

# Trabajo práctico inicial: contenedores

## Objetivo

El objetivo es afirmar conocimiento sobre contenedores e ir profundizando algunas herramientas y conceptos vistos en la parte inicial de la materia y también refrescar conocimiento de la materia predecesora.

## Requerimientos no funcionales y condiciones de entrega

- El código debe ser prolijo y con una **buena separación de responsabilidades**.
- Debe estar subido a un repositorio de código Git.
- Debe tener un README con las instrucciones de instalación y ejemplos de uso. Por favor poner énfasis en este punto, deben estar las instrucciones para que alguien -de cero- y sin experiencia en el trabajo realizado tenga que poder verlo. También debe tener ejemplos de uso
  - Aclaración: no pongan énfasis en la extensión del texto de instrucciones, al contrario. Más corto y conciso, mejor.
  - Debe correr el software ejecutando exactamente las órdenes que ustedes describan en el README, de lo contrario, se devuelve.
- Entregar una forma simplificada de probar la o las API:
  - un proyecto en Postman (usando el Export)
  - una serie de sentencias con cURL.
  - de querer usar otra herramienta, consultarlo con el docente
- La primera entrega debe ser antes del **22/09/2022**
- La entrega del trabajo práctico debe ser un tag del repositorio de código y la URL asociada.
- La solución debe estar **dockerizada**.

## Requerimientos funcionales

- Levantar un servidor que nos devuelva la hora actual del servidor realizado con la tecnología [gRPC](#).
  - Se debe poder pasar un parámetro con la opción de timezone en cuál queremos la respuesta.
  - El contenedor gRPC debe tener una configuración de acceso lo más restrictiva posible.
- Levantar una API REST que a su vez llame a este servicio en el endpoint "time", que también se pueda pasar un parámetro timezone como parámetro.
  - La respuesta debe contener en diferentes campos:

- el timezone elegido
  - la hora en formato yy/mm/dd
  - el [Unix epoch time](#)
- La API REST debe guardar un log de acceso, con cada request, la hora y la ip de origen.
  - Esta información debe sobrevivir si el contenedor es eliminado.
- La API REST debe poder devolver el resultado tanto en JSON como en otro formato a elegir.
- Agregar un archivo en el repositorio en formato Markdown que responda las siguientes preguntas:
  - ¿Cómo resolvió la conexión entre el contenedor REST y el gRPC, cómo se ven entre sí? ¿Utilizó algún modo de restricción?
  - Comparar REST sobre HTTP con gRPC. Definir en qué contexto una tecnología es conveniente por sobre la otra o si no existen diferencias de uso.