DevOps Project Lab

Andrés Vallejo
Final Demo

http://computers-dpl-ecs-cluster-alb-1889931541.us-east-1.elb.amazonaws.com/computers/

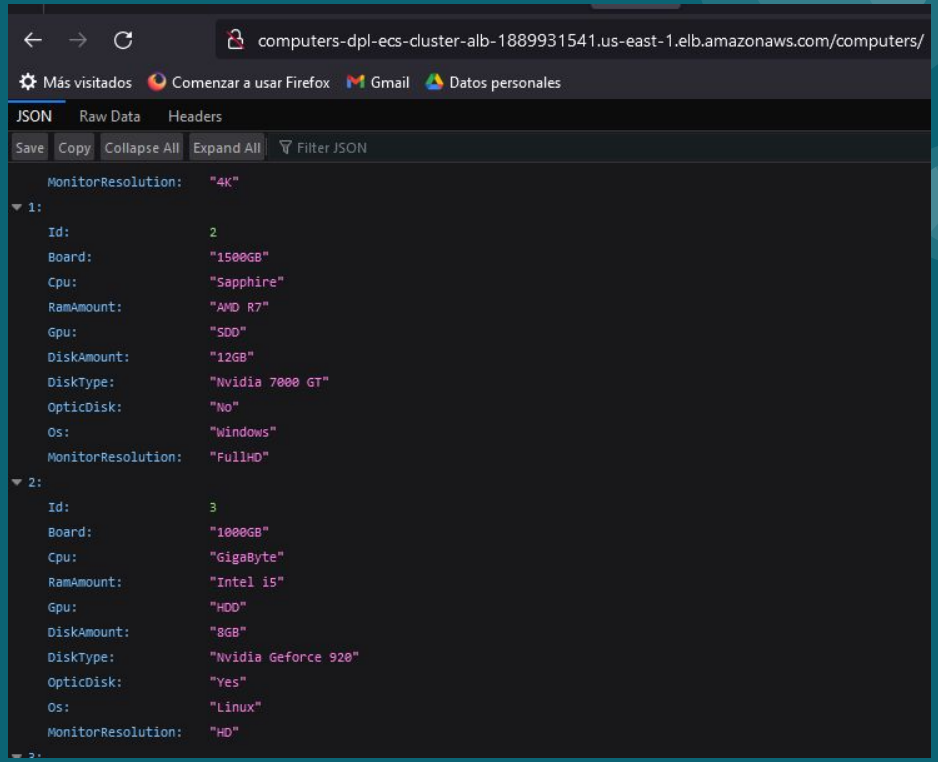
http://computers-dpl-ecs-cluster-alb-1889931541.us-east-1.elb.amazonaws.com/computers/2

```
curl --request POST
http://computers-dpl-ecs-cluster-alb-1889931541.us-east-1.elb.amazonaws.com/computers/ --data-raw '{"Board": "UnaBoardAsus", "Cpu": "Intel i7", "RamAmount": "16GB", "Gpu": "Nvidia GTX", "DiskAmount": "1000GB", "DiskType": "SSD", "OpticDisk": "Yes", "Os": "Windows", "MonitorResolution": "4K"}'
```

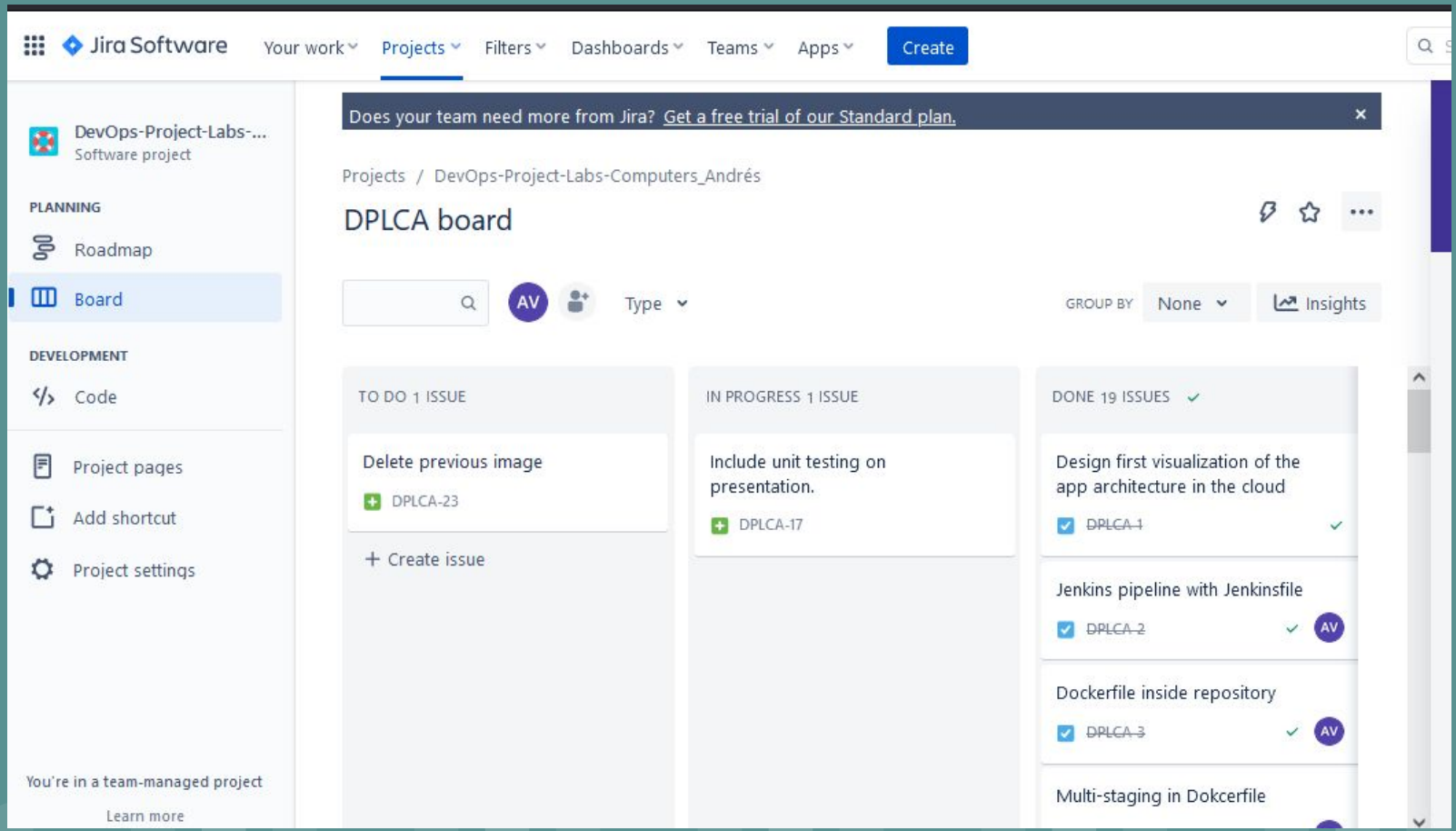
https://github.com/VallDev/Computers

A little view to my API

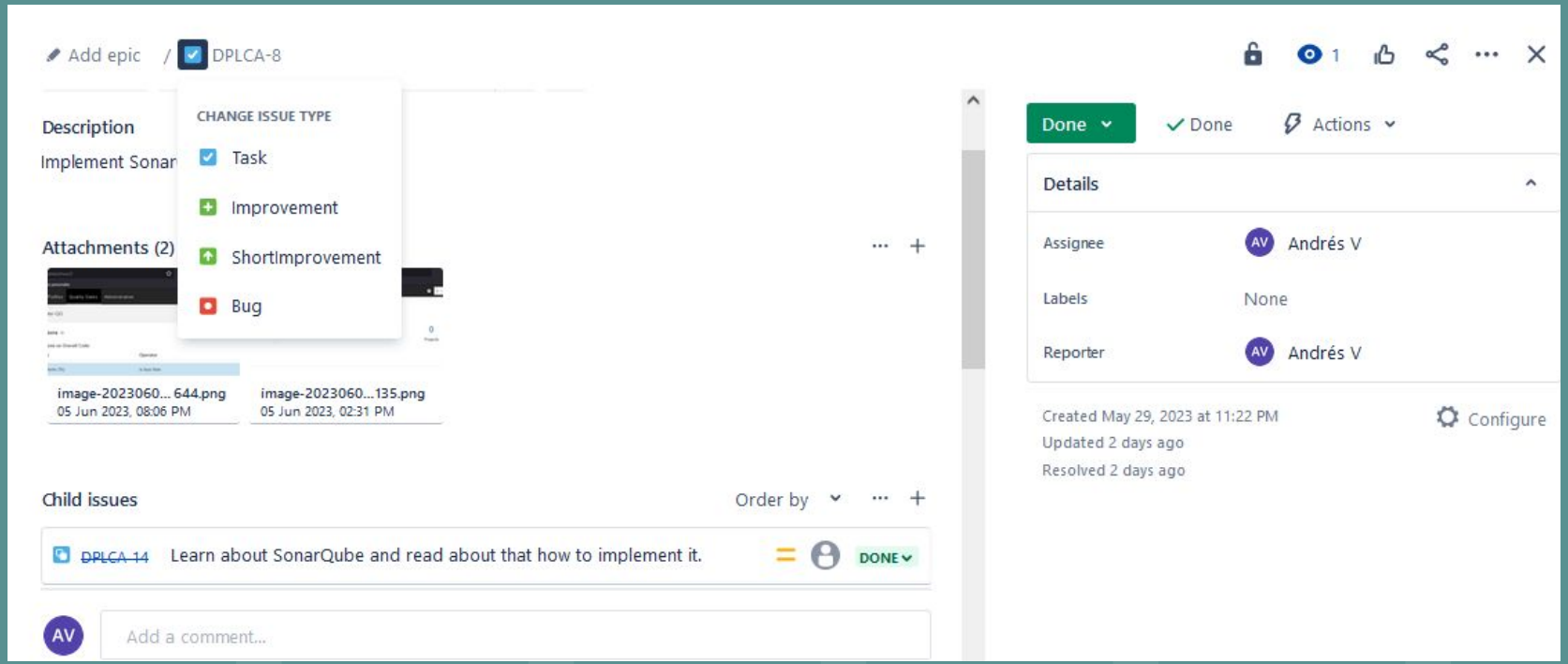
```
type Computer struct {
        Id                int
        Board             string
        Cpu               string
        RamAmount         string
        Gpu               string
        DiskAmount        string
        DiskType          string
        OpticDisk         string
        Os                string
        MonitorResolution string
}
```



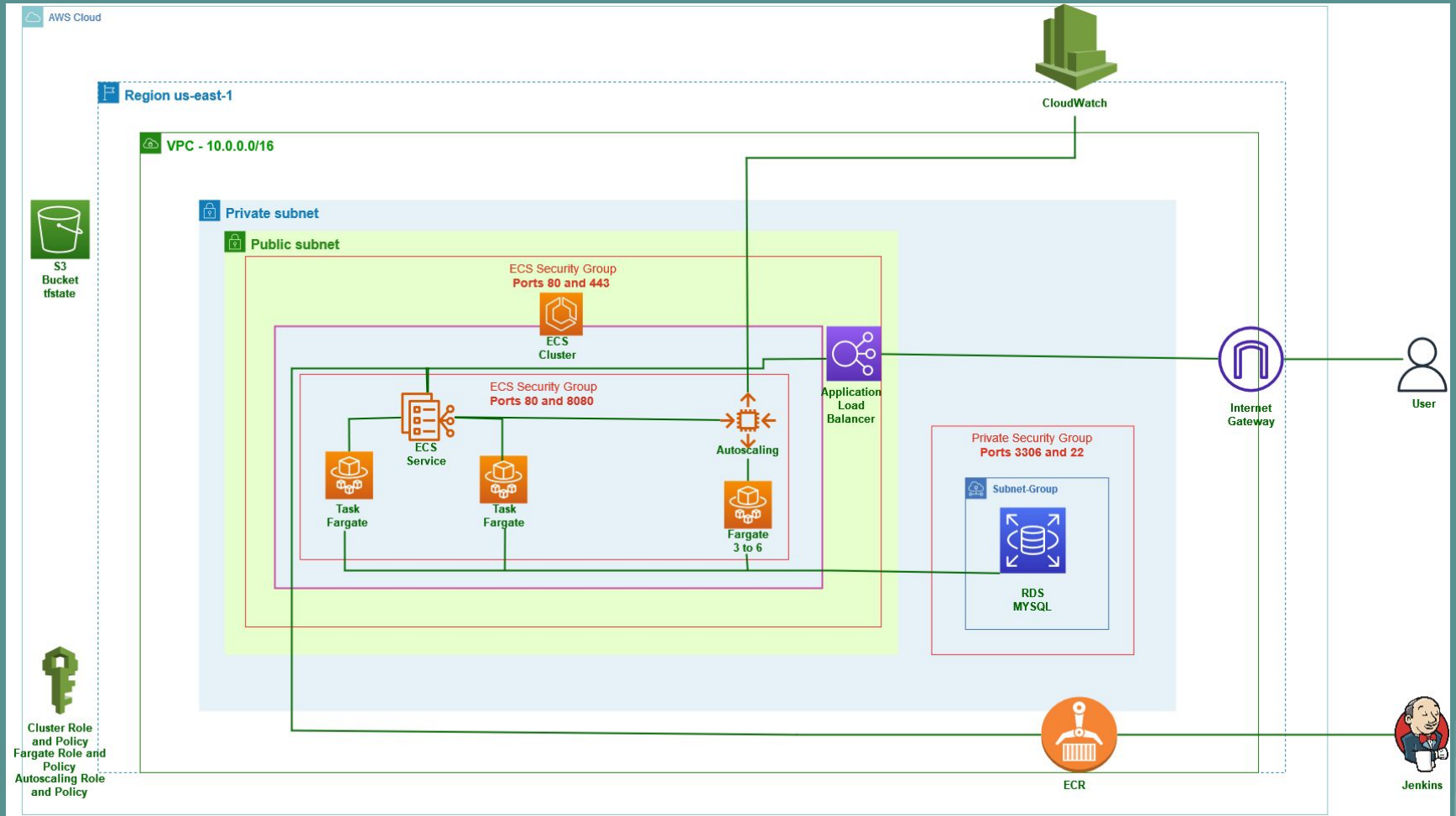
Plan


Jira Software
Your work
Projects
Filters
Dashboards
Teams
Apps
Create
DevOps-Project-Labs-...
Software project
PLANNING
Roadmap
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / DevOps-Project-Labs-Computers_Andrés
DPLCA board
AV
Type
GROUP BY
None
Insights
TO DO 1 ISSUE
Delete previous image
DPLCA-23
Create issue
IN PROGRESS 1 ISSUE
Include unit testing on presentation.
DPLCA-17
DONE 19 ISSUES
Design first visualization of the app architecture in the cloud
DPLCA-1
Jenkins pipeline with Jenkinsfile
DPLCA-2
Dockerfile inside repository
DPLCA-3
Multi-staging in Dokcerfile

Jira

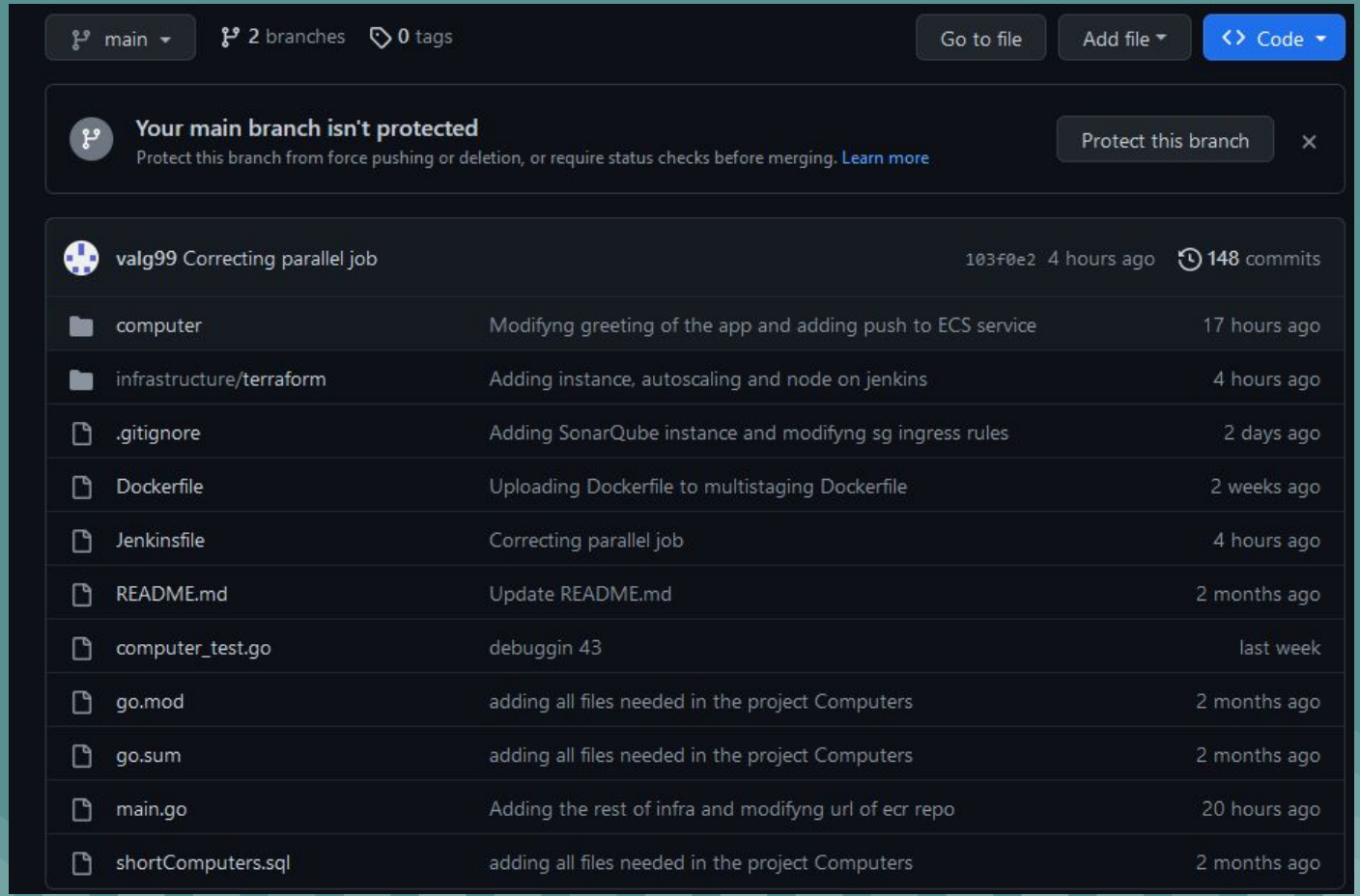

Add epic
DPLCA-8
1
Description
Implement Sonar
CHANGE ISSUE TYPE
Task
Improvement
ShortImprovement
Bug
Attachments (2)
image-2023060... 644.png
05 Jun 2023, 08:06 PM
image-2023060...135.png
05 Jun 2023, 02:31 PM
Child issues
Order by
DPLCA-14
Learn about SonarQube and read about that how to implement it.
DONE
AV
Add a comment...
Done
Done
Actions
Details
Assignee
Andrés V
Labels
None
Reporter
Andrés V
Created May 29, 2023 at 11:22 PM
Updated 2 days ago
Resolved 2 days ago
Configure

Architecture of the App



Code and build

Little view of my Github



```
3   import (
4           "context"
5           "database/sql"
6           "encoding/json"
7           "fmt"
8           "log"
9           "net/http"
10  )
11
12  func index(w http.ResponseWriter, r *http.Request) {
13          if r.Method != http.MethodGet {
14                  w.WriteHeader(http.StatusMethodNotAllowed)
15                  fmt.Fprintf(w, "Method not allowed")
16                  return
17          }
18
19          fmt.Fprintf(w, "Hello there %s", "visitor --> This was a project for DevOp
20  }
21
```

Test

```go
func TestNewEmptyComputer(t *testing.T) {

	comp := computer.NewEmptyComputer()

	if comp.Board != "" || comp.Cpu != "" || comp.DiskAmount != "" || comp.DiskType != "" || comp
		comp.Id != 0 || comp.MonitorResolution != "" || comp.OpticDisk != "" || comp.Os != ""

		t.Errorf("Expected new computer empty, but got: %v", comp)
	}
}

func TestSliceOfStrings(t *testing.T) {

	comp := computer.NewEmptyComputer()
	compSlice := comp.SliceOfStrings()

	if len(compSlice) != 9 {
		t.Errorf("Expected the slice of computers with length: 9, but got: %v", len(compSlice
	}

	if compSlice[2] != "RAM: " {
		t.Errorf("Expected RAM in third position of slice, but got: %v", compSlice[2])
	}
```

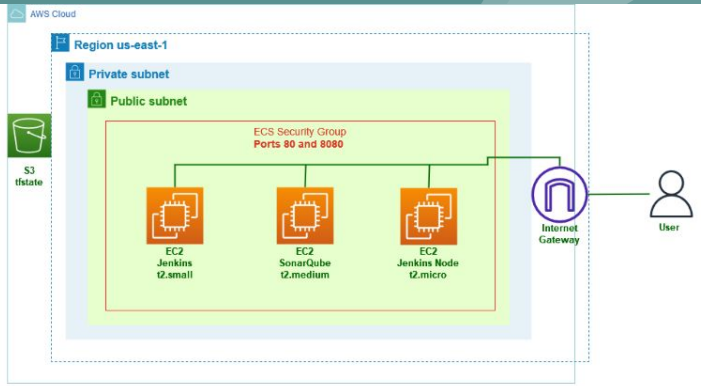
Build, integration and deploy

Dockerfile

```
1   FROM golang:1.20-alpine AS BUILDER
2
3   #RUN go get github.com/gorilla/mux && go get github.com/go-sql-driver/mysql
4   #RUN go mod download github.com/gorilla/mux &&  go mod download github.com/gorilla/mux
5
6   RUN mkdir /build
7   RUN echo "...aqui...." && pwd
8   COPY . /build
9
10  WORKDIR /build
11
```

```
RUN export GOPROXY=https://goproxy.cn
RUN go mod download && go mod verify

RUN GOOS=linux && GOARCH=amd64 && go build -v main.go
RUN echo "archivos despues de build main" && ls -la
RUN chmod +x main.go


#EXPOSE 8080

#ENTRYPOINT [ "./main" ]

FROM alpine:latest
WORKDIR /root/
COPY --from=BUILDER /build/main .
RUN echo "---aqui estoy en alpine-----" && pwd && ls -la
RUN chmod +x main
RUN echo "-------------observando donde esta e. ./main--------" && pwd && ls -la
EXPOSE 8080
CMD [ "./main" ]
```

Jenkins architecture and pipeline

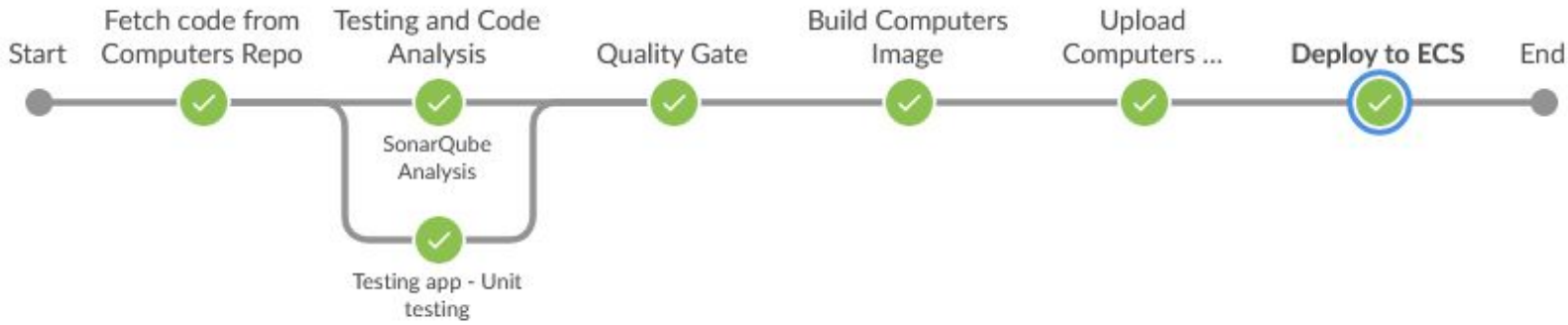


Jenkinsfile

```
environment {
    CURRENT_STAGE = ""
    STAGE_RESULT = 0

    REGISTRY_CREDENTIAL = "ecr:us-east-1:awscreds"
    COMPUTER_REGISTRY = "855149291285.dkr.ecr.us-east-1.amazonaws.com/computers-dpl-ecr-repo-img-andres"
    LINK_REGISTRY = "https://855149291285.dkr.ecr.us-east-1.amazonaws.com"

    CLUSTER = "COMPUTERS-DPL-FARGATE-CLUSTER"
    SERVICE = "computersapp"
}
```

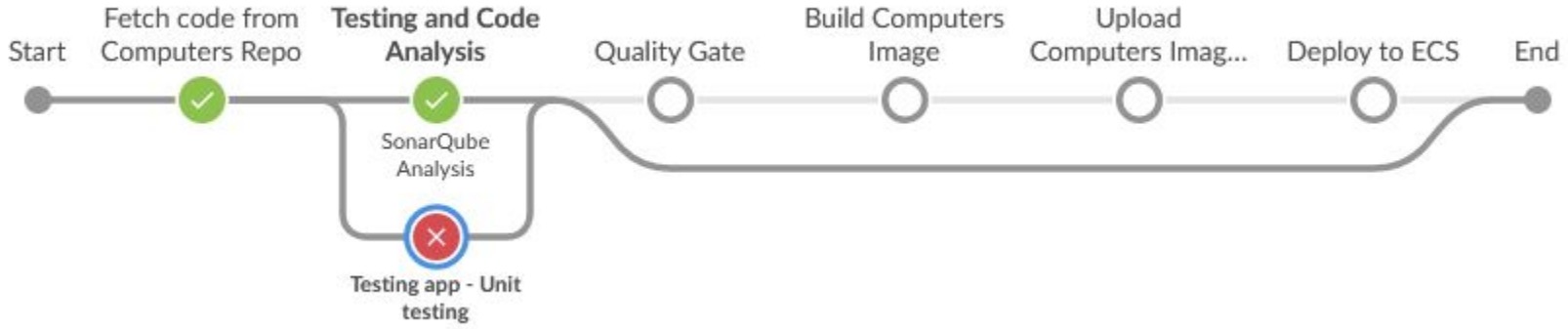
```
stage('Fetch code from Computers Repo') {
    steps {
        script {
            CURRENT_STAGE = env.STAGE_NAME
        }
        echo '---------------STARTING PIPELINE---------------------'
        echo '---------------FETCHING CODE FROM DEV BRANCH---------'
        git branch: 'dev', url: 'https://github.com/VallDev/Computers.git'
    }
}
```

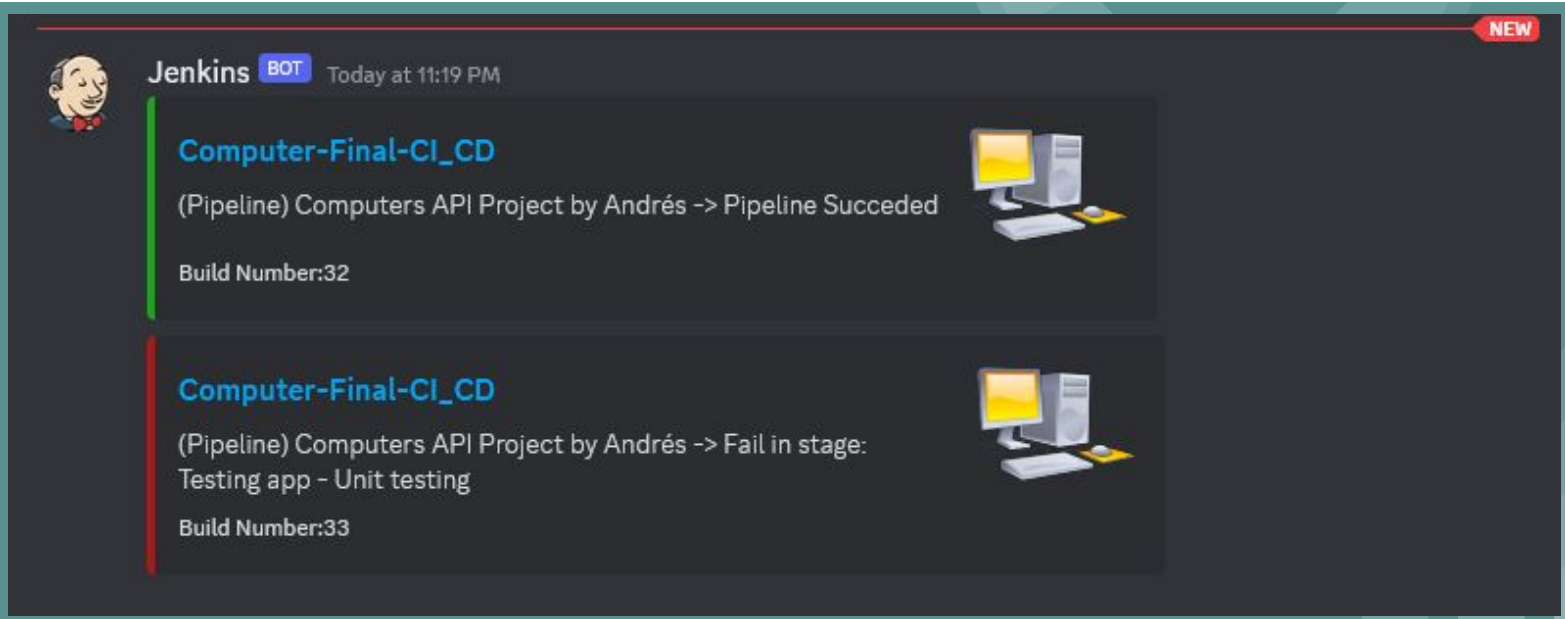
```
stage('Testing and Code Analysis'){
    parallel{
        stage('Testing app - Unit testing') {
            agent {
                node {
                    label 'NODE-MICRO'
                    customWorkspace '/home/ubuntu/node'
                }
            }
            steps{
                script {
                    CURRENT_STAGE = env.STAGE_NAME
                }
                script{
                    echo '---------------TESTING GOLANG COMPUTERS APP-----------------'
                    def testResult = sh(returnStatus: true, script: "go test")
                    env.STAGE_RESULT = testResult
                    if (testResult == 0) {
                        echo "---------SUCCESS TESTING GOLANG COMPUTERS APP-----------------"
                        STAGE_RESULT = 0
                    } else {
                        error "---------FAILED TESTING GOLANG COMPUTERS APP-----------------"
                        STAGE_RESULT = 1
                    }
                }
            }
```

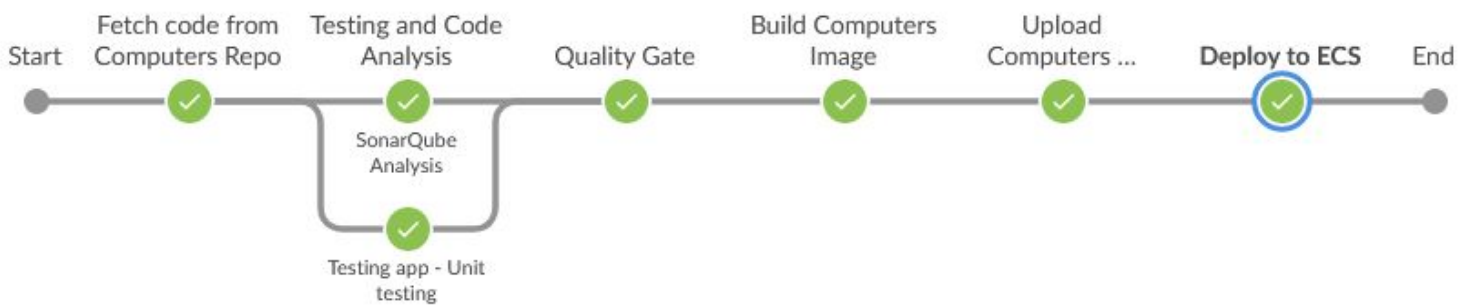



```
    //7
    if compSlice[7] != "Operative System: " {
        t.Errorf("Expected Operative System  in 8 position of slice, but got: %v", compSlice[7])
    }
```

```
//7
if compSlice[5] != "Operative System: " {
    t.Errorf("Expected Operative System  in 8 position of slice, but got: %v", compSlice[5])
}
```


Start
Fetch code from Computers Repo
Testing and Code Analysis
SonarQube Analysis
Testing app - Unit testing
Quality Gate
Build Computers Image
Upload Computers Imag...
Deploy to ECS
End


NEW
Jenkins BOT Today at 11:19 PM
Computer-Final-CI_CD
(Pipeline) Computers API Project by Andrés -> Pipeline Succeded
Build Number:32
Computer-Final-CI_CD
(Pipeline) Computers API Project by Andrés -> Fail in stage: Testing app - Unit testing
Build Number:33


Start
Fetch code from Computers Repo
Testing and Code Analysis
SonarQube Analysis
Testing app - Unit testing
Quality Gate
Build Computers Image
Upload Computers ...
Deploy to ECS
End

Quality Gates

Create

Sonar way DEFAULT BUILT-IN

computer-QG

computer-QG

Rename Copy Set as Default Delete

Conditions

Add Condition

Conditions on Overall Code

Metric	Operator	Value	Edit	Delete
Comments (%)	is less than	5.0%		

Projects

With Without All

Search

Uncovered Lines 135

Line Coverage 0.0%

Duplications

Size COMMENTS (%)

New Lines 6

Lines of Code 259

Lines 356

Statements 135

Functions 14

Classes 1

Files 5

Comment Lines 14

Comments (%) 5.1%

Comments (%) **5.1%**

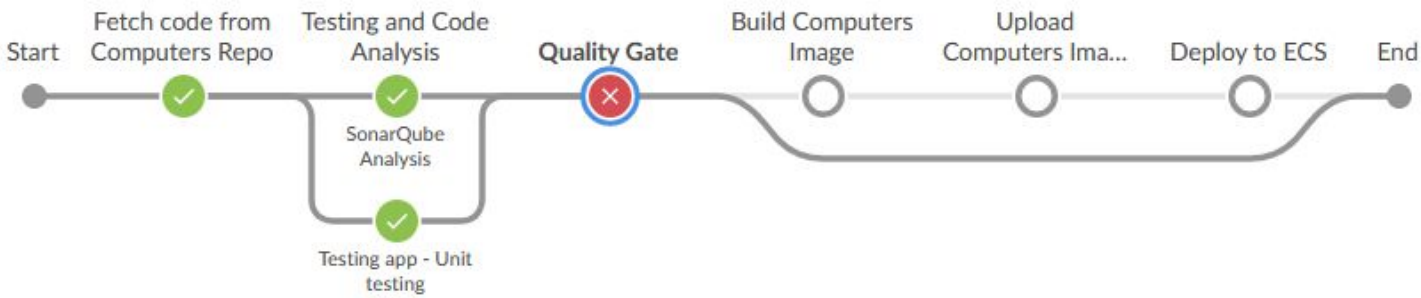
computer

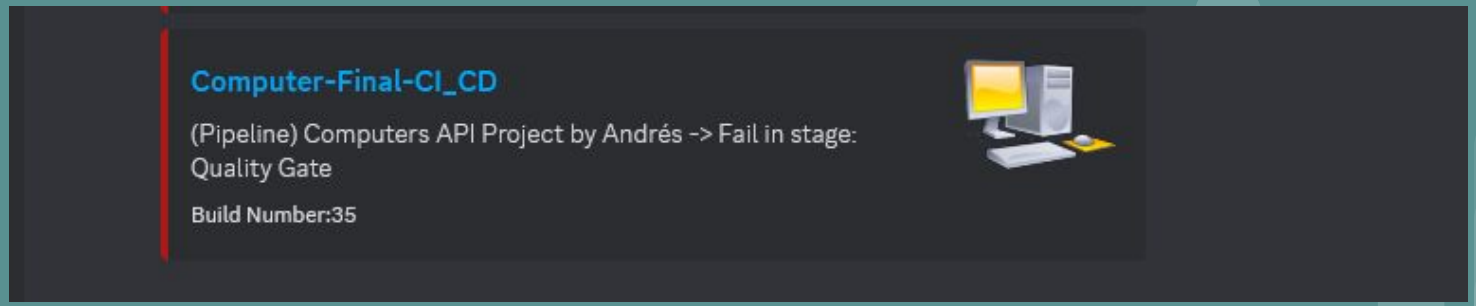
main.go

computer_test.go

Conditions on Overall Code

Metric	Operator	Value	Edit	Delete
Comments (%)	is less than	10.0%		





```
stage('Build Computers Image'){
    steps{
        script {
            CURRENT_STAGE = env.STAGE_NAME
        }

        echo "-------------BUILDING DOCKER IMAGE----------------------"

        script{
            dockerImage = docker.build( COMPUTER_REGISTRY + ":${BUILD_NUMBER}", ".")
        }
    }
}
```

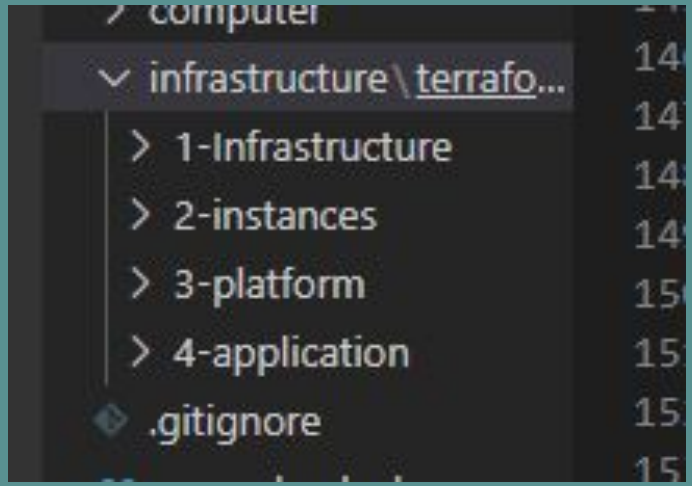
```
stage('Upload Computers Image to ECR') {
    steps{
        script {
            CURRENT_STAGE = env.STAGE_NAME
        }

        echo "-----------PUSHING DOCKER IMAGE TO ECR SERVICE OF AWS-----------------"

        script {
            docker.withRegistry( LINK_REGISTRY, REGISTRY_CREDENTIAL) {
                dockerImage.push("${BUILD_NUMBER}")
                dockerImage.push('latest')
            }
        }
    }
}
```

```
    stage('Deploy to ECS') {
        steps {
            script {
                CURRENT_STAGE = env.STAGE_NAME
            }

            echo "---------------DEPLOYING APPLICATION ON ECS SERVICE OF AWS-----------------"

            withAWS(credentials: 'awscreds', region: 'us-east-1') {
                sh 'aws ecs update-service --cluster ${CLUSTER} --service ${SERVICE} --force-new-deployment'
            }
        }
    }
}
```

```
    post{
        success {
            echo '-------------SENDING MESSAGE OF SUCCESS TO DISCORD CHANNEL ANDRES'
            discordSend description: "(Pipeline) Computers API Project by Andrés -> Pipeline Succeded", footer: "Build Numb
        }

        failure {
            echo '-------------SENDING MESSAGE TO DISCORD CHANNEL ANDRES'
            discordSend description: "(Pipeline) Computers API Project by Andrés -> Fail in stage: ${CURRENT_STAGE}", foote
        }

    }
}
```

Infrastructure building and deploying

Terraform

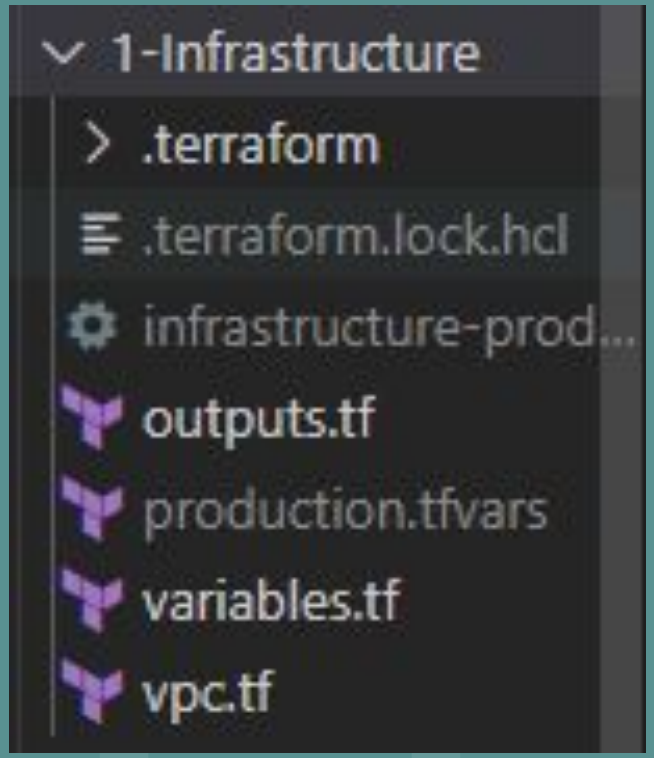


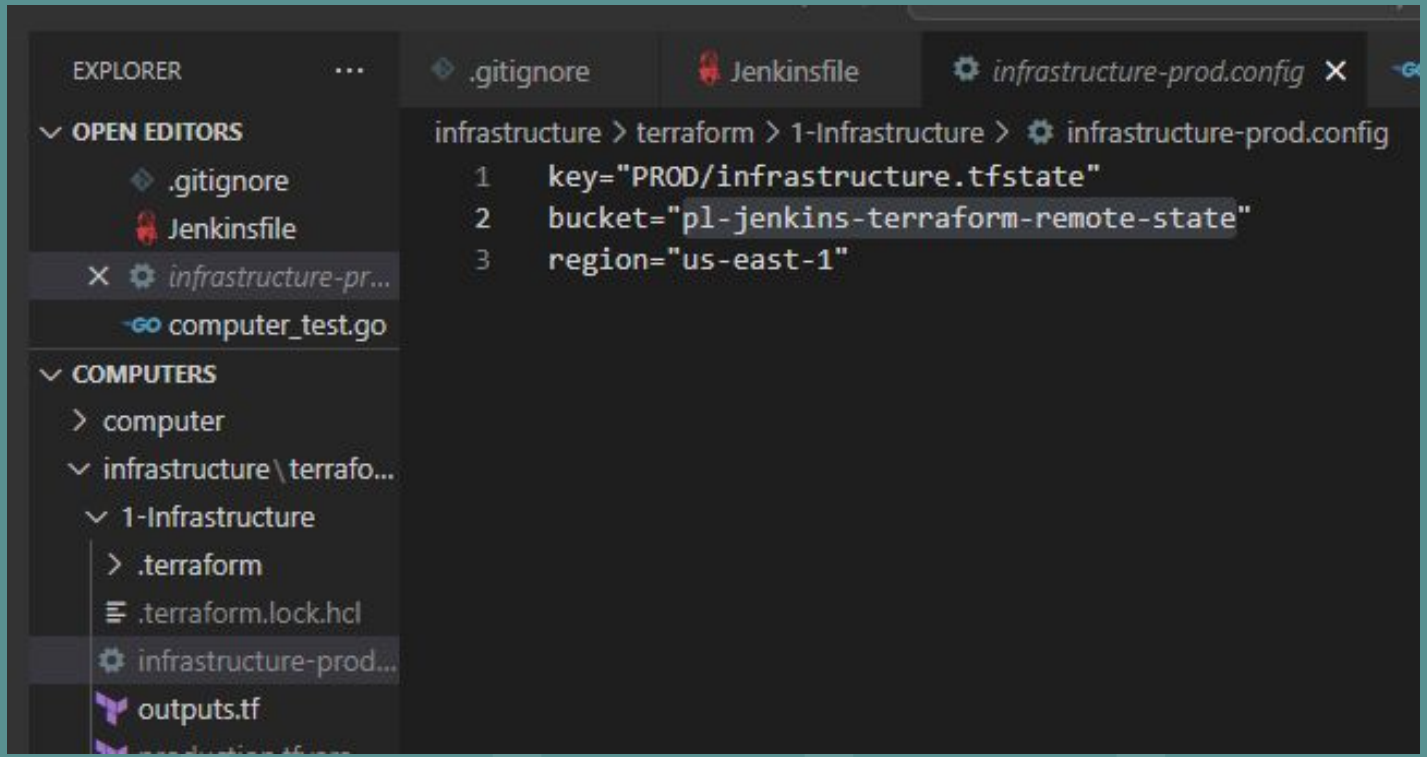
Create folder | Upload

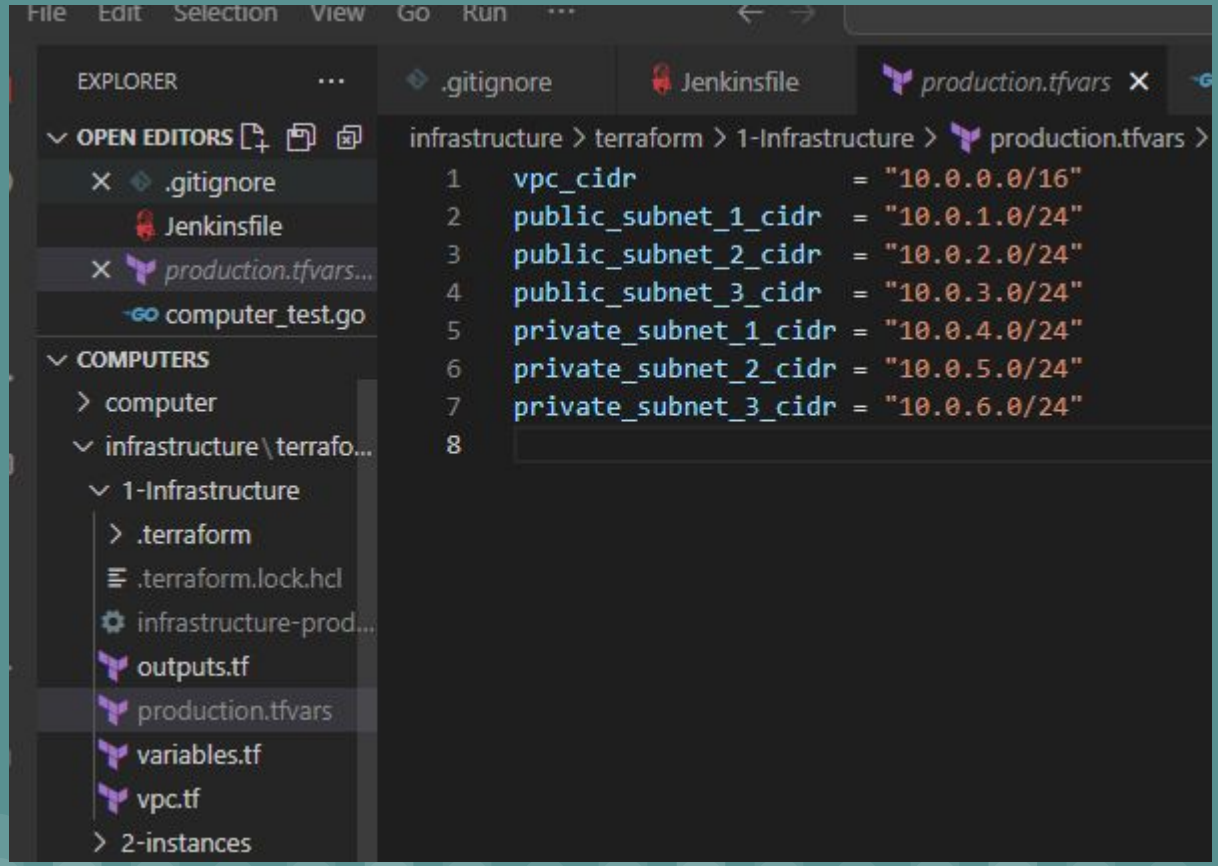
Find objects by prefix

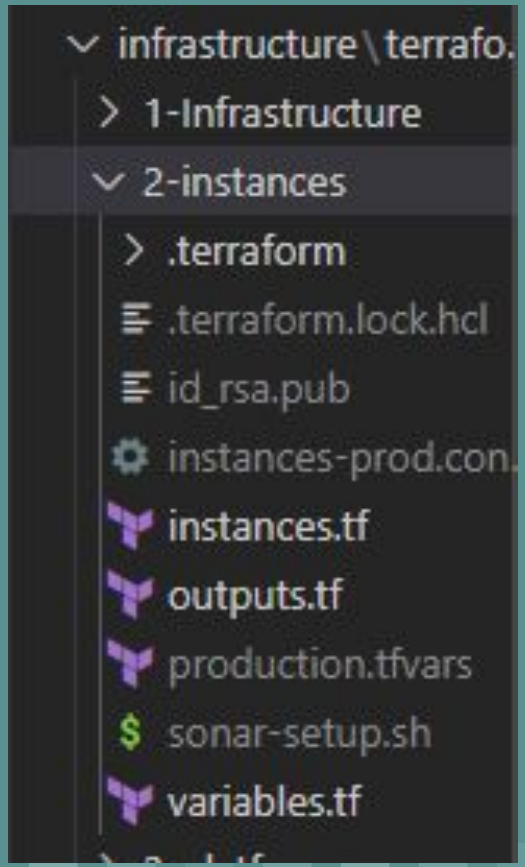
1

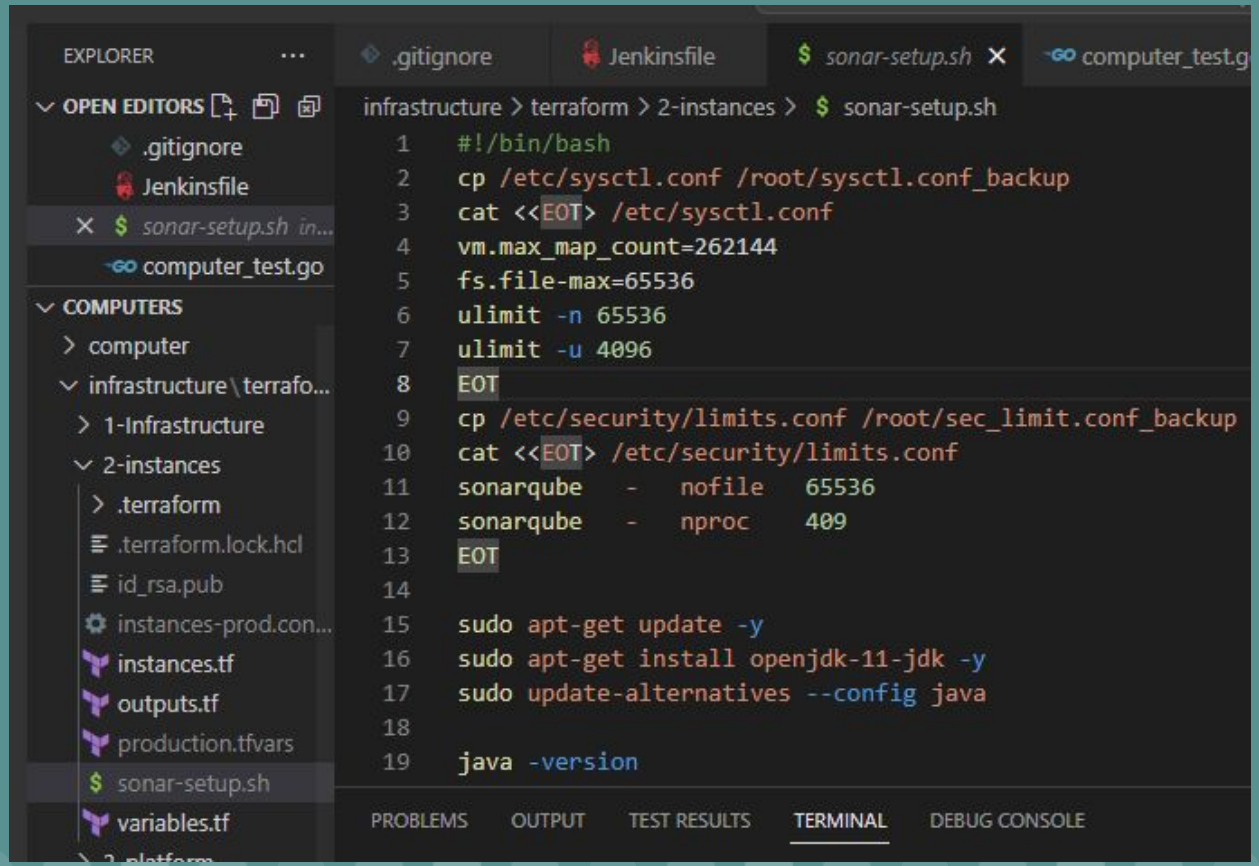
Name	Type	Last modified	Size	Storage class
application.tfstate	tfstate	June 7, 2023, 23:17:48 (UTC-05:00)	27.8 KB	Standard
infrastructure.tfstate	tfstate	June 6, 2023, 16:13:52 (UTC-05:00)	23.3 KB	Standard
instances.tfstate	tfstate	June 7, 2023, 16:19:24 (UTC-05:00)	27.6 KB	Standard
platform.tfstate	tfstate	June 6, 2023, 21:28:49 (UTC-05:00)	16.6 KB	Standard

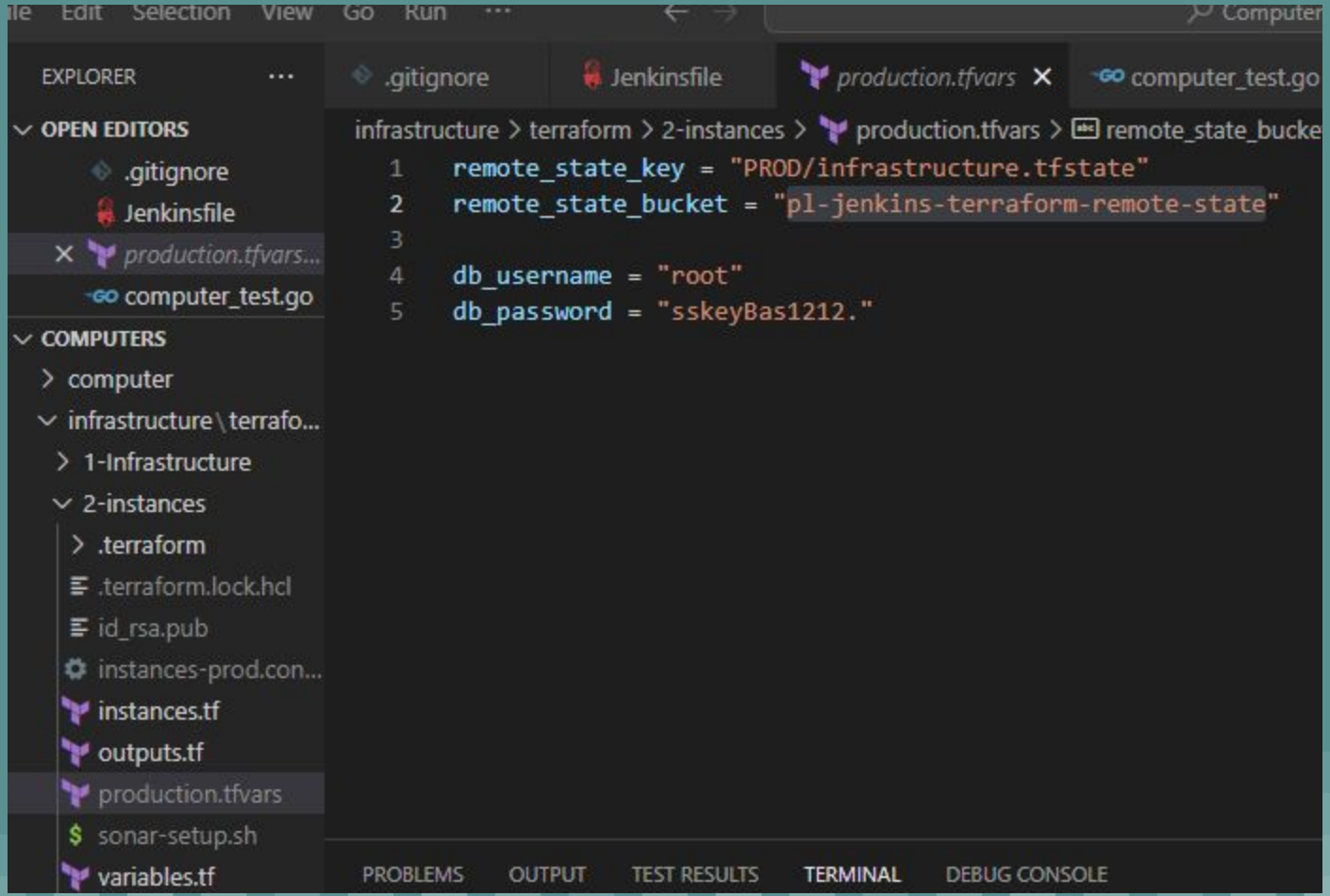

1-Infrastructure
.terraform
.terraform.lock.hcl
infrastructure-prod...
outputs.tf
production.tfvars
variables.tf
vpc.tf

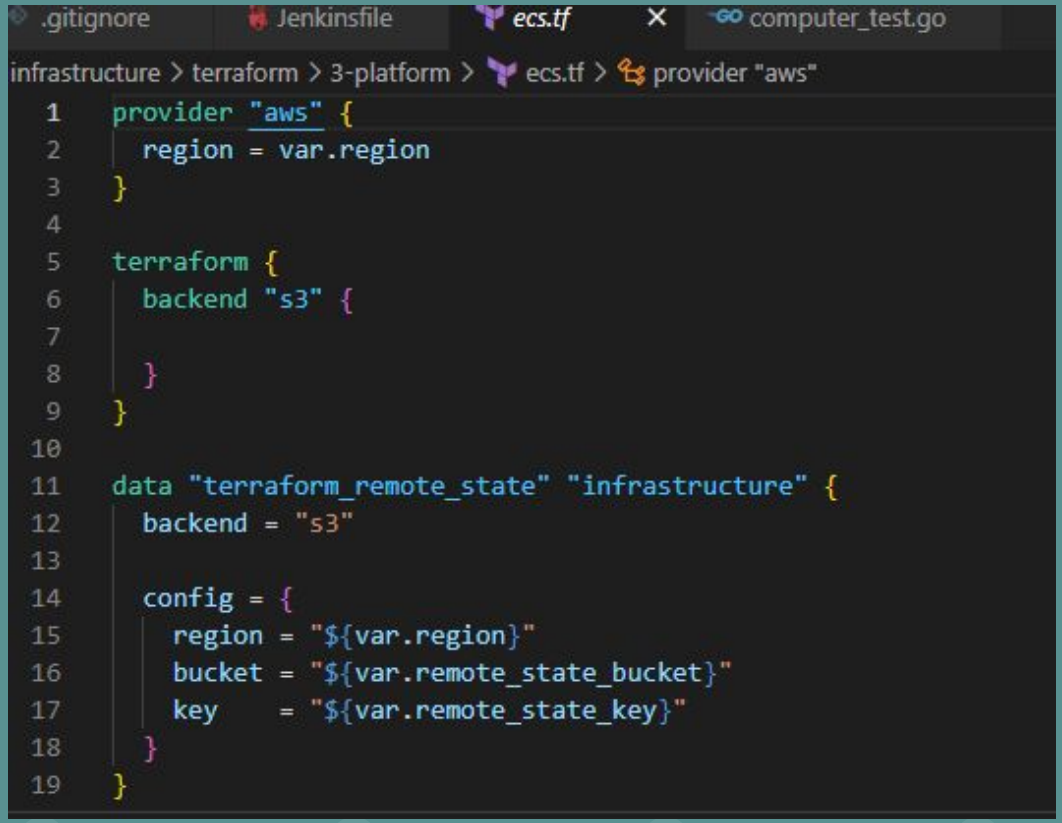

EXPLORER
.gitignore
Jenkinsfile
infrastructure-prod.config
OPEN EDITORS
.gitignore
Jenkinsfile
infrastructure-pr...
computer_test.go
COMPUTERS
computer
infrastructure \ terrafo...
1-Infrastructure
.terraform
.terraform.lock.hcl
infrastructure-prod...
outputs.tf
infrastructure > terraform > 1-Infrastructure > infrastructure-prod.config
1 key="PROD/infrastructure.tfstate"
2 bucket="pl-jenkins-terraform-remote-state"
3 region="us-east-1"


File Edit Selection View Go Run ···
EXPLORER ···
.gitignore
Jenkinsfile
production.tfvars
OPEN EDITORS
.gitignore
Jenkinsfile
production.tfvars...
computer_test.go
COMPUTERS
computer
infrastructure \ terrafo...
1-Infrastructure
.terraform
.terraform.lock.hcl
infrastructure-prod...
outputs.tf
production.tfvars
variables.tf
vpc.tf
2-instances
infrastructure > terraform > 1-Infrastructure > production.tfvars >
1 vpc_cidr = "10.0.0.0/16"
2 public_subnet_1_cidr = "10.0.1.0/24"
3 public_subnet_2_cidr = "10.0.2.0/24"
4 public_subnet_3_cidr = "10.0.3.0/24"
5 private_subnet_1_cidr = "10.0.4.0/24"
6 private_subnet_2_cidr = "10.0.5.0/24"
7 private_subnet_3_cidr = "10.0.6.0/24"
8

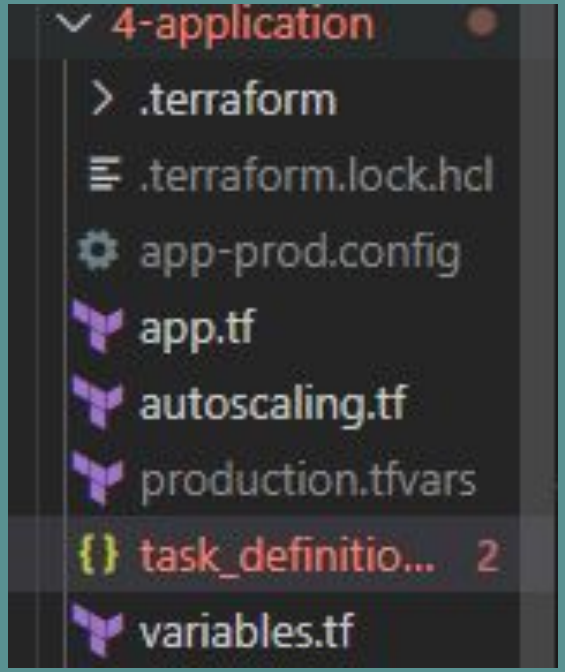

infrastructure \ terrafo.
1-Infrastructure
2-instances
.terraform
.terraform.lock.hcl
id_rsa.pub
instances-prod.con.
instances.tf
outputs.tf
production.tfvars
sonar-setup.sh
variables.tf

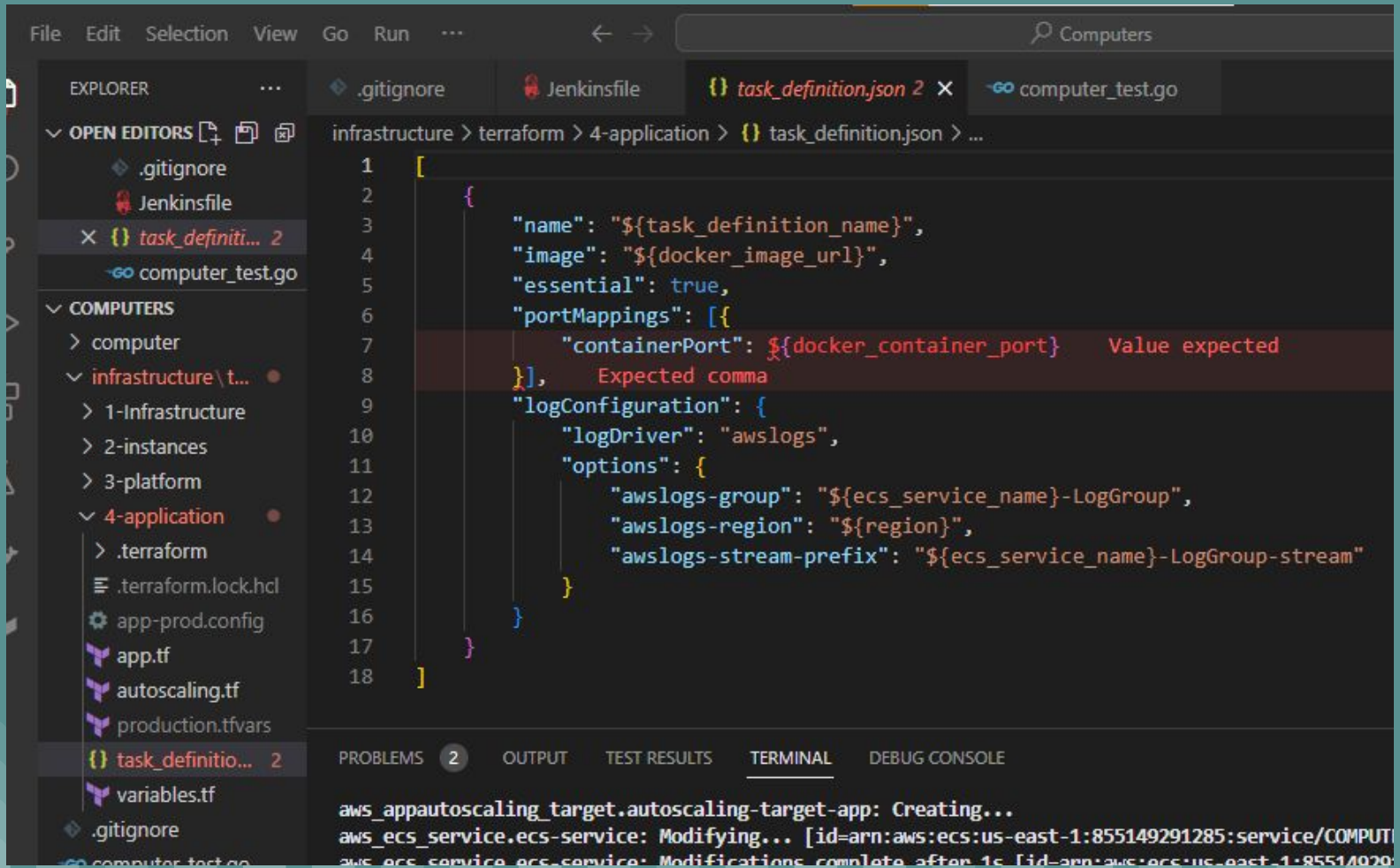

EXPLORER
OPEN EDITORS
.gitignore
Jenkinsfile
sonar-setup.sh in...
computer_test.go
COMPUTERS
computer
infrastructure \ terrafo...
1-Infrastructure
2-instances
.terraform
.terraform.lock.hcl
id_rsa.pub
instances-prod.con...
instances.tf
outputs.tf
production.tfvars
sonar-setup.sh
variables.tf
.gitignore
Jenkinsfile
sonar-setup.sh
computer_test.g
infrastructure > terraform > 2-instances > sonar-setup.sh
1 #!/bin/bash
2 cp /etc/sysctl.conf /root/sysctl.conf_backup
3 cat <<EOT> /etc/sysctl.conf
4 vm.max_map_count=262144
5 fs.file-max=65536
6 ulimit -n 65536
7 ulimit -u 4096
8 EOT
9 cp /etc/security/limits.conf /root/sec_limit.conf_backup
10 cat <<EOT> /etc/security/limits.conf
11 sonarqube - nofile 65536
12 sonarqube - nproc 409
13 EOT
14
15 sudo apt-get update -y
16 sudo apt-get install openjdk-11-jdk -y
17 sudo update-alternatives --config java
18
19 java -version
PROBLEMS
OUTPUT
TEST RESULTS
TERMINAL
DEBUG CONSOLE


File Edit Selection View Go Run ···
Computer
EXPLORER
.gitignore
Jenkinsfile
production.tfvars
computer_test.go
OPEN EDITORS
.gitignore
Jenkinsfile
production.tfvars...
computer_test.go
COMPUTERS
computer
infrastructure \ terrafo...
1-Infrastructure
2-instances
.terraform
.terraform.lock.hcl
id_rsa.pub
instances-prod.con...
instances.tf
outputs.tf
production.tfvars
sonar-setup.sh
variables.tf
infrastructure > terraform > 2-instances > production.tfvars > remote_state_bucke
1 remote_state_key = "PROD/infrastructure.tfstate"
2 remote_state_bucket = "pl-jenkins-terraform-remote-state"
3
4 db_username = "root"
5 db_password = "sskeyBas1212."
PROBLEMS
OUTPUT
TEST RESULTS
TERMINAL
DEBUG CONSOLE



Each layer has terraform_remote_state that is from previous layer


4-application
.terraform
.terraform.lock.hcl
app-prod.config
app.tf
autoscaling.tf
production.tfvars
task_definitio... 2
variables.tf


File Edit Selection View Go Run ···
Computers
EXPLORER
OPEN EDITORS
.gitignore
Jenkinsfile
task_definiti... 2
computer_test.go
COMPUTERS
computer
infrastructure \ t...
1-Infrastructure
2-instances
3-platform
4-application
.terraform
.terraform.lock.hcl
app-prod.config
app.tf
autoscaling.tf
production.tfvars
task_definitio... 2
variables.tf
.gitignore
.gitignore
Jenkinsfile
task_definition.json 2
computer_test.go
infrastructure > terraform > 4-application > task_definition.json > ...
1 [
2 {
3 "name": "${task_definition_name}",
4 "image": "${docker_image_url}",
5 "essential": true,
6 "portMappings": [{
7 "containerPort": ${docker_container_port}
8 }],
9 "logConfiguration": {
10 "logDriver": "awslogs",
11 "options": {
12 "awslogs-group": "${ecs_service_name}-LogGroup",
13 "awslogs-region": "${region}",
14 "awslogs-stream-prefix": "${ecs_service_name}-LogGroup-stream"
15 }
16 }
17 }
18]
Value expected
Expected comma
PROBLEMS 2
OUTPUT
TEST RESULTS
TERMINAL
DEBUG CONSOLE
aws_appautoscaling_target.autoscaling-target-app: Creating...
aws_ecs_service.ecs-service: Modifying... [id=arn:aws:ecs:us-east-1:855149291285:service/COMPUT

```
#service variables
ecs_service_name      = "computersapp"
docker_container_port = 8080
desired_task_number   = "2"
memory                = 1024

#task variables
```

```
resource "aws_appautoscaling_target" "autoscaling-target-app" {
  max_capacity       = 6
  min_capacity       = 3
  resource_id        = "service/${var.ecs_cluster_name_this_infra}/${var.ecs_service_name}"
  scalable_dimension = "ecs:service:DesiredCount"
  service_namespace  = "ecs"
  role_arn           = aws_iam_role.ecs-autoscale-role.arn

  depends_on = [aws_appautoscaling_target.autoscaling-target-app]
}
```

computersapp Info

Update service | Delete service

Health and metrics | Tasks | Logs | Deployments | Events | Configuration | Networking | Tags

Status Info

ARN
COMPUTERS-DPL-FARGATE-CLUSTER/computersapp

Status
Active

Tasks (3 Desired)
0 Pending | 3 Running

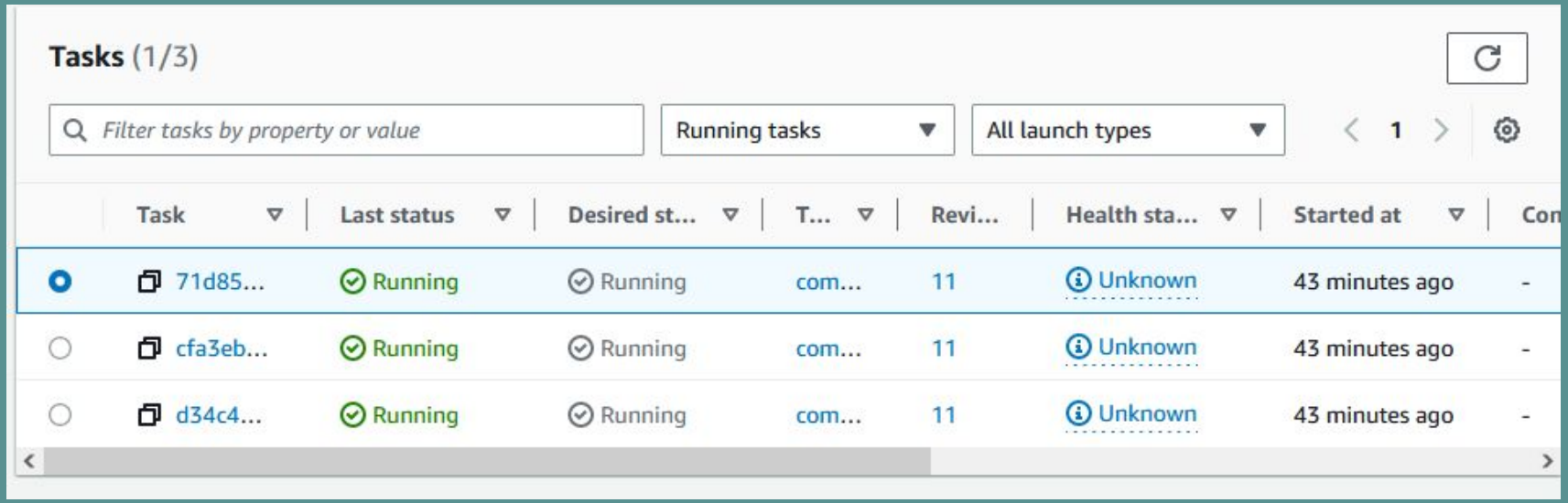
Deployments current state
3 Completed

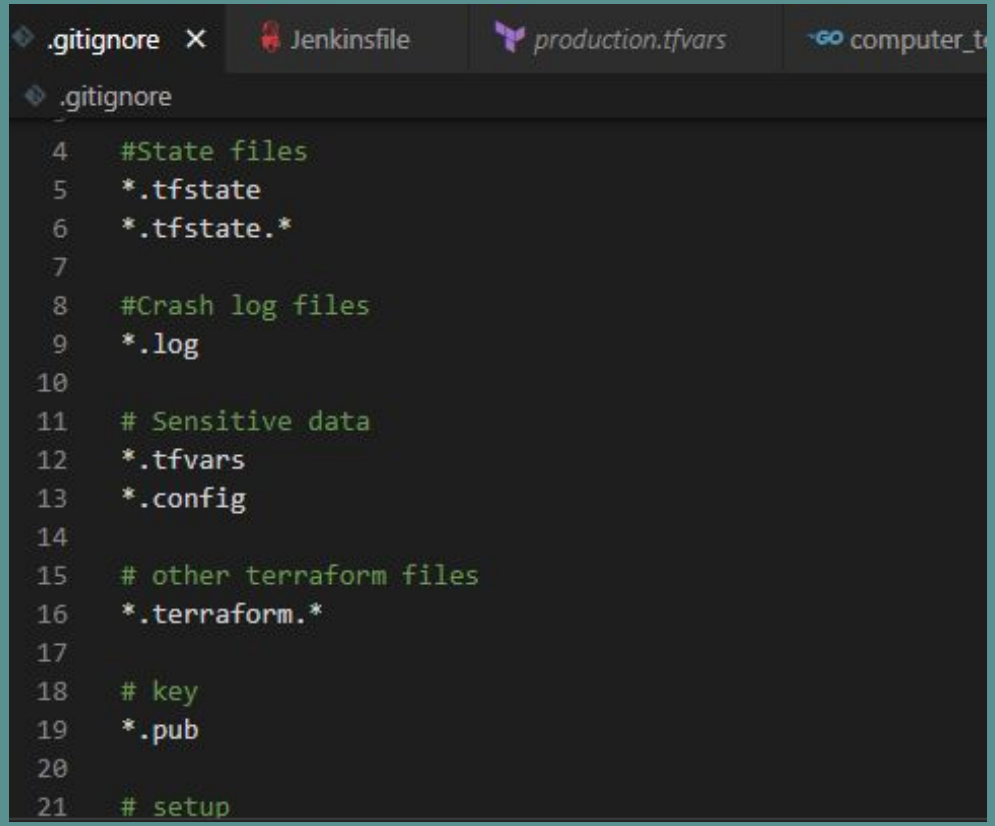
Health check grace period
0 seconds

▼ Load balancer health

(Application Load Balancer) COMPUTERS-DPL-ECS-CLUSTER-ALB

View load balancer


Tasks (1/3)
Filter tasks by property or value
Running tasks
All launch types
1
Task
Last status
Desired st...
T...
Revi...
Health sta...
Started at
Con
71d85...
Running
Running
com...
11
Unknown
43 minutes ago
-
cfa3eb...
Running
Running
com...
11
Unknown
43 minutes ago
-
d34c4...
Running
Running
com...
11
Unknown
43 minutes ago
-

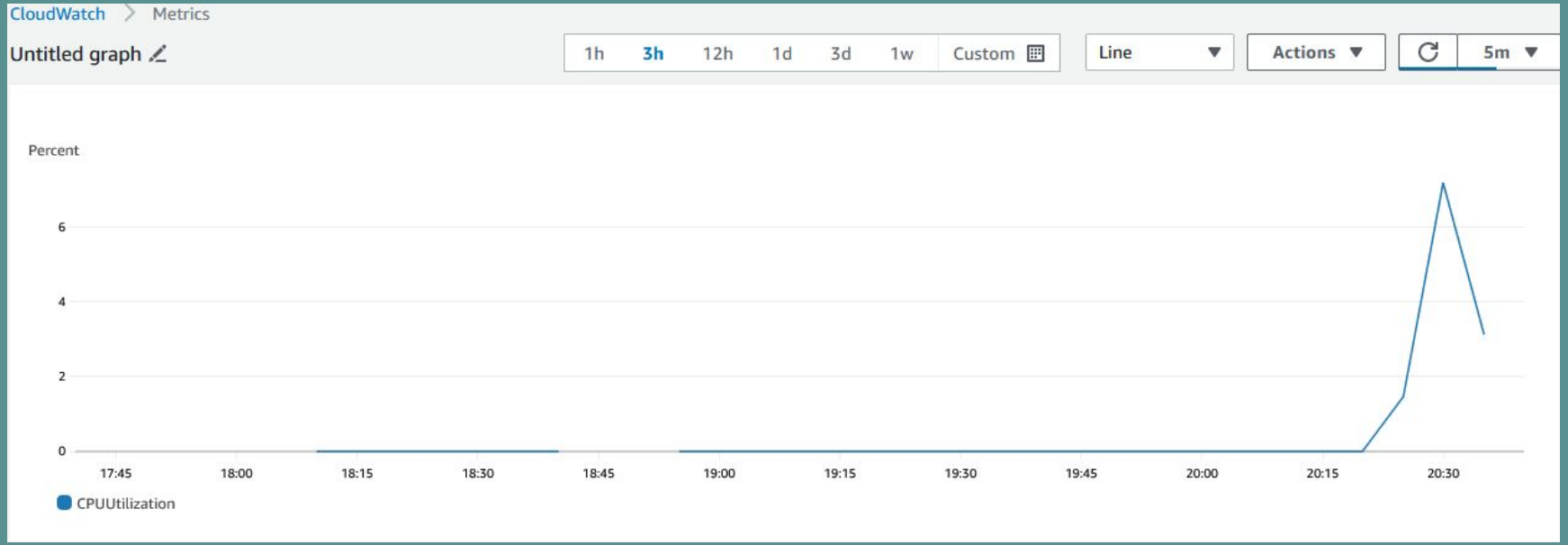


My .gitignore file

Autoscaling and monitoring

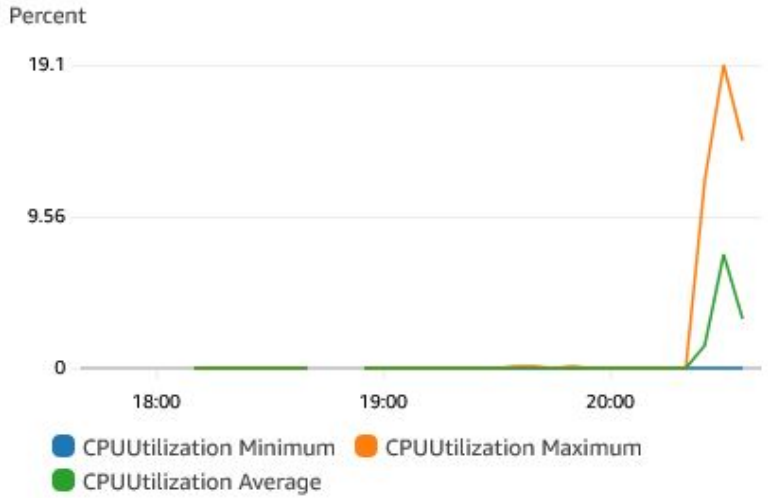
```
Benchmarking computers-dpl-ecs-cluster-alb-1889931541.us-east-1.elb.amazonaws.com (be patient)
Completed 100 requests
Completed 200 requests
Completed 300 requests
Completed 400 requests
Completed 500 requests
Completed 600 requests
Completed 700 requests
Completed 800 requests
Completed 900 requests
Completed 1000 requests
Finished 1000 requests


Server Software:
Server Hostname:        computers-dpl-ecs-cluster-alb-1889931541.us-east-1.elb.amazonaws.com
Server Port:            80

Document Path:          /computers/
Document Length:        737 bytes

Concurrency Level:      1000
Time taken for tests:   38.466 seconds
Complete requests:      1000
Failed requests:        836
   (Connect: 0, Receive: 0, Length: 836, Exceptions: 0)
Total transferred:      63156758 bytes
```

Apache Bench to test

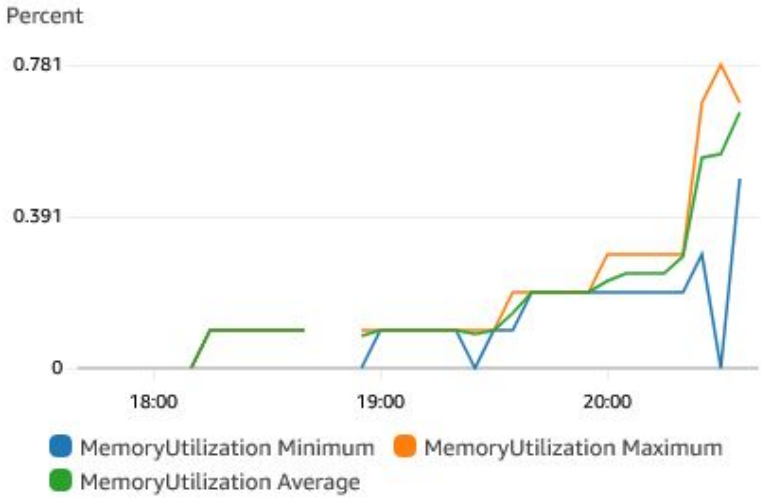

CloudWatch
Metrics
Untitled graph
1h
3h
12h
1d
3d
1w
Custom
Line
Actions
5m
Percent
6
4
2
0
17:45
18:00
18:15
18:30
18:45
19:00
19:15
19:30
19:45
20:00
20:15
20:30
CPUUtilization

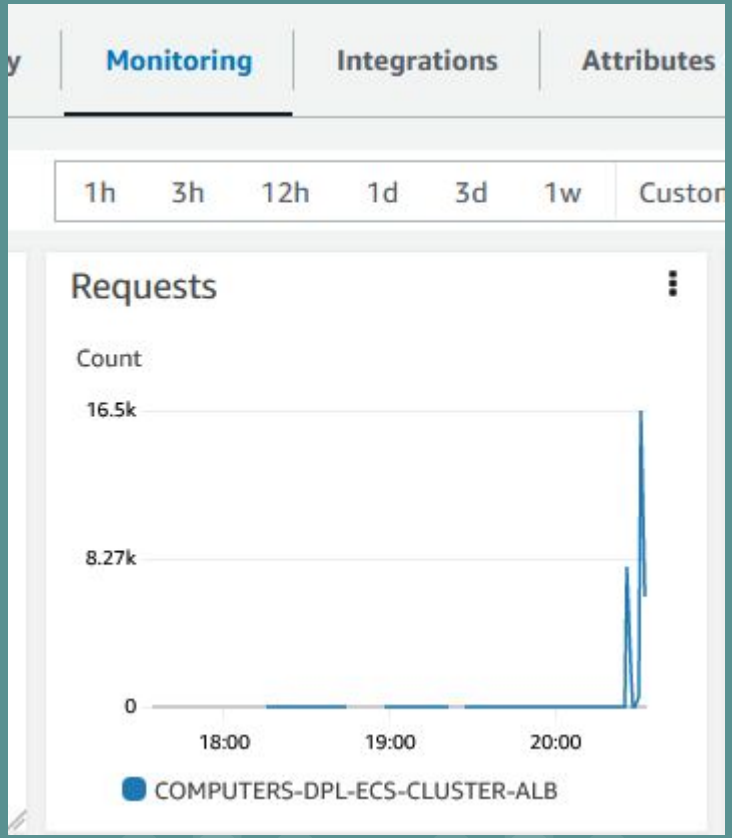
1h 3h 12h 1d 3d 1w Custom 3h Add to dashboard

CPU utilization

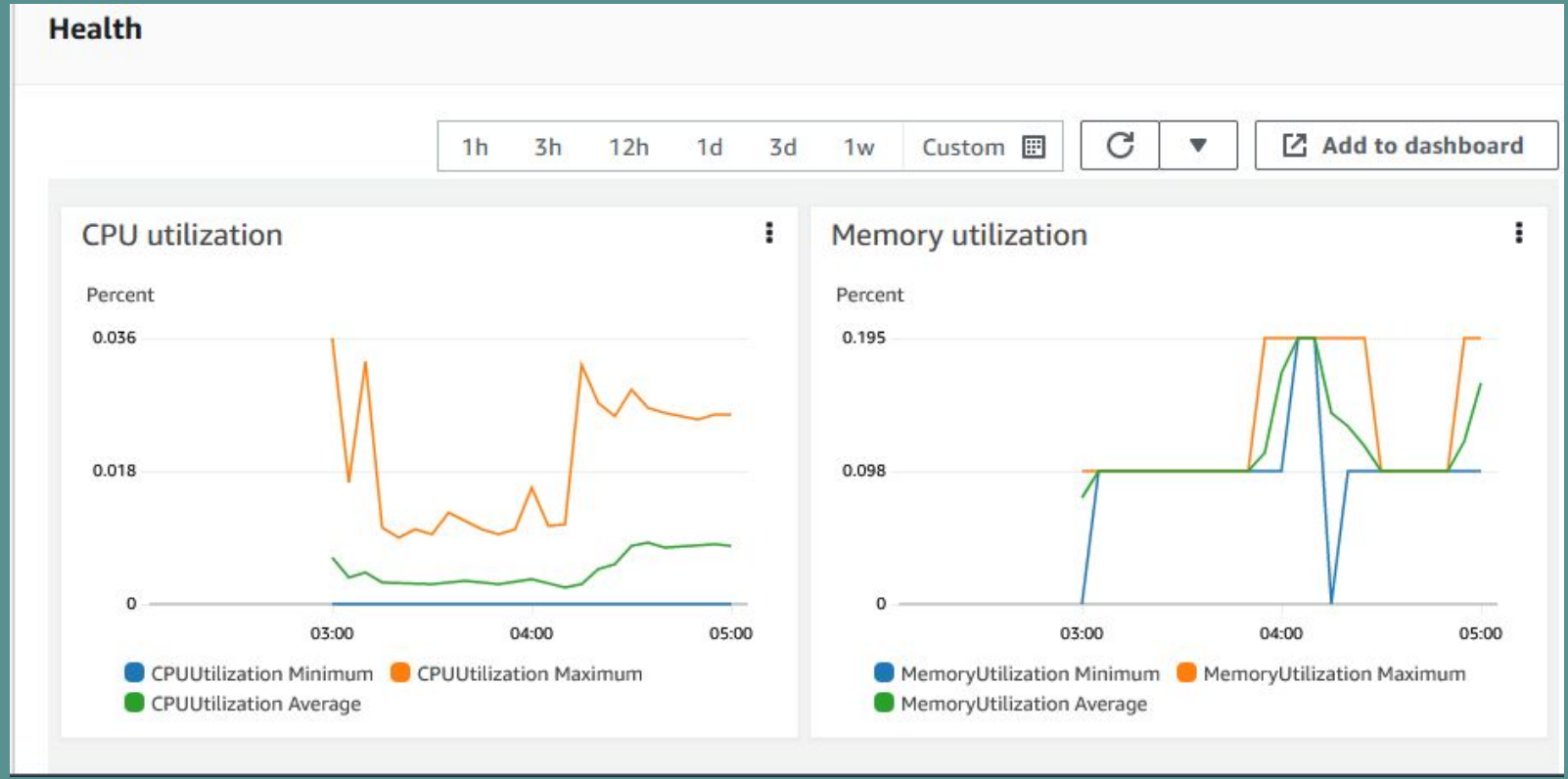


Memory utilization

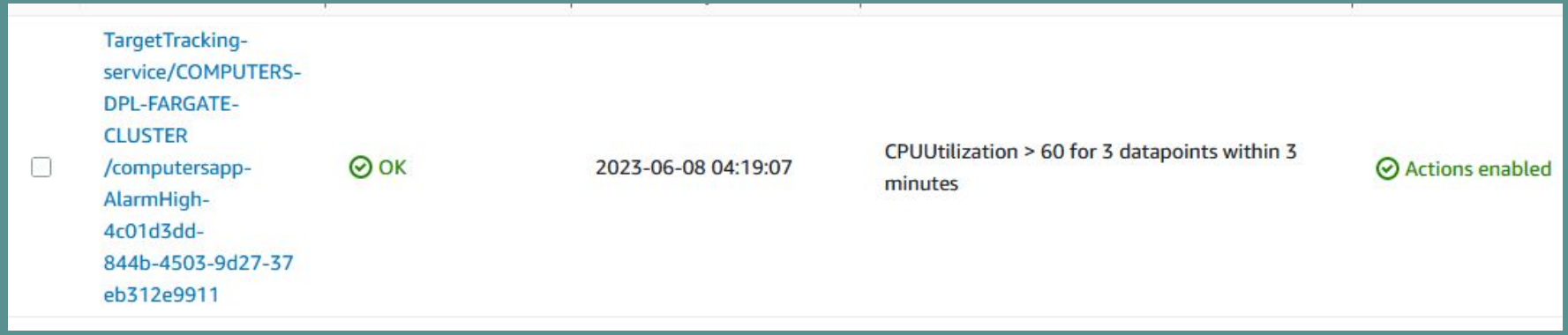




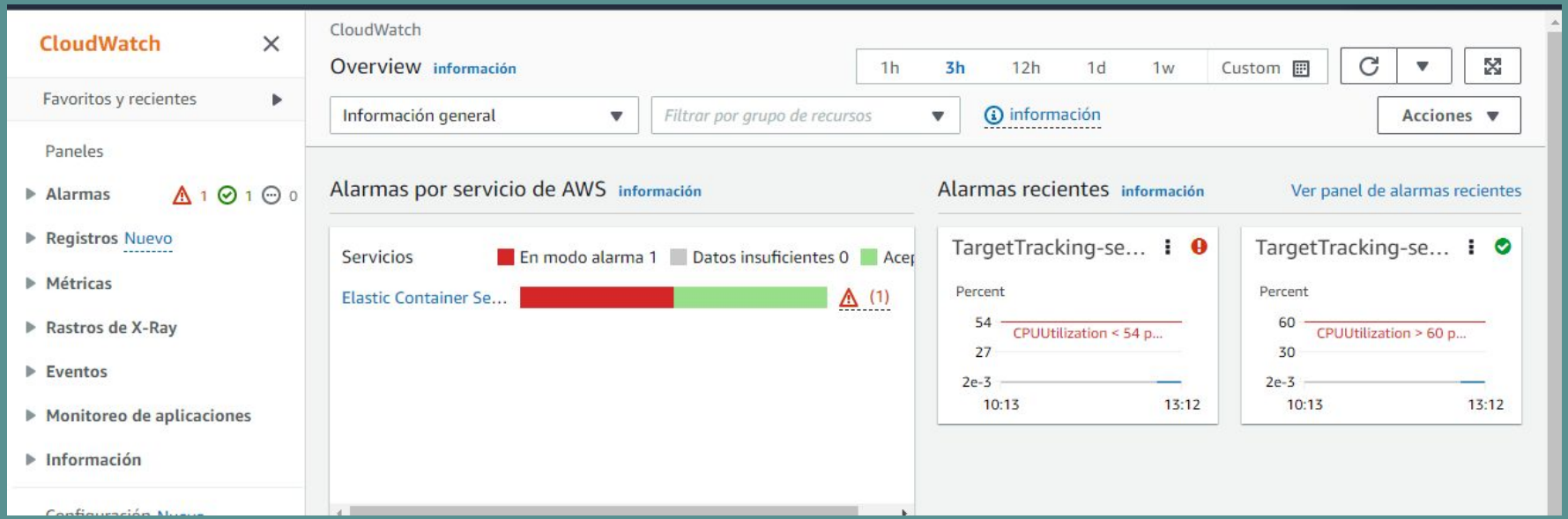
Load Balancer



Usually utilization



Alarm to scale


CloudWatch
Favoritos y recientes
Paneles
Alarmas 1 1 0
Registros Nuevo
Métricas
Rastros de X-Ray
Eventos
Monitoreo de aplicaciones
Información
CloudWatch
Overview información
1h 3h 12h 1d 1w Custom
Información general
Filtrar por grupo de recursos
información
Acciones
Alarmas por servicio de AWS información
Servicios
En modo alarma 1
Datos insuficientes 0
Elastic Container Se...
(1)
Alarmas recientes información
Ver panel de alarmas recientes
TargetTracking-se...
Percent
54
CPUUtilization < 54 p...
27
2e-3
10:13
13:12
TargetTracking-se...
Percent
60
CPUUtilization > 60 p...
30
2e-3
10:13
13:12

Thank you very much!