



La multinacional *MyHealth* es una empresa del sector salud con presencia en el país durante los últimos treinta años. A lo largo de este tiempo, ha logrado consolidar y fidelizar una gran base de pacientes (los pacientes son los clientes de la organización y suman ciento cincuenta mil aproximadamente). Paralelamente con el crecimiento de sus clientes, la compañía ha tenido que evolucionar las soluciones de TI que soportan los procesos críticos de negocio.

MyHealth ha almacenado los registros de sus pacientes en una base de datos SQL conectada a un Backend desarrollado en Java. Como ha tenido que afrontar los múltiples retos que supone la gestión y la atención de tan numerosa base de clientes, no ha podido realizar su actualización y el Backend ha venido operando, de forma relativamente estable e invariante durante los últimos años con una arquitectura monolítica. También, las reglas del negocio han sido escritas en Java.

Se ha desarrollado una aplicación, denominada *Icesi-Health*, una interfaz para un sistema de registros de pacientes. En ella, la interfaz de usuario está programada utilizando librerías de código abierto (open source) de JavaScript, CSS y HTML5 Canvas. La aplicación se encuentra constituida por: un frontend, un backend y una base de datos. La interfaz de usuario (frontend) y la API (backend) son servidas por un servidor Node.JS Express. Además, la base de datos es servida por una base de datos NoSQL denominada CouchDB. Por ahora, *Icesi-Health* ha evolucionado sus viejas reglas de negocio escritas en Java, y bajo una arquitectura monolítica, a una arquitectura microservicio desplegada en Node.Js

La compañía también ha estado revisando el tema de la computación en la nube, vislumbrando la posibilidad de efectuar una posible migración de *Icesi-Health*. En la actualidad, todo su código se ejecuta en servidores tradicionales (on-premise), funciona bien, por ahora.... Sin embargo, algunos de sus arquitectos de software consideran importante, evolutivo y complementario acelerar el desarrollo de nuevas funcionales de la aplicación (que ya se han venido vislumbrando) y explorar las posibilidades que podría ofrecer el machine learning para aprovechar la base de información con la cuenta y que han acumulado por años.

El área de Investigación, Desarrollo e Innovación (I+D+i) de *MyHealth* ha considerado que, en el curso de Sistemas Distribuidos, podrían ayudarle a resolver dos grandes retos que ha decidido abordar: por una parte, el despliegue de la aplicación usando Kubernetes con miras a poderse desplegar en un entorno productivo; y por otra, documentar las actividades y las lecciones aprendidas del

proceso de despliegue de la aplicación para poder consolidar corporativamente el concepto de Despliegue de Microservicios usando K8S y devops como una cultura de trabajo y para replicar el proceso y las lecciones aprendidas en otros proyectos.

MyHealth considera que este proyecto puede ser desarrollado en 2 horas de trabajo y que, a lo largo de ellas, se pueden desarrollar tres entregas que incluyen:

1. Despliegue frontend:
 - a. El frontend es desplegado como deployment.
 - b. El frontend es alcanza vía internet a través de una IP pública
2. Despliegue backend:
 - a. El backend es desplegado como deployment
 - b. El backend no puede ser alcanzable vía internet
 - c. El backend se conecta con el frontend
3. Despliegue BD:
 - a. La BD es desplegado como deployment
 - b. La BD ha sido inicializada con los datos proporcionados (patient_data)
 - c. La base de datos no se puede ser alcanzable vía internet
 - d. La base de datos tiene conexión con el backend

Bonus:

1. Efectuar una Documentación General del PROCESO y de la CONFIGURACIÓN de las Herramientas.
2. Realizar y Documentar una RETROSPECTIVA DEL PROYECTO (¿Qué salió bien? ¿Qué salió mal? ¿Cómo mejorar?).
3. PULL REQUEST hacia <https://github.com/icesi-ops/sd-finalproject3>

Entradas:

- IcesiHealth APP -> <https://github.com/icesi-ops/ds-exams/tree/master/2021-2/midterm3/IcesiHealth>
- How to create a deployment -> https://github.com/icesi-ops/training_kubernetes/tree/master/01_started
- How to expose APP -> https://github.com/icesi-ops/training_kubernetes/tree/master/02_services
- How to run a node project -> https://github.com/icesi-ops/training_microservices/tree/master/hello-world-nodejs
https://github.com/icesi-ops/training_dev/tree/master/hello-world-nodejs
- Apache CouchDB official image -> https://hub.docker.com/_/couchdb
- Old CouchDB image -> <https://hub.docker.com/r/ibmcom/cloudant-developer/>