

Instrucciones de Instalación "AutoGOAL Web"

Requisitos previos:

Instalación de Python

Es necesario contar con Python instalado en el sistema para el correcto funcionamiento de "Autogoal Web". Se podrá obtener desde el sitio web oficial de Python. Durante la instalación, es recomendable seleccionar la opción "Add Python to PATH" para facilitar su ejecución desde la línea de comandos.

Instalación de Docker

Docker es esencial para contenerizar y ejecutar la aplicación. Para instalar Docker, se debe visitar la página oficial de Docker y seguir las instrucciones correspondientes.

Instalación VS Code:

Visual Studio Code nos proporciona un entorno para programar, al cual podemos añadir diferentes extensiones, como la extensión de Docker, que es muy recomendable para poner en marcha esta aplicación.

Paso Previo

Antes de nada, habrá que realizar la integración en GitHub de la versión de AutoGOAL-web, desde el fork de esta dirección <https://github.com/jlm111-ua/autogoal>.

Paso 1: Descargar imagen de AutoGOAL

Para poner en marcha el servidor dentro de la librería de AutoGOAL, será necesario descargarse la imagen Docker que contiene el framework. Para ello, desde DockerHub buscaremos la imagen más actualizada de este framework.

En mi caso utilice la imagen con el tag "full-latest", pero con la imagen "core" sería suficiente.

Para descargar la imagen escribiremos por terminal el siguiente comando:

```
docker pull autogoal/autogoal:core
```

Paso 2: Creación del contenedor de AutoGOAL y preparación del entorno

Una vez descargada la imagen en Docker, pasaremos a lanzar un contenedor. Para ello, se podrá hacer desde terminal con el siguiente comando:

```
docker run -it autogoal/autogoal:core
```

Una vez ejecutado el contenedor, mediante la opción de "Attach Visual Studio Code" que aparece al hacer clic derecho sobre el contenedor en el entorno de VS Code, podremos ver que la estructura se asemeja a un directorio de Linux.

Dentro del contenedor, habrá que instalar el paquete de websockets, fastapi y uvicorn. Para ello, nos haremos superusuarios e instalaremos los paquetes a continuación

```
sudo su -
```

```
pip install websockets
```

```
pip install uvicorn
```

```
pip install fastapi
```

Además, también instalaremos autogoal-contrib con:

```
pip install autogoal-contrib
```

Paso 3: Poner en marcha el servidor y la API para la escucha de peticiones.

Nos situaremos en la ubicación `/home/coder/autogoal/autogoal/web` y ejecutaremos desde terminal:

```
python backend.py (Para ejecutar el servidor)
```

Posteriormente, en otro terminal en la misma ubicación, ejecutaremos la API simultáneamente:

```
uvicorn main:app --host 0.0.0.0 --port 4239 (Para ejecutar la API)
```

Paso 4: Poner en funcionamiento la aplicación web

La aplicación web en este caso se encuentra en un fichero zip comprimido llamada "autogoal-python". Para ponerla en marcha, extraemos los archivos. Una vez extraídos, desde el terminal nos situamos en el directorio raíz del proyecto, donde se encuentra fichero **Dockerfile**.

Una vez situados, creamos la imagen Docker con el comando:

```
docker build -t autogoal-python .
```

Una vez construida la imagen, la aplicación se puede ejecutar con el siguiente comando:

```
docker run -p 3000:3000 autogoal-python
```

Cabe recalcar que es importante ejecutar previamente el contenedor de AutoGOAL, ya que, si no, habrá que realizar un ajuste en las IPs utilizadas. Para verificar su funcionamiento, accederemos desde el navegador web a la dirección <http://localhost:3000>. Desde esta dirección, accederemos a la aplicación web.

En caso de realizar cambios en el código fuente y querer actualizar la imagen Docker, se debe reconstruir la imagen con el comando **docker build** y después reiniciar el contenedor.

Para detener el contenedor, se puede utilizar **docker ps** para encontrar el ID del contenedor y luego **docker stop** [CONTAINER_ID].