

## MANUAL TECNICO

La practica se compone por medio de un clúster de kubernetes que orquesta las imágenes necesarias para su funcionamiento, realizado con herramientas de kubernetes de GCP.

El servicio de comunicación de peticiones es de tipo load balancer con la siguiente configuración, expuesto en el puerto 4000.

```
---
apiVersion: v1
kind: Service
metadata:
  name: svc-demo
  labels:
    app: grpcapp
spec:
  type: LoadBalancer
  selector:
    app: grpcapp
  ports:
    - port: 4000
      targetPort: 4000
      protocol: TCP
```

Se declaro un servicio tipo deployment con una replica de pod, el pod maneja las imágenes de cliente-servidor de nuestra aplicación con comunicación grpc.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: grpc-deploymentn
  labels:
    app: grpcapp
spec:
  replicas: 1
  selector:
    matchLabels:
      app: grpcapp
  template:
    metadata:
      labels:
        app: grpcapp
    spec:
      hostname: grpc-host
      containers:
        - name: client
          image: cocacore7/client_go_201908335
          ports:
            - containerPort: 4000
          env:
            - name: GRCP_SERVER
              value: "grpc-host:50051"
        - name: server
          image: cocacore7/server_node_201908335
          ports:
            - containerPort: 50051
          env:
            - name: RABBIT_HOST
              value: "rabbitmq-0.rabbitmq.rabbits.svc.cluster.local"
            - name: RABBIT_PORT
              value: "5672"
            - name: RABBIT_USERNAME
              value: "guest"
            - name: RABBIT_PASSWORD
              value: "guest"
```

Se realizaron configuraciones para manejar un entorno de rabbitmq como sistema de mensajería, donde se comunica con una aplicación en Go, que consume los mensajes del sistema de mensajería con la siguiente configuración.

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: subscriber-deployment
5    labels:
6      app: subscriber
7  spec:
8    replicas: 1
9    selector:
10     matchLabels:
11       app: subscriber
12    template:
13     metadata:
14       labels:
15         app: subscriber
16     spec:
17       hostname: grpc-host
18       containers:
19         - name: subscriber-cont
20           image: cocacore7/suscriber_go_rabbit_201908335
21           env:
22             - name: ADD_RABBIT
23               value: rabbitmq-0.rabbitmq.rabbits.svc.cluster.local:5672
```

Y por ultimo lo registra en una instancia de VM con base de datos MongoDB para guardar los logs de los juegos que procesa nuestras aplicaciones.

