

**FORMATO DE GUÍA DE PRÁCTICA DE LABORATORIO / TALLERES /  
CENTROS DE SIMULACIÓN – PARA DOCENTES**

**CARRERA:** Computación

**ASIGNATURA:** Sistemas Distribuidos

**NRO. PRÁCTICA:**

4

**TÍTULO PRÁCTICA:** Sistemas de Alta disponibilidad

**OBJETIVO**

- Conocer y experimentar las características de alta disponibilidad y su implementación en sistemas distribuidos

**INSTRUCCIONES** (Detallar las instrucciones que se dará al estudiante):

1. Revisar lo estudiado en relación con tolerancia a fallas

2. Desarrollar las actividades planteadas a continuación

3. La practica será desarrollada en parejas

**ACTIVIDADES POR DESARROLLAR**

1. Desplegar una aplicación de tres capas que contenga como mínimo:
  - a. Un servidor web
  - b. Un servidor de aplicaciones
  - c. Un servidor de base de datos

**RESULTADO(S) OBTENIDO(S):**

Conocimiento básico e implementación de características de alta disponibilidad en sistemas distribuidos

Generación de Pod

```
johnnecul5@cloudshell:~/kubernetes/DockerKubernetes (metal-air-305715)$ cat dockerjava-deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    io.kompose.service: dockerjava
  name: dockerjava
spec:
  replicas: 1
  selector:
    matchLabels:
      io.kompose.service: dockerjava
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        io.kompose.service: dockerjava
    spec:
      containers:
        - image: jtenesacac2/dockerjava
          name: dockerjava
          ports:
            - containerPort: 8080
            - containerPort: 9990
          resources: {}
      restartPolicy: Always
status: {}
```

```
johnnecul5@cloudshell:~/kubernetes/DockerKubernetes (metal-air-305715)$ cat dockerangular-deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: dockerangular
  labels:
    io.kompose.service: dockerangular
spec:
  replicas: 1
  selector:
    matchLabels:
      io.kompose.service: dockerangular
  template:
    metadata:
      labels:
        io.kompose.service: dockerangular
    spec:
      containers:
      - name: dockerangular
        image: jtenesacac2/dockerangular
        imagePullPolicy: Always
        ports:
        - containerPort: 4200
```

Generación del Pod Base De Datos.

```

johnecui5@cloudshell:~/kubernetes/DockerBase/DockerKubernetes (metal-air-305715)$ cat dockerdb-deployment.yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  annotations:
    kompose.cmd: /snap/kompose/19/kompose-linux-amd64 convert -f docker-compose.yml
    kompose.version: 1.21.0 (992df58d8)
  creationTimestamp: null
  labels:
    io.kompose.service: dockerdb
  name: dockerdb
spec:
  replicas: 1
  selector:
    matchLabels:
      io.kompose.service: dockerdb
  strategy:
    type: Recreate
  template:
    metadata:
      annotations:
        kompose.cmd: /snap/kompose/19/kompose-linux-amd64 convert -f docker-compose.yml
        kompose.version: 1.21.0 (992df58d8)
      creationTimestamp: null
      labels:
        io.kompose.network/docker_puente: "true"
        io.kompose.service: dockerdb
    spec:
      containers:
        - env:
            - name: MYSQL_DATABASE
              value: SistemasDistribuidos
            - name: MYSQL_PASSWORD
              value: john0105598254
            - name: MYSQL_ROOT_PASSWORD
              value: root
            - name: MYSQL_USER
              value: john
          image: mysql:latest
          imagePullPolicy: ""
          name: dockerdb
          ports:
            - containerPort: 3306
          resources: {}

```

```

      ports:
        - containerPort: 3306
      resources: {}
      volumeMounts:
        - mountPath: /docker-entrypoint-initdb.d/
          name: dockerdb-claim0
      hostname: dockerdb
      restartPolicy: Always
      serviceAccountName: ""
      volumes:
        - name: dockerdb-claim0
          persistentVolumeClaim:
            claimName: dockerdb-claim0
status: {}

```

Generación de la Persistencia

```
johnnecui15@cloudshell:~/kubernetes/DockerBase/DockerKubernetes (metal-air-305715)$ cat dockerdb-claim0-persistentvolumeclaim.yaml
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  creationTimestamp: null
  labels:
    io.kompose.service: dockerdb-claim0
  name: dockerdb-claim0
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 100Mi
status: {}
```

Generar primero la persistencia después generar el deployment.

### Resultados

```
johnnecui15@cloudshell:~/kubernetes/DockerBase/DockerKubernetes (metal-air-305715)$ kubectl get all

NAME                                READY   STATUS    RESTARTS   AGE
pod/dockerangular-69fbc9c468-bx5h4  1/1     Running   0           45h
pod/dockerdb-cfb749698-mwv9r        1/1     Running   0           46h
pod/dockerjava-7b44645-jjkfx        1/1     Running   0           45h

NAME                                TYPE                      CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
service/dockerangular              LoadBalancer            10.48.14.139    35.238.70.198    4200:32708/TCP   45h
service/dockerdb                   NodePort                 10.48.0.158     <none>           3306:30586/TCP   46h
service/dockerjava                 LoadBalancer            10.48.9.34      35.194.45.101    8080:30926/TCP,9990:32102/TCP 45h
service/kubernetes                 ClusterIP                 10.48.0.1       <none>           443/TCP          2d20h

NAME                                READY   UP-TO-DATE   AVAILABLE   AGE
deployment.apps/dockerangular       1/1     1             1           45h
deployment.apps/dockerdb            1/1     1             1           46h
deployment.apps/dockerjava          1/1     1             1           45h

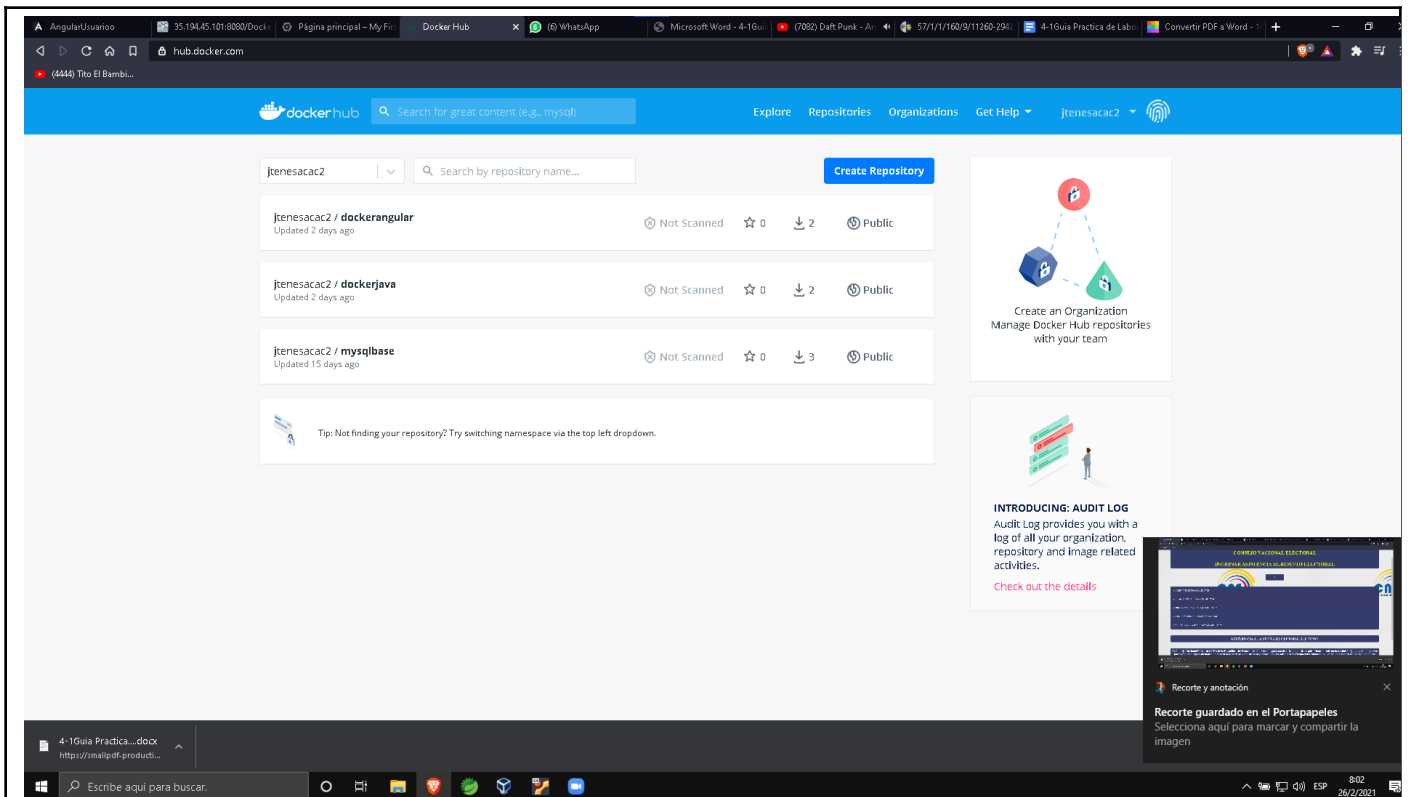
NAME                                DESIRED   CURRENT   READY   AGE
replicaset.apps/dockerangular-69fbc9c468  1         1         1       45h
replicaset.apps/dockerdb-cfb749698        1         1         1       46h
replicaset.apps/dockerjava-7b44645        1         1         1       45h
```

The screenshot shows the official website of the Consejo Nacional Electoral (CNE) of Ecuador. The page has a dark blue header with the text "CONSEJO NACIONAL ELECTORAL" and "INGRESAR ASISTENCIA AL RESINTO ELECTORAL". Below the header is a large blue button with a white plus sign. Underneath the button is a list of email addresses for login:

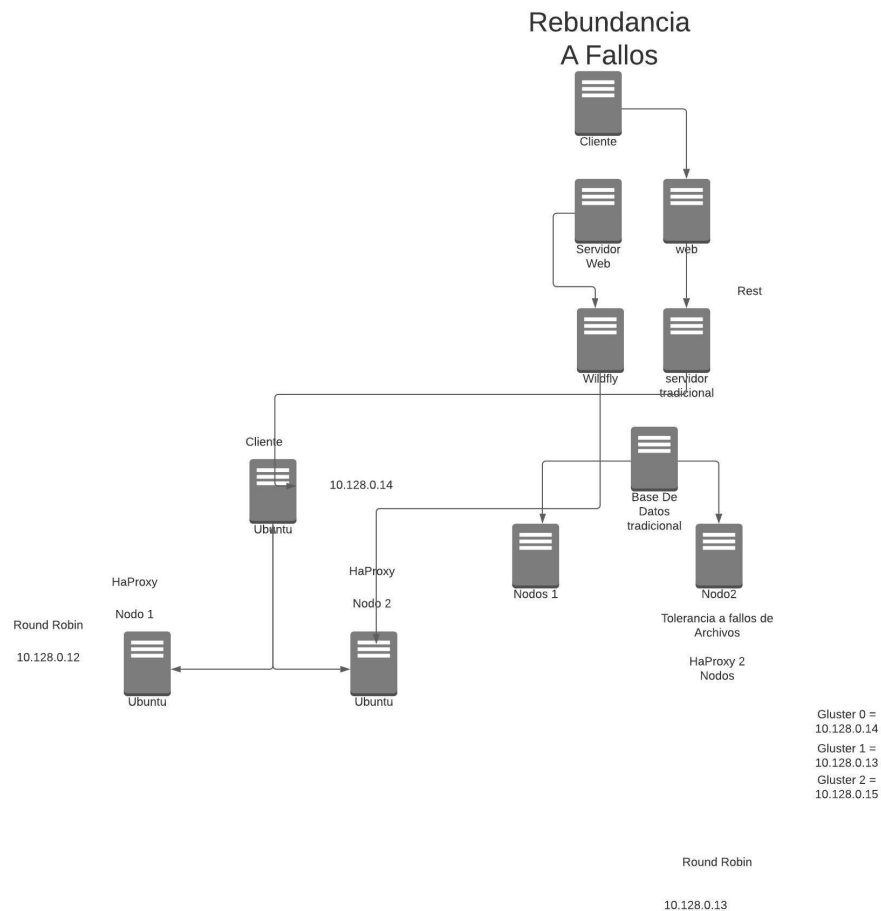
- JOHN.JOHNECUI@GMAIL.COM
- JUAN.ANDRES.JUAN@GMAIL.COM
- SILVIA.PATRICIA.SILVIA@GMAIL.COM
- LENIN.MORENO.LENIN@GMAIL.COM
- CRISTIAN.GARCIA.CRISTIAN@GMAIL.COM

Below the list is another dark blue button with the text "ASISTENCIA A LA JUNTA RECEPTORA DEL VOTO." At the bottom of the page, there is a paragraph of text about the CNE's role in the electoral process.

El Consejo Nacional Electoral (CNE) de la República del Ecuador es el máximo organismo de sufragio en dicho país. Tiene su sede en la ciudad de Quito, está constituido por 5 vocales elegidos mediante un concurso de méritos y oposición organizado por el Consejo de Participación Ciudadana y Control Social para un periodo de 6 años



## Diagrama De Alta Disonibilidad



**CONCLUSIONES:**

- Los estudiantes refuerzan lo aprendido durante las clases, por medio de la experimentación de una practica orientada a la implementación de alta disponibilidad en un sistema distribuido

**RECOMENDACIONES:**

- Asistir a las sesiones de clase.
- Consultar con los docentes las dudas que puedan haber surgido durante las clases y durante las prácticas.

**Docente / Técnico Docente:** Ing. Cristian Timbi

**Firma:** \_\_\_\_\_