memory 4GB , 2 vCPUs	42,75	1	42,75
	EBS SSD 20GB			
	Operating System Linux			
EC2 (Node K8s)	t2.medium, Memory 4GB , 2 vCPUs	42,75	6	256,5
	EBS SSD 20GB			
	Operating System Linux			
EC2 (Master K8s)	t2.large, Memory 8GB , 2 vCPUs	88,62	1	88,62
	EBS SSD 20GB			
	Operating System Linux			
S3	S3 Standard 10GB, with calculate data transfer 100GB	17,53	1	17,53
Classic Elastic Load Balancer	Process Data (100GB)	21,30	3	63,9
Elastic Container Registry	Outbound & Inbound: Internet (20GB)	5,5	4	22
Route 53	Hosted Zone	0,5	1	0,5
	Domain .com	12	1	12
		TOTAL (Per month)	\$	514,49
		TOTAL (Per 6 months)	\$	3086,94



Provisioning Cluster with KOPS

initialing environment variabel

```
export bucket_name=bigproject-cilsy
export KOPS_CLUSTER_NAME=bigproject.ical.host
export KOPS_STATE_STORE=s3://${bucket_name}
```

create cluster k8s

```
kops create cluster --zones=ap-southeast-1a --node-count=3 --master-count=1 --node-size=t2.medium --master-size=t2.large
--name=${KOPS_CLUSTER_NAME} --ssh-public-key=~/.ssh/id_rsa.pub

kops update cluster --name ${KOPS_CLUSTER_NAME} --yes --admin

kops validate cluster
```

result

```
Validating cluster bigproject.ical.host

INSTANCE GROUPS
NAME                ROLE    MACHINETYPE  MIN  MAX  SUBNETS
master-ap-southeast-1a  Master  t2.large     1    1    ap-southeast-1a
nodes-ap-southeast-1a  Node    t2.medium    3    6    ap-southeast-1a

NODE STATUS
NAME                                                    ROLE  READY
ip-172-20-43-156.ap-southeast-1.compute.internal      node  True
ip-172-20-50-77.ap-southeast-1.compute.internal       master True
ip-172-20-60-46.ap-southeast-1.compute.internal       node  True
ip-172-20-61-214.ap-southeast-1.compute.internal      node  True

Your cluster bigproject.ical.host is ready
```



Manifest File

Production

Backend

- p-backend-cert.yaml
- p-backend-deployment.yaml
- p-backend-hpa.yaml
- p-backend-ingress.yaml
- p-backend-service.yaml

Database

- p-mongo-deployment.yaml
- p-mongo-pvc.yaml
- p-mongo-service.yaml

Frontend

- p-frontend-cert.yaml
- p-frontend-deployment.yaml
- p-frontend-hpa.yaml
- p-frontend-ingress.yaml
- p-frontend-service.yaml

Staging

Backend

- p-backend-deployment.yaml
- p-backend-ingress.yaml
- p-backend-service.yaml

Database

- p-mongo-deployment.yaml
- p-mongo-pvc.yaml
- p-mongo-service.yaml

Frontend

- p-frontend-deployment.yaml
- p-frontend-ingress.yaml
- p-frontend-service.yaml

Logging

elastic

- p-elasticsearch-configmap.yaml
- p-elasticsearch-statefulset.yaml
- p-elasticsearch-svc.yaml

fluentd

- p-fluentd-configmap.yaml
- p-fluentd-deployment.yaml
- p-fluentd-sa.yaml

kibana

- p-kibana-cert.yaml
- p-kibana-configmap.yaml
- p-kibana-deployment.yaml
- p-kibana-ingress.yaml
- p-kibana-svc.yaml

Monitoring

dashboard

- p-dashboard-cert.yaml
- p-dashboard-ingress.yaml
- p-recommended.yaml

grafana

- p-cert.yaml
- p-datasource-config.yaml
- p-deployment.yaml
- p-ingress.yaml
- p-pvc.yaml
- p-service.yaml

kube-state-metrics

- p-cluster-role-binding.yaml
- p-cluster-role.yaml
- p-deployment.yaml
- p-service-account.yaml
- p-service.yaml

prometheus

- p-cert.yaml
- p-cluster-role.yaml
- p-config-map.yaml
- p-deployment.yaml
- p-ingress.yaml
- p-pvc.yaml
- p-service.yaml

Others

Cluster-Autoscaler

- cluster-autoscaler-autodiscover.yaml
- config-cluster-for-auto-scaler
- config-policies-cluster

HPA

- metrics-server.yaml

Ingress-Controller

- mandatory.yaml
- patch-configmap-l4.yaml
- service-l4.yaml

- cert-manager.yaml

```
root@bastion:~# kubectl get all -n production
NAME                                READY   STATUS    RESTARTS   AGE
pod/backend-prod-764c85cc7c-5b9bt   1/1     Running   0           3h1m
pod/backend-prod-764c85cc7c-tgl7l   1/1     Running   0           3h1m
pod/frontend-prod-564bffc44-9zttb   1/1     Running   0           77m
pod/frontend-prod-564bffc44-j9bfn   1/1     Running   0           77m
pod/mongo-prod-67d467b6-dk2cl       1/1     Running   0           12h

NAME                                PORT(S)    AGE
service/backend-prod-service        4000/TCP   12h
service/frontend-prod-service       3000/TCP   12h
service/mongo-prod-service          27017:30456/TCP 12h

NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/backend-prod        2/2     2             2           3h1m
deployment.apps/frontend-prod       2/2     2             2           77m
deployment.apps/mongo-prod          1/1     1             1           12h

NAME                                DESIRED   CURRENT   READY   AGE
replicaset.apps/backend-prod-764c85cc7c 2         2         2       3h1m
replicaset.apps/frontend-prod-564bffc44 2         2         2       77m
replicaset.apps/mongo-prod-67d467b6     1         1         1       12h

NAME                                REFERENCE                               TARGETS          MINPODS   MAXPODS   REPLICAS   AGE
horizontalpodautoscaler.autoscaling/backend-prod-hpa  Deployment/backend-prod  11%/90%, 0%/90%  2         10        2           3h
horizontalpodautoscaler.autoscaling/frontend-prod-hpa  Deployment/frontend-prod  47%/90%, 0%/90%  2         10        2           65m
```

< Production



Staging >

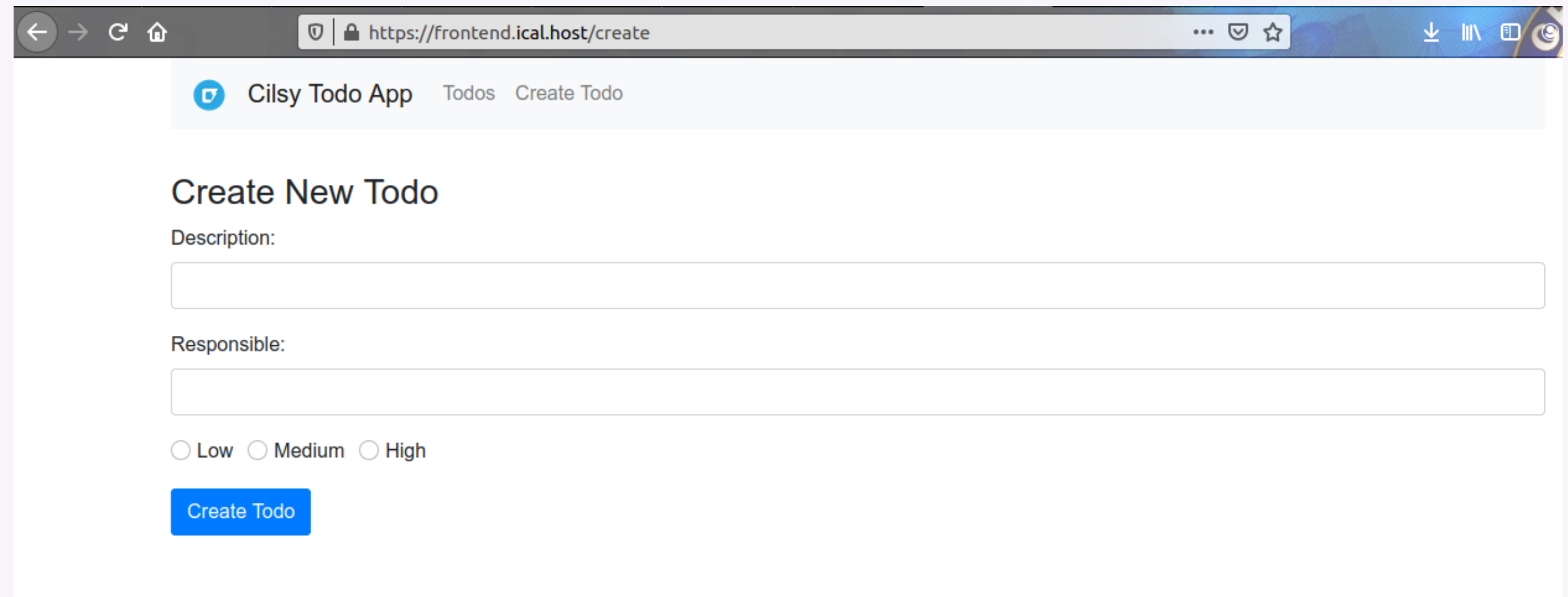
```
root@bastion:~# kubectl get all -n staging
NAME                                READY   STATUS    RESTARTS   AGE
pod/backend-dev-fcc95fcb5-4qlkc     1/1     Running   0           171m
pod/frontend-dev-667d66dcdd-wtzt7   1/1     Running   0           12h
pod/mongo-dev-77896d98f4-ct25v      1/1     Running   0           12h

NAME                                TYPE                                CLUSTER-IP    EXTERNAL-IP
service/backend-dev-service          ClusterIP      100.66.54.86  <none>
service/frontend-dev-service         ClusterIP      100.64.88.46  <none>
service/mongo-dev-service            LoadBalancer  100.70.85.147 a597f6f797fb443729e9c843d8104e70-981655818.ap-southeast-1.elb.amazonaws.com

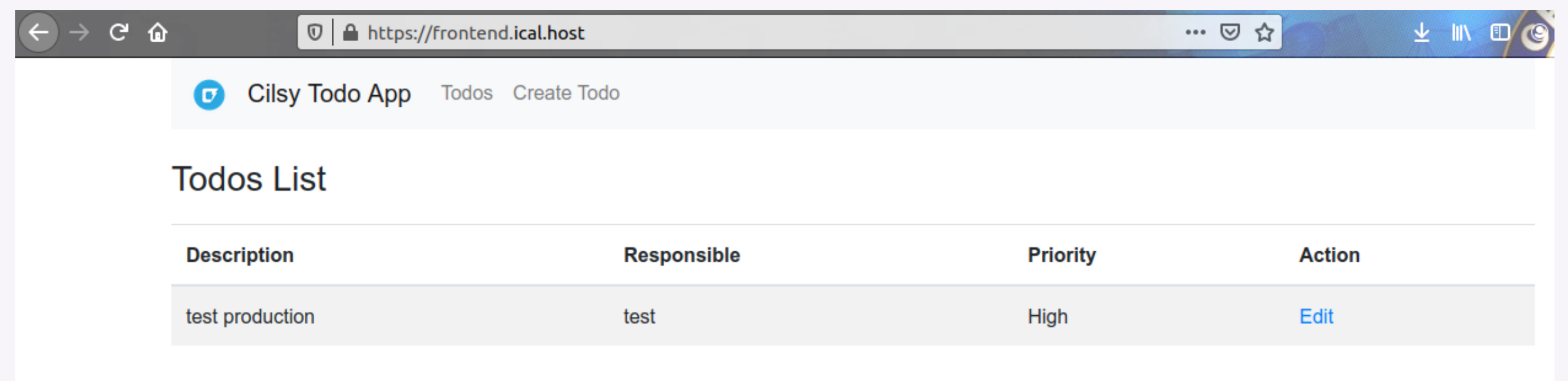
NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/backend-dev          1/1     1             1           12h
deployment.apps/frontend-dev         1/1     1             1           12h
deployment.apps/mongo-dev            1/1     1             1           12h

NAME                                DESIRED   CURRENT   READY   AGE
replicaset.apps/backend-dev-fcc95fcb5 1         1         1       12h
replicaset.apps/frontend-dev-667d66dcdd 1         1         1       12h
replicaset.apps/mongo-dev-77896d98f4    1         1         1       12h
```

https://frontend.ical.host



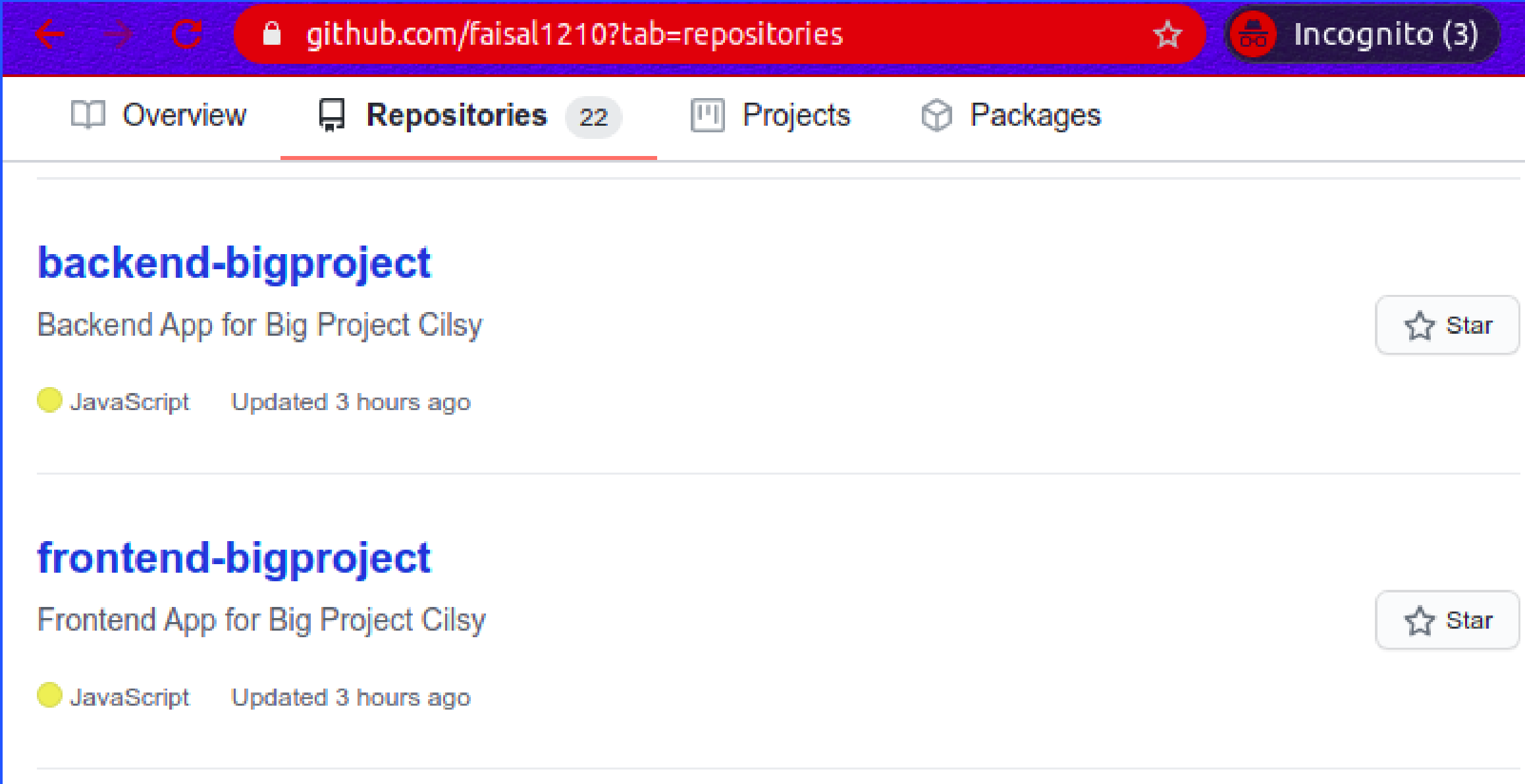
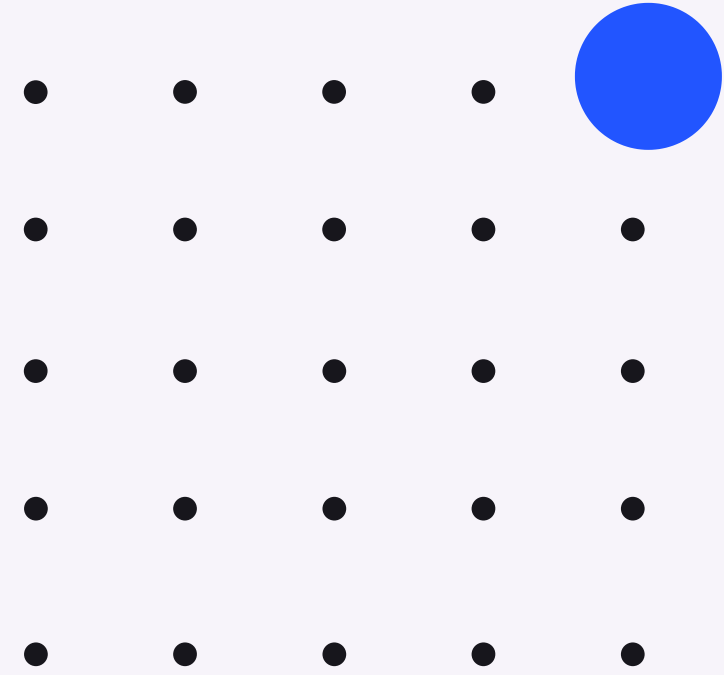
The screenshot shows a web browser window with the URL `https://frontend.ical.host/create`. The page title is "Cilsy Todo App" and the breadcrumb is "Todos Create Todo". The main heading is "Create New Todo". Below it, there are two input fields: "Description:" and "Responsible:". Under the "Responsible:" field, there are three radio buttons labeled "Low", "Medium", and "High". At the bottom, there is a blue button labeled "Create Todo".



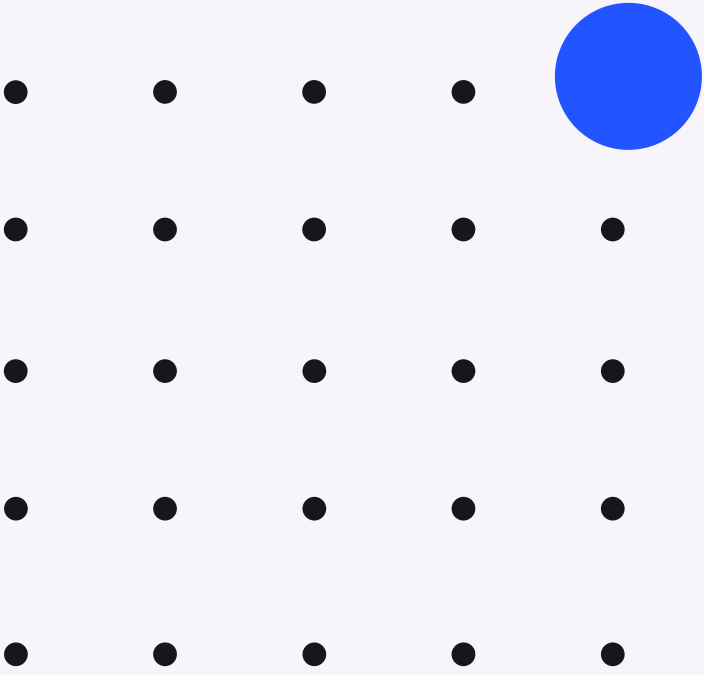
The screenshot shows a web browser window with the URL `https://frontend.ical.host`. The page title is "Cilsy Todo App" and the breadcrumb is "Todos Create Todo". The main heading is "Todos List". Below the heading is a table with the following data:

Description	Responsible	Priority	Action
test production	test	High	Edit

Code Repository



Images Repository



aws

Services

Search for services, features, marketplace products, and docs

[Alt+S]

Faisal

Singapore

Support

PrivatePublic

Private repositories (10)

Find repositories

View push commandsDeleteEditCreate repository

	Repository name	URI	Created at	Tag immutability	Scan on push	Encryption type
<input type="radio"/>	backend-dev	768876311475.dkr.ecr.ap-southeast-1.amazonaws.com/backend-dev	Apr 08, 2021 03:54:15 PM	Disabled	Disabled	AES-256
<input type="radio"/>	frontend-dev	768876311475.dkr.ecr.ap-southeast-1.amazonaws.com/frontend-dev	Apr 08, 2021 03:53:43 PM	Disabled	Disabled	AES-256
<input type="radio"/>	backend-prod	768876311475.dkr.ecr.ap-southeast-1.amazonaws.com/backend-prod	Apr 08, 2021 03:53:22 PM	Disabled	Disabled	AES-256
<input type="radio"/>	frontend-prod	768876311475.dkr.ecr.ap-southeast-1.amazonaws.com/frontend-prod	Apr 08, 2021 03:53:03 PM	Disabled	Disabled	AES-256

Jenkins

search

?

3

Admin

log out

Dashboard

Big Project - Cilsy

Up

Status

Configure

New Item

Delete Folder

People

Build History

Project Relationship

Check File Fingerprint

Move

Big Project - Cilsy

add description

All

+

S	W	Name ↓	Last Success	Last Failure	Last Duration	Fav
		Backend Dev Pipeline	5 hr 54 min - #2	N/A	6.3 sec	★
		Backend Prod Pipeline	11 hr - #3	N/A	14 sec	★
		Frontend Dev Pipeline	5 hr 48 min - #2	N/A	5.3 sec	★
		Frontend Prod Pipeline	11 hr - #11	N/A	7.2 sec	★

Icon: S M L

Legend

Atom feed for all

Atom feed for failures

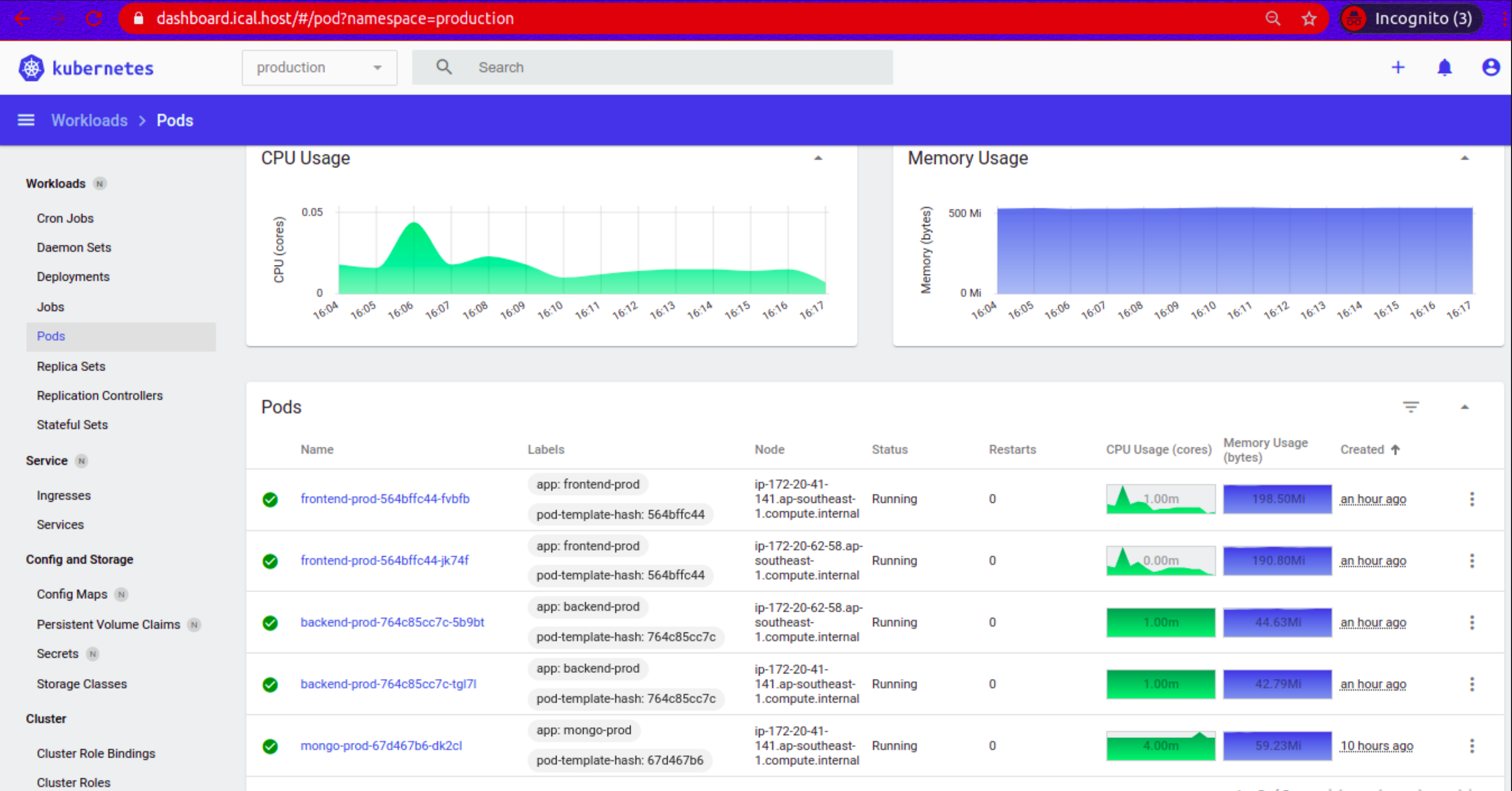
Atom feed for just latest builds

```
env.DOCKER_REGISTRY = '768876311475.dkr.ecr.ap-southeast-1.amazonaws.com'
env.DOCKER_IMAGE_NAME = 'frontend-prod'
pipeline {
  agent any
  stages {
    stage('Git Pull from Github') {
      steps {
        git url: 'https://github.com/faisal1210/frontend-bigproject.git'
      }
    }
    stage('Build Docker Image') {
      steps {
        sh "docker build -t $DOCKER_REGISTRY/$DOCKER_IMAGE_NAME:${BUILD_NUMBER} ."
      }
    }
    stage('Push Docker Image to ECR') {
      steps {
        sh "docker push $DOCKER_REGISTRY/$DOCKER_IMAGE_NAME:${BUILD_NUMBER}"
      }
    }
    stage('Deploy To Kubernetes Cluster') {
      steps {
        sh'''sed -i "15d" p-frontend-deployment.yaml'''
        sh'''sed -i "14 a '\\\\'          image: $DOCKER_REGISTRY/$DOCKER_IMAGE_NAME:${BUILD_NUMBER}" p-frontend-deployment.yaml && sed -i "s/'/'/" p-frontend-deployment.yaml'''
        sh "kubectl apply -f p-frontend-deployment.yaml"
      }
    }
    stage('Remove Docker Image in local') {
      steps {
        sh "docker rmi $DOCKER_REGISTRY/$DOCKER_IMAGE_NAME:${BUILD_NUMBER}"
      }
    }
    stage('Clean Workspace') {
      steps {
        cleanWs()
      }
    }
  }
}
```

CICD Jenkins

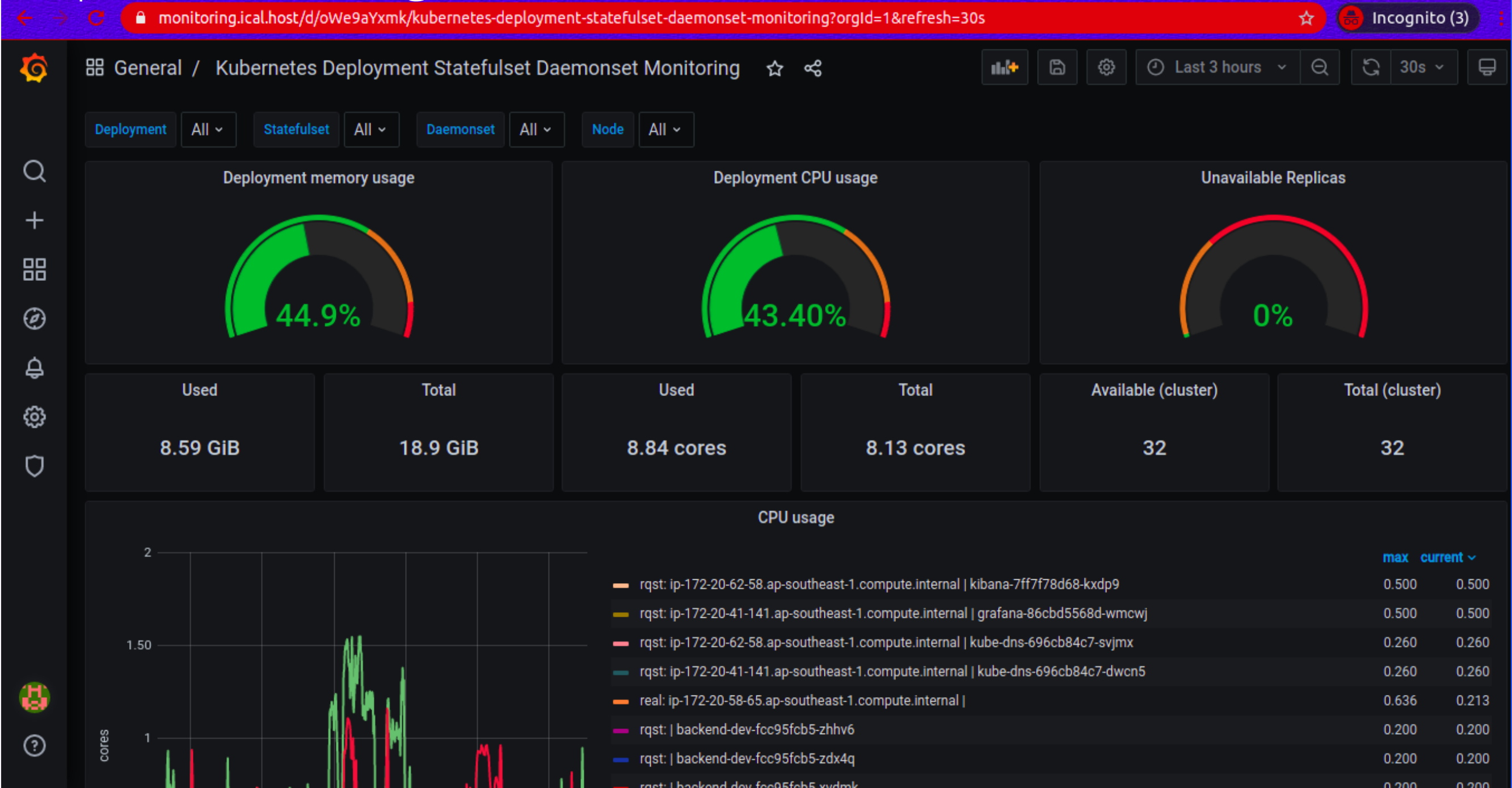
Kubernetes Dashboard

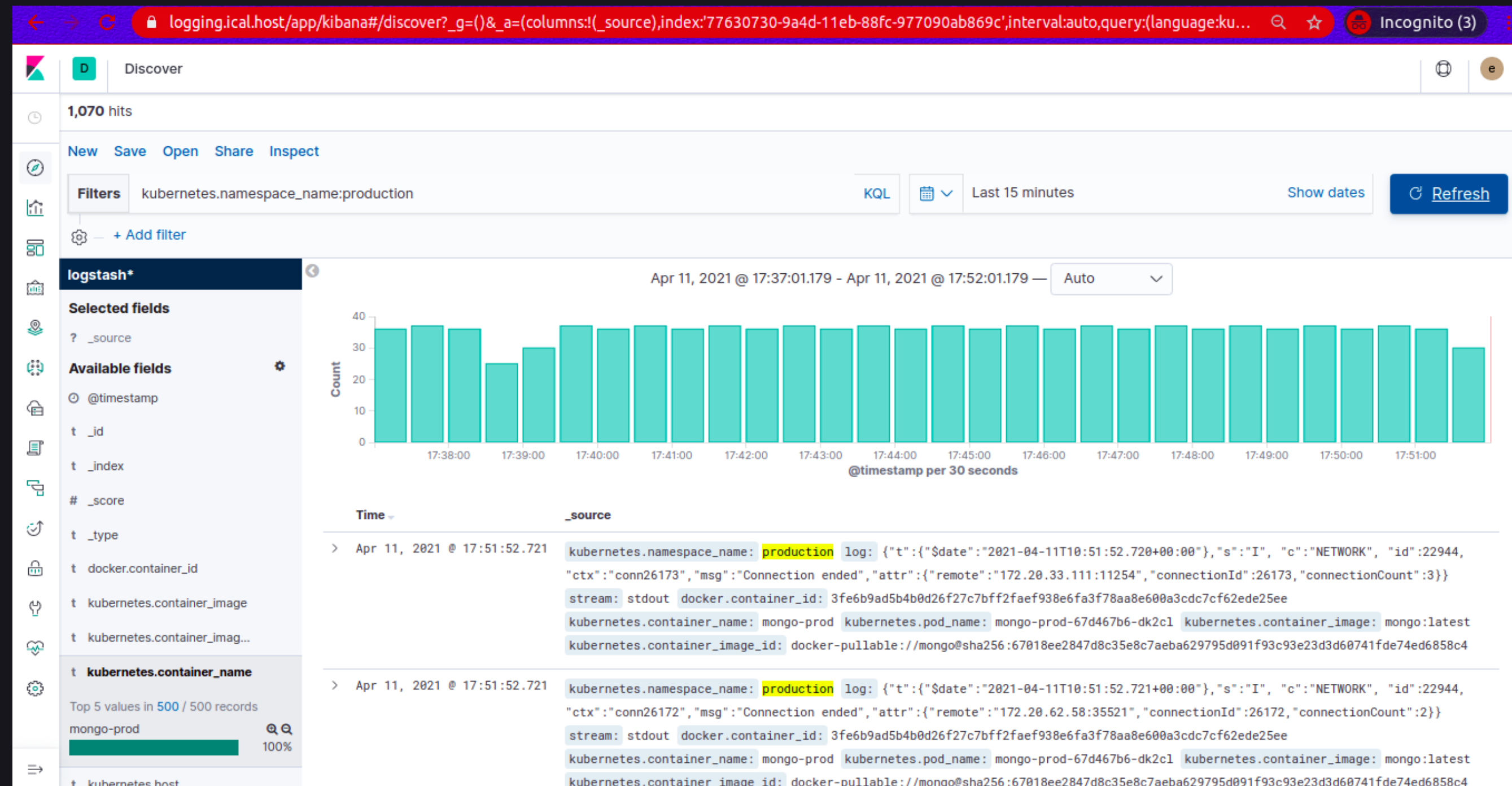
https://dashboard.ical.host



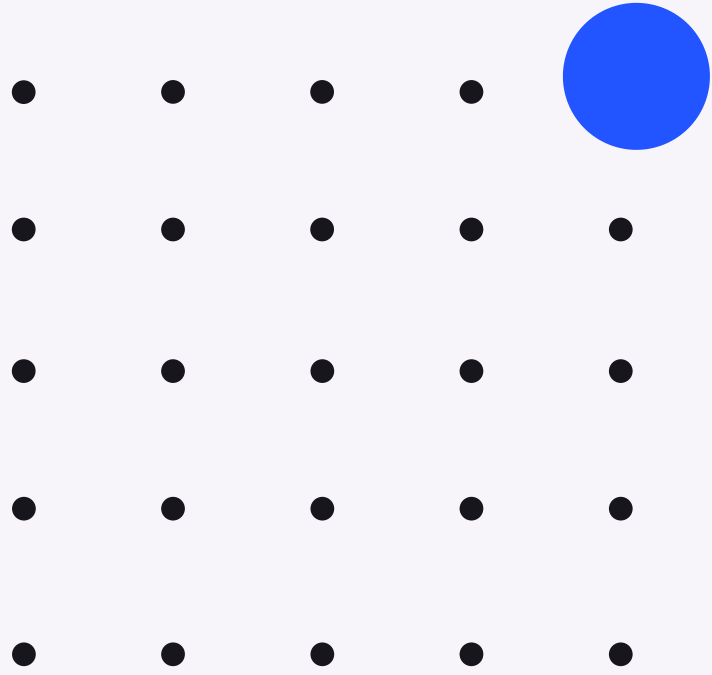
Monitoring

https://monitoring.ical.host





Cluster Autoscaler Testing



root@bastion: ~ 76x14

I0411 08:07:37.306016 1 scale_down.go:463] Node ip-172-20-41-141.ap-southeast-1.compute.internal is not suitable for removal - cpu utilization too big (0.980000)
I0411 08:07:37.306027 1 scale_down.go:459] Node ip-172-20-33-111.ap-southeast-1.compute.internal - cpu utilization 0.910000
I0411 08:07:37.306033 1 scale_down.go:463] Node ip-172-20-33-111.ap-southeast-1.compute.internal is not suitable for removal - cpu utilization too big (0.910000)
I0411 08:07:37.306139 1 static_autoscaler.go:410] Scale down status: unneededOnly=true lastScaleUpTime=2021-04-11 08:04:26.286963779 +0000 UTC m=+37280.631143137 lastScaleDownDeleteTime=2021-04-10 21:43:30.397836826 +0000 UTC m=+24.742016103 lastScaleDownFailTime=2021-04-10 21:43:30.397838031 +0000 UTC m=+24.742017308 scaleDownForbidden=false isDeleteInProgress=false

root@bastion: ~ 87x25

root@bastion:~# kubectl scale deployment backend-dev -n staging --replicas=20
deployment.apps/backend-dev scaled
root@bastion:~#
root@bastion:~#

root@bastion: ~ 103x21

Every 1.0s: kubectl get pods -n staging | grep backend-dev bastion: Sun Apr 11 08:07:40 2021

backend-dev-fcc95fcb5-4qlkc	1/1	Running	0	3m25s
backend-dev-fcc95fcb5-64dd6	1/1	Running	0	3m25s
backend-dev-fcc95fcb5-6vx6k	1/1	Running	0	3m25s
backend-dev-fcc95fcb5-9q7l2	1/1	Running	0	3m25s
backend-dev-fcc95fcb5-c65xn	0/1	ContainerCreating	0	3m25s
backend-dev-fcc95fcb5-cpmhj	1/1	Running	0	3m25s
backend-dev-fcc95fcb5-d6zn2	1/1	Running	0	9h
backend-dev-fcc95fcb5-f88xz	1/1	Running	0	3m25s
backend-dev-fcc95fcb5-hr4cx	0/1	ContainerCreating	0	3m25s
backend-dev-fcc95fcb5-htjb7	0/1	ContainerCreating	0	3m25s
backend-dev-fcc95fcb5-kpkgl	1/1	Running	0	3m25s
backend-dev-fcc95fcb5-pr644	1/1	Running	0	3m25s
backend-dev-fcc95fcb5-q97pp	0/1	ContainerCreating	0	3m25s
backend-dev-fcc95fcb5-rxvw8	1/1	Running	0	3m25s
backend-dev-fcc95fcb5-sk6zf	0/1	ContainerCreating	0	3m25s
backend-dev-fcc95fcb5-vwbnm	1/1	Running	0	3m25s
backend-dev-fcc95fcb5-wxvnr	0/1	ContainerCreating	0	3m25s
backend-dev-fcc95fcb5-xvdmk	0/1	ContainerCreating	0	3m25s
backend-dev-fcc95fcb5-zdx4q	0/1	ContainerCreating	0	3m25s

root@bastion: ~ 90x18

Every 1.0s: kubectl get nodes bastion: Sun Apr 11 08:07:40 2021

NAME	STATUS	ROLES	AGE	VERSION
ip-172-20-33-111.ap-southeast-1.compute.internal	Ready	node	18h	v1.19.9
ip-172-20-41-141.ap-southeast-1.compute.internal	Ready	node	18h	v1.19.9
ip-172-20-51-58.ap-southeast-1.compute.internal	Ready	node	23s	v1.19.9
ip-172-20-58-65.ap-southeast-1.compute.internal	Ready	master	18h	v1.19.9
ip-172-20-62-58.ap-southeast-1.compute.internal	Ready	node	18h	v1.19.9

Horizontal Pod Autoscaler Testing

```
root@bastion: ~ 63x25
```

```
</head>
<body>
  <noscript>You need to enable JavaScript to run this app.</noscript>
  <div id="root"></div>
  <!--
    This HTML file is a template.
    If you open it directly in the browser, you will see an empty page.

    You can add webfonts, meta tags, or analytics to this file.
    The build step will place the bundled scripts into the <body> tag.

    To begin the development, run `npm start` or `yarn start`
    To create a production bundle, use `npm run build` or `yarn build`.
  -->
  <script src="/static/js/bundle.js"></script><script src="/static/js/vendors~main.chunk.js"></script><script src="/static/js/main.chunk.js"></script></body>
</html>
```

```
root@bastion: ~/BigProject/Manifest/Production/Frontend 119x10
```

```
Every 1.0s: kubectl get hpa -A
```

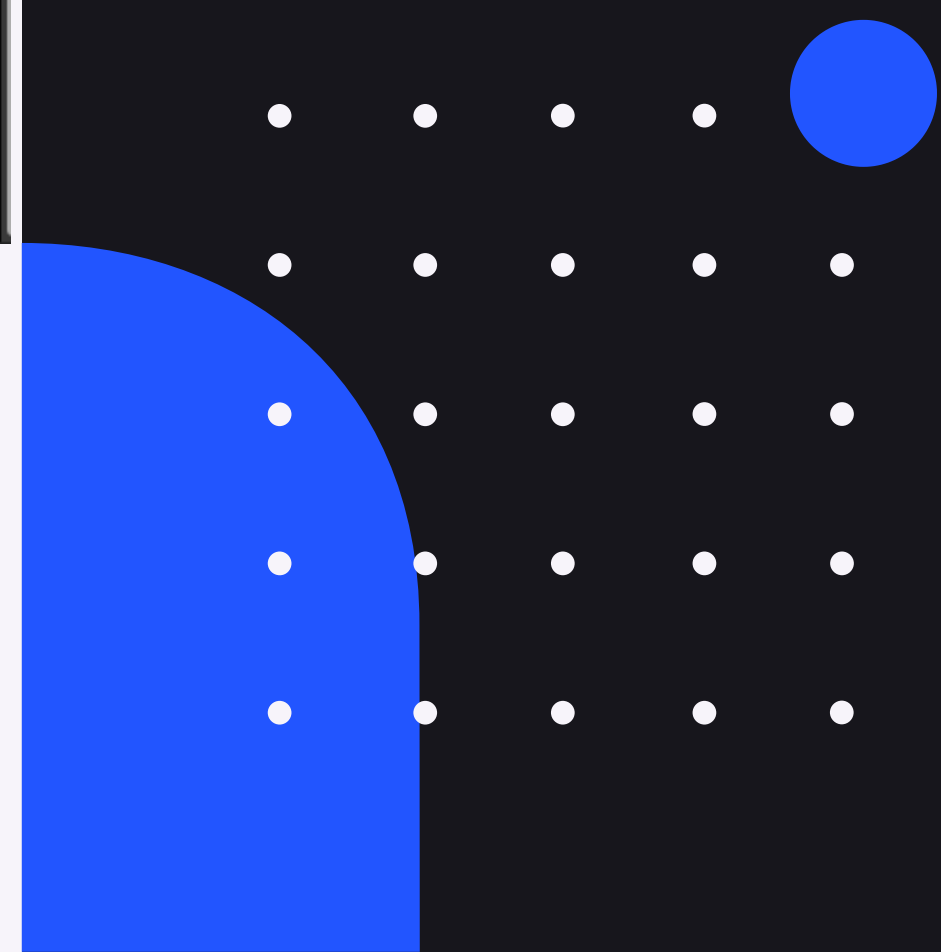
NAMESPACE	NAME	REFERENCE	TARGETS	MINPODS	MAXPODS	REPLICAS	AGE
production	backend-prod-hpa	Deployment/backend-prod	11%/90%, 0%/90%	2	10	2	95m
production	frontend-prod-hpa	Deployment/frontend-prod	74%/90%, 144%/90%	2	10	4	92m

```
root@bastion: ~ 104x15
```

```
Every 1.0s: kubectl get pods -n production
```

NAME	READY	STATUS	RESTARTS	AGE
backend-prod-764c85cc7c-5b9bt	1/1	Running	0	95m
backend-prod-764c85cc7c-tgl7l	1/1	Running	0	95m
frontend-prod-564bffc44-89mxw	1/1	Running	0	37s
frontend-prod-564bffc44-cgddd	1/1	Running	0	7m32s
frontend-prod-564bffc44-rfpqg	1/1	Running	0	37s
frontend-prod-564bffc44-xthqg	1/1	Running	0	82s
mongo-prod-67d467b6-dk2cl	1/1	Running	0	10h

```
root@bastion:~# kubectl run -i --tty load-generator --image=busybox /bin/sh
If you don't see a command prompt, try pressing enter.
/ # while true; do wget -q -O- http://frontend-prod-service.production.svc.cluster.local; done
```



Closing

Kelebihan

- Infrastruktur yang dibangun sudah menggunakan Kubernetes sebagai Container Orchestration
- Sistem telah High Availability karena menerapkan Cluster Autoscaler dan HPA
- Sistem logging dan monitoring yang tersentral
- Telah menerapkan CI/CD pipeline

Kekurangan

- Belum menggunakan Infrastructure as Code seperti ansible dan terraform dalam provisioning
- Koneksi antar backend dan database masih menggunakan LB (ter-ekspose public)



Thank You!



FAISAL

find me on wa/telegram/signal: +628158337989

faisal.alfatih90@gmail.com

