

Instalación de Simulador Genérico en entorno Docker

Contenido

1	Introducción.....	2
2	Instalación.....	2
3	Adición de una aplicación	3
4	Ejecución.....	3
4.1	Abrir visualizador	3
5	Interacción con el simulador.....	3

1 Introducción

El simulador genérico AlMarSimulator es un software que permite la simulación de sistemas físicos y de su interacción con las E/S de un computador. Se ejecuta en un contenedor Docker para aislarlo completamente del equipo del usuario.


El usuario puede realizar aplicaciones cliente en su computador que interactúen con las E/S del simulador, y observar el comportamiento obtenido. Se dispone de librerías para C/C++, Python y Matlab que permiten realizar las aplicaciones cliente de forma sencilla.

El simulador se puede ejecutar en cualquier entorno Windows (10 y superior), Linux (64 bit) ó Mac.

2 Instalación

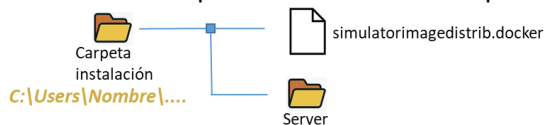
Para utilizar el simulador, es necesario instalar previamente la aplicación Docker Desktop <https://www.docker.com/products/docker-desktop/> . Para entorno Windows, será necesario habilitar la instalación adicional de WSL 2.0 (Windows System for Linux) según se indica en <https://docs.docker.com/desktop/install/windows-install/> .

Si no es posible ejecutar Docker Desktop por las características del ordenador disponible (o por el espacio de disco consumido por Docker+WSL), contactar con ialvarez@uniovi.es para solicitar una copia ejecutable directamente en Windows; utilice esta opción solamente si confía en el desarrollador de la aplicación específica, ya que el programa que se ejecute tiene acceso a los recursos de su máquina.

Al ejecutar por primera vez Docker Desktop, se recomienda entrar en Settings (icono ) y deshabilitar en la pestaña general el checkbox "Start Docker Desktop when you sign in to your computer", para evitar el consumo de recursos cuando no se necesite utilizar la aplicación.

A continuación, con el programa Docker Desktop en funcionamiento, se descarga desde <http://isa.uniovi.es/~ialvarez/Curso/descargas/SimuladorDocker/simuladorimagedistrib.zip> la imagen de la aplicación general, y se deben realizar los siguientes pasos:

- ❑ Descomprimir en una carpeta de la máquina (carpeta instalación a partir de ahora)
- ❑ Crear una subcarpeta Server en dicha carpeta



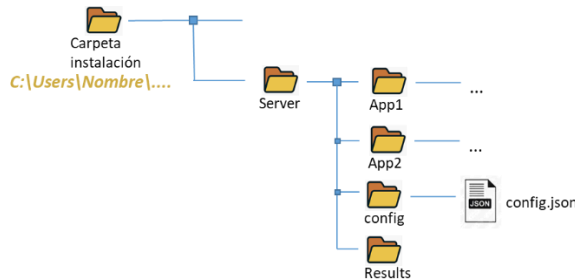
- ❑ Abrir una ventana de comandos, y cambiar (comando Cd) a la carpeta de instalación:

```
Simbolo del sistema
C:\Users\Nombre> cd Ruta\A\Carpeta\Instalacion
C:\Users\Nombre\Ruta\A\Carpeta\Instalacion >
```

- ❑ Ejecutar en la ventana de comandos (una sola línea por cada comando):
 - `docker load -i simuladorimagedistrib.docker`
 - `docker run -it -p 8080:8080 -p 8022:22 -v "%CD%\Server":/opt/UserApps --name=SimulatorContainer --network=bridge simuladorimagedistrib /home/almar/Server/start.sh`
- ❑ Finalmente, se puede borrar el archivo `simuladorimagedistrib.docker` de la carpeta de instalación.

3 Adición de una aplicación

Las aplicaciones que permiten simular diferentes sistemas físicos y su interacción con el computador se descargan individualmente. Para cada una de ellas, se descomprime el archivo de aplicación en la carpeta **Server** de la instalación, antes de arrancar el simulador.



Adicionalmente, puede requerirse modificar algunos archivos en el directorio de aplicación (usuario, clave, tiempo máximo de utilización), cuyos contenidos e instrucciones serán enviados por e-mail.

4 Ejecución

Cada vez que se desee ejecutar el simulador, se deben realizar los pasos siguientes:

- ❑ Ejecutar Docker Desktop (si se ha deshabilitado su ejecución al inicio, como se ha recomendado)

A continuación:

- ❑ Abrir una ventana de comandos
- ❑ Ejecutar:
 - `docker start SimulatorContainer`
 - `docker exec SimulatorContainer /home/almar/Server/start.sh`
- ❑ Para terminar, pulsar Ctrl-C repetidamente hasta que finalice la aplicación, y a continuación
 - Ejecutar `docker kill SimulatorContainer`
 - Finalizar la aplicación Docker Desktop
 - Terminar el servicio Docker Desktop en la bandeja de iconos del sistema

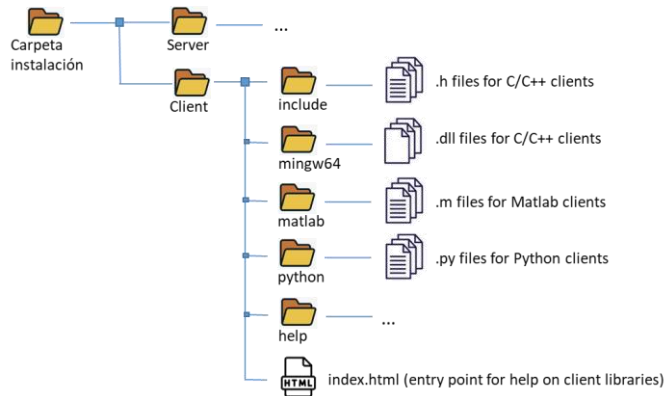
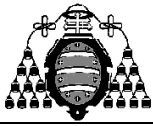
4.1 Abrir visualizador

La ejecución del simulador se puede visualizar desde la página web <http://127.0.0.1:8080>. Solicitará usuario y clave (por defecto: alumno / ISAUNIOVI), y a continuación mostrará un menú con las aplicaciones disponibles en la imagen. Una vez seleccionada la que se desea, pulsar la opción Run para que comience su funcionamiento.

Algunas aplicaciones disponen de un botón Grabar que permite registrar los resultados. Debe utilizarse cuando se dispone de una ejecución operativa cuyos resultados se desean registrar.

5 Interacción con el simulador

La interacción con el simulador se puede realizar desde un programa realizado en C/C++, Matlab ó Python, según las necesidades del usuario. Para ello, es necesario descargar y descomprimir en una carpeta del usuario (se recomienda carpeta Client en la misma carpeta de instalación) el archivo http://isa.uniovi.es/~ialvarez/Curso/descargas/SimuladorDocker/Client_Libraries.zip. El resultado será un árbol de directorios como el siguiente:



En todos los casos, las operaciones a ejecutar desde el programa de usuario son similares:

- 1) Conectar con el simulador, mediante una función `Simulator_ConnectWss()`. Esta función requiere varios argumentos (aplicación, usuario, etc.) que se pueden obtener haciendo click en "Mostrar info clientes" en el interfaz web. Los argumentos mostrados se pueden copiar y pegar directamente en el código de la aplicación.
- 2) Enviar y recibir datos del simulador. Se debe realizar a intervalos regulares, múltiplos de 25ms, para evitar la saturación del programa. Existirán funciones para interactuar con las E/S analógicas y digitales, PWM, contadores, LCD, etc. dependiendo del simulador ejecutado.

El archivo de ayuda `index.html` que se obtiene en la descarga es el punto de entrada para la ayuda de las funciones y clases disponibles, según el lenguaje de programación que se desee utilizar para el cliente. Cada una de ellas contiene ejemplos de su uso.

Librería para clientes de AIMarSimulator 1
Librería para E/S de simuladores genéricos (C/C++/Matlab/Python)

Main Page Related Pages Classes Files

Librería para clientes de AIMarSimulator

- Documentación para clientes de AIMarSi
- Documentación para clientes Matlab de
- Documentación para clientes Python de
- Classes
- Files

Documentación para clientes de AIMarSimulator

Cliente C/C++ (computador genérico)
Ayuda funciones disponibles: [UserLibSimulator.h](#)
Compilación en entorno Qt-Creator y ejemplo de uso: [Usar UserLibSimulator en Qt-Creator](#)

Cliente C++ (estilo Arduino)
Ayuda: [AlMarDuino.h](#)
Compilación en entorno Qt-Creator y ejemplo de uso: [Usar UserLibAlMarDuino en Qt-Creator](#)
Clase [Servo](#)
Clase [Serial](#)

Cliente Matlab
Ayuda: [Matlab](#)

Cliente Python
Ayuda: [Python](#)