

# INF 343 - Sistemas Distribuidos

## Informe Laboratorio N°1: Encapsulamiento con Docker y Comunicación Síncrona y Asíncrona

José Pinto - Ernesto Barría

4 de abril de 2024

### Resumen

El presente informe consta de dos ejes principales, la **comunicación** y el **encapsulamiento**. Para la primera se realizan dos programas en **Go** que se comunican entre sí mediante paso de mensajes, y para la segunda se encapsula uno de los dos programas utilizando la herramienta **Docker**.

## 1. Comunicación Mediante Paso de Mensajes

Para la comunicación entre archivos se realizó una conexión UDP, manteniendo a una comunicación síncrona entre el servidor central y los capitanes. El servidor central se mantiene escuchando los mensajes de los capitanes que son enviados en un tiempo de espera aleatorio pero acotado, además que se asegura que estos enviarán los mensajes. Asimismo el servidor envía una notificación a los capitanes por cada mensaje que recibe, avisando si sigue escuchando o si todos los planetas tienen la misma cantidad de botín, por lo que pueden cesar sus actividades.

```
Capitán 1 encontró botín en Planeta: PE , enviando solicitud de asignación
Capitán 2 encontró botín en Planeta: PA , enviando solicitud de asignación
Capitán 1 encontró botín en Planeta: PC , enviando solicitud de asignación
Capitán 1 encontró botín en Planeta: PD , enviando solicitud de asignación
Capitán 2 encontró botín en Planeta: PD , enviando solicitud de asignación
Capitán 2 encontró botín en Planeta: PC , enviando solicitud de asignación
Capitán 3 encontró botín en Planeta: PA , enviando solicitud de asignación
Capitán 1 encontró botín en Planeta: PB , enviando solicitud de asignación
Capitán 1 encontró botín en Planeta: PE , enviando solicitud de asignación
Capitán 1 encontró botín en Planeta: PA , enviando solicitud de asignación
Capitán 1 encontró botín en Planeta: PC , enviando solicitud de asignación
Capitán 1 encontró botín en Planeta: PF , enviando solicitud de asignación
```

Figura 1: Consola de usuario.go

```

Recepción de solicitud desde el Planeta :PE , del capitán 1

Estado actual de los planetas:
PA: 44 PB: 31 PC: 20 PD: 48 PE: 3 PF: 26

Planeta con menos botín: PE
Botín asignado al planeta PE , cantidad actual: 4

Recepción de solicitud desde el Planeta :PA , del capitán 2

```

Figura 2: Consola de servidor\_central cuando recibe mensajes de usuario.go

```

Todos los planetas tienen la misma cantidad de botín.
Notificando a los capitanes...
El servidor se ha detenido.

```

Figura 3: Consola de servidor\_central cuando todos los planetas tienen la misma cantidad de botín

## 2. Encapsulamiento

Para el encapsulamiento se utilizó la herramienta **Docker** para generar una imagen de servidor\_central.go. En un comienzo hubo problemas con la comunicación entre los archivos al correr la imagen, sin embargo, al utilizar el comando "docker run -p 8080:8080/udp servidor\_central" se solucionó.

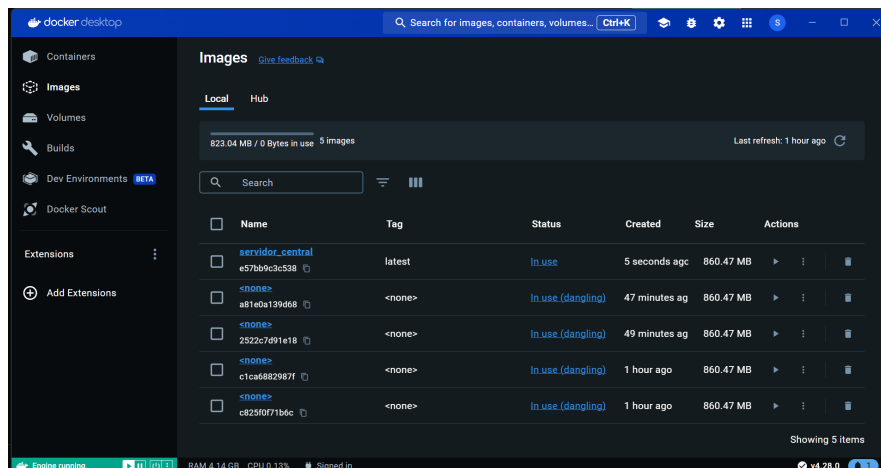


Figura 4: Vista de imágenes desde herramienta Docker

### **3. Preguntas**

#### **3.1. Si se agregara una central que ejecute las mismas funciones que la actualmente implementada. ¿Esto beneficiaría o complicaría al proceso?**

El agregar una central que ejecute las mismas funciones complicaría el proceso, porque se debe crear la conexión entre ellas para mantener actualizados los valores de la cantidad de botín por planeta. Haciendo que cada vez que los capitanes se comuniquen con alguna de las centrales, ambas deban recibir ya sea del capitán o del otro servidor un mensaje, por lo que el que reciba el mensaje primero tiene que notificar al otro, duplicando sus tareas. Una forma de beneficiar el proceso, sería repartir las tareas entre almacén de información (base de datos) y mensajeros, teniendo un servidor central que mantenga el control de la cantidad de botín por planeta, además de poseer la capacidad de encolar solicitudes de ser necesario. Mientras otros 2 o más servidores pueden hacer de mensajeros entre capitanes y el servidor central, con su respectivo balanceador de carga. Esto haría el esquema de comunicación más complejo, sin embargo, podría recibir una mayor cantidad de solicitudes.