

# Trabajo Práctico Final

## Entorno de Programación

### Introducción

#### Descripción

El objetivo del trabajo práctico es diseñar y escribir un programa para procesar un lote de imágenes. Este programa consta de tres partes principales:

1. En primer lugar se debe poder generar y descargar las imágenes a través de un servicio web.
2. Luego se debe aplicar una transformación solamente a las imágenes de personas.
3. Finalmente se debe generar un archivo comprimido con las imágenes procesadas.

Todo el trabajo debe ser realizado bajo control de versiones, con participación de todos los integrantes y debe ejecutarse dentro de un contenedor.

#### Pautas generales

- El trabajo práctico deberá ser realizado en grupos de dos o tres estudiantes.
- Deberá informarse al docente de la comisión cómo está conformado su equipo de trabajo vía mail.
- Todos los integrantes deben conocer todos los aspectos del trabajo entregado.
- Se admite una única entrega final, es por ello que solicitamos revisen muy bien las funcionalidades previamente a la entrega formal, que debe comunicarse con un mail al docente de la comisión.

#### Aspectos evaluativos

- Programación bash.
- Manejo de control de versiones.
- Uso de contenedores.
- Generación de documentación.

### Enunciado

El trabajo consiste en generar un contenedor que al ejecutarse presente un menú de opciones para procesar un lote de imágenes. Deberán editar un **Dockerfile** y construir una imagen.

Las imágenes a analizar deben ser descargadas o generadas desde internet y deben almacenarse dentro de un contenedor para su posterior análisis. Luego del análisis, deberá generarse un archivo disponible fuera del contenedor.

Tanto los scripts, como el **Dockerfile**, deben estar en el repositorio gitlab o github que creen para realizar el desarrollo en equipo.

Deberán trabajar en el repositorio manteniendo la prolijidad y las buenas prácticas de git. Una estrategia muy recomendable es realizar una rama por cada ejercicio e ir integrando a la rama principal a medida que los scripts que resuelven cada ejercicio estén listos. Todos los integrantes deben realizar commits en el repositorio.

En ese mismo repositorio debe estar la documentación suficiente para comprender cómo desplegar el contenedor y ejecutar la aplicación, con algunas capturas de ejecución o lo que consideren logre explicitar el funcionamiento y visualizar resultados. Debe ser utilizando un archivo **README.md** el cual estará presente cuando se accede al repositorio en gitlab o github.

## Scripts

Se deberán programar los siguientes scripts:

- **generar.sh**: Genera imágenes utilizando algún servicios web. Tener en cuenta que al descargar de una página conviene usar un sleep entre descarga y descarga para no saturar el servicio y evitar problemas. Se debe poder indicar por argumento cuantas imágenes generar y se deben asignar nombres de archivo al azar de una lista de nombres de personas. Luego se deben comprimir las imágenes, y generar un archivo con su suma de verificación.
- **descomprimir.sh**: Se debe poder indicar por argumento dos archivos (uno con las imágenes comprimidas y otro con una suma de verificación). Si ocurrió algún error se debe informar al usuario de lo contrario se procede a descomprimir.
- **procesar.sh**: Se deberán recortar las imágenes a una resolución de 512\*512 con alguna utilidad como **imagemagick**. Solamente deben procesarse aquellas imágenes que tengan nombres de personas *válidos*. Entiéndase por nombres de personas *válidos* a cualquier combinación de palabras que empiecen con un letra mayúscula y sigan por minúsculas.
- **comprimir.sh**: Finalmente, luego de procesadas las imágenes, se debe:
  - generar un archivo con la lista de nombres de todas las imágenes.
  - generar un archivo con la lista de nombres válidos.
  - generar un archivo con el total de personas cuyo nombre finaliza con la letra **a**.
  - por último, generar un archivo comprimido que incluya los archivos generados en los items anteriores y *todas* las imágenes. El archivo comprimido debe poder accederse desde fuera del contenedor.
- **menu.sh**: Debe incluirse un menú para elegir cada una de las opciones anteriores.

## Herramientas útiles

**Generación de imágenes** Para generar imágenes de personas, puede utilizarse el siguiente enlace: <https://thispersondoesnotexist.com/>. Pueden modificarse tanto las dimensiones de la imagen como la clase a la cual pertenece.

**Nombres de personas** El siguiente dataset <https://raw.githubusercontent.com/adalessandro/EdP-2023-TP-Final/main/dict.csv> contiene una lista de nombres de personas que pueden utilizarse en los scripts.

**Imagemagick** ImageMagick es un conjunto de utilidades de código abierto para mostrar, manipular y convertir imágenes. En particular la utilidad `convert` permite lograr la transformación planteada en este TP. Puede utilizarse el comando:

```
convert entrada.jpg -gravity center -resize 512x512+0+0 \  
-extent 512x512 salida.jpg
```