**Indicaciones para el despliegue y prueba del reto**

1. Crear BD de MongoDB

docker run -d --name mongo-bootcamp -d -p 27017:27017 mongo:latest

1. Iniciar un servicio de Identity Management con Keycloak

docker run -d --name start-keycloak -p 9999:8080 -e KEYCLOAK\_ADMIN=admin -e KEYCLOAK\_ADMIN\_PASSWORD=admin quay.io/keycloak/keycloak:17.0.0 start-dev

1. Crear una instancia de redis para la cache

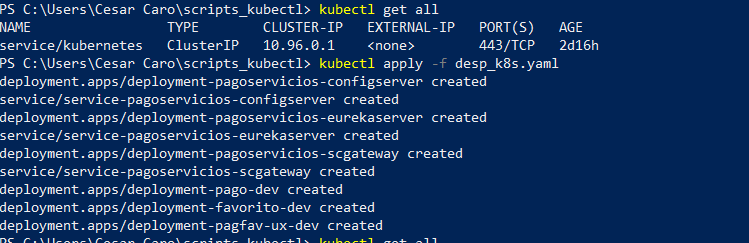
docker run -d --name start-redis -p 6379:6379 redis

1. Iniciar un cluster de kubernetes (puede ser minikube o el que viene en el Docker Desktop de Windows)
2. Desplegar los microservicios desarrollados en el cluster de kubernetes

kubectl apply -f desp\_k8s.yaml

El archivo desp\_k8s.yaml se encuentra en el repositorio:

<https://github.com/cesarac561/pago_de_servicios.git>



**Fuentes de microservicios desarrollados en GitHub**

1. API Negocio de Pago de servicios + Documentación

<https://github.com/cesarac561/pago_de_servicios.git>

1. API Negocio de Favoritos

<https://github.com/cesarac561/pago_servicios_favoritos.git>

1. API de Experiencia

<https://github.com/cesarac561/pago_de_servicios_ux.git>

1. Spring Cloud Gateway

<https://github.com/cesarac561/pago_servicios_scgateway.git>

1. Config Server

<https://github.com/cesarac561/pago_servicios_configserver.git>

1. Eureka Server

<https://github.com/cesarac561/pago_servicios_svc_discovery.git>

**Repositorio de Config Server en GitHub**

<https://github.com/cesarac561/pago_servicios_repoconfig.git>

**Imágenes de los contenedores en DockerHub**

1. API Negocio de Pago de servicios

cesarac561/bootcamp-pago:0.0.1

1. API Negocio de Favoritos

cesarac561/bootcamp-favorito:0.0.1

1. API de Experiencia

cesarac561/bootcamp-pagfav\_ux:0.0.1

1. Spring Cloud Gateway

cesarac561/bootcamp-pagoservicios-scgateway:latest

1. Config Server

cesarac561/bootcamp-pagoservicios-configserver:latest

1. Eureka Server

cesarac561/bootcamp-pagoservicios-eurekaserver:latest

(\*) Todas las imágenes fueron generadas con la IP 192.168.1.9, se debería cambiar por la IP de la máquina donde se haga el despliegue

**Endpoints requeridos de los microservicios**

(\*) En los microservicios se han dejado otros endpoints solo para pruebas

1. API Negocio de Pago de servicios

* obtenerServiciosPorCanal (de ServicioHandler)
* registrarPago (de PagoHandler)

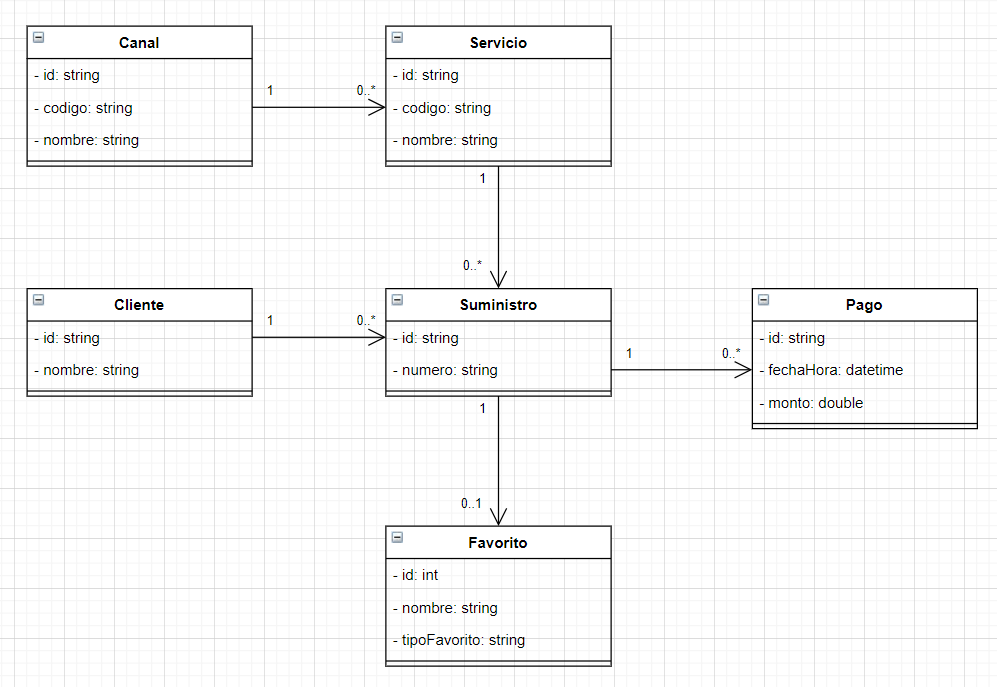
1. API Negocio de Favoritos

* registrarFavorito (de FavoritoHandler)

1. API de Experiencia

* obtenerServiciosPorCanal (de ServicioUxHandler)
* registrarPagoFavorito (de ServicioUxHandler

**Diagrama de Clases**



**Arquitectura**

