

## Despliegue en servidor

Actualmente se desplegó en un servidor de la biblioteca, reemplazando la versión anterior con esta nueva versión, y se dejó a disposición un manual de implementación para que pueda ser desplegado posteriormente a necesidad del cliente.

## Manual de implementación

En este manual se detallarán los pasos para poder instalar el sistema mediante dockerización en un entorno Linux

Cabe destacar que los componentes generados pueden también instalarse directamente en servidores web si así lo desean.

## Requisitos del Servidor de Hardware

- Al menos 16 GB de espacio de memoria RAM
- Núcleo de 64 bits y soporte de CPU para la virtualización
- Compatible con tecnología de virtualización KVM
- Entorno de escritorio de Gnome, KDE o MATE
- QEMU a partir de la versión 5.2, por lo que se recomienda usar la más actualizada

## Requisitos del Servidor de Software

- Ubuntu 20.4 o superior
- Mysql 5 o superior
- Elasticsearch 8 o superior

## Instalación de Docker

### 1. Preparar el sistema

Primero debes asegurarte de tener las **últimas versiones** de tu sistema y todos los paquetes. Solo has de introducir los siguientes comandos en tu terminal para actualizar tu sistema:

Shell

```
sudo apt-get update  
sudo apt-get upgrade
```

Antes de poder instalar Docker, si ya has utilizado una versión beta o de previsualización de Docker, tienes que **eliminarlas**, incluyendo todos los archivos creados:

Shell

```
sudo apt remove docker-desktop
```

```
rm -r $HOME/.docker/desktop  
sudo rm /usr/local/bin/com.docker.cli  
sudo apt purge docker-desktop
```

## 2. Añadir el repositorio de Docker

Shell

```
sudo apt-get install \  
ca-certificates \  
curl \  
gnupg \  
lsb-release
```

Luego, puedes añadir la **clave GPG** de Docker al sistema:

Shell

```
sudo mkdir -p /etc/apt/keyrings  
curl -fsSL https://download.docker.com/linux/ubuntu/gpg |  
sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
```

Por último, activa el repositorio de Docker con este comando:

Shell

```
echo \  
"deb [arch=$(dpkg --print-architecture) signed-  
by=/etc/apt/keyrings/docker.gpg]  
https://download.docker.com/linux/ubuntu \  
$(lsb_release -cs) stable" | sudo tee  
/etc/apt/sources.list.d/docker.list > /dev/null
```

## 3. Instalar el motor de Docker

Ahora puedes descargar el motor de Docker en Ubuntu 20.04, algo que también puedes hacer desde la terminal. Usa el siguiente comando para instalar la **versión más reciente de Docker** en tu sistema:

Shell

```
sudo apt-get install docker-ce dock ber-ce-cli containerd.io  
docker-compose-plugin
```

Si tienes pensado instalar una **versión específica de Docker**, consulta primero las versiones disponibles:

Shell

```
apt-cache madison docker-ce | awk '{ print $3 }'
```

#### 4. Comprobar que la instalación se ha realizado correctamente

Una vez instalado el programa, puedes usar el siguiente comando para iniciar un “Hello World” en un Docker container y comprobar que se ha instalado correctamente:

Shell

```
sudo docker run hello-world
```

### Instalación de Docker Compose

#### 1. Descargar e instalar Docker Compose

Descarga la última versión de Docker Compose del repositorio de GitHub oficial. Ejecuta el siguiente comando en tu terminal:

Shell

```
$ curl -SL  
https://github.com/docker/compose/releases/download/v2.17.2/docker  
-compose-linux-x86_64 -o /usr/local/bin/docker-compose
```

A continuación, otorga permisos de ejecución a Docker Compose:

Shell

```
$ sudo chmod +x /usr/local/bin/docker-compose
```

La opción --version te permite comprobar si Compose se ha instalado correctamente.

Shell

```
$ docker-compose --version
```

## Instalación del sistema

Teniendo los pasos previos configurados vamos a pasar a la instalación del sistema

1. Loguearse en la terminal de linux como root
2. Crear una carpeta donde se copiara el archivo docker-compose.yml proporcionado en el código fuente
3. Ejecutar el comando parado en la carpeta que contiene el .yml

Shell

```
docker compose stop && docker compose rm && docker compose pull && docker-compose up -d
```

Este proceso levantara los servicios necesarios del sistema

## Configuración del sistema

El sistema cuenta con dos archivos de configuración que deben ser establecidos antes de generar las imágenes de docker

1 - Configuración de la api:

El archivo es el que se encuentra ubicado en la dirección

- **<código-fuente>\biblioteca\Biblioteca-back\Biblioteca.API\appsettings.Production.json**

Los parámetros más importantes son

```
"ConnectionStrings": {
```

```
  "sqlconnection":
```

```
  "server=mysql;port=3306;database=Biblioteca;user=root;password="
```

```
  },
```

El cual posee la dirección a la base de datos Mysql

```
"ELK": {
```

```
  "Connection": "http://es01:9200",
```

```
  "DefaultIndex": "library_document",
```

```
  "EnableLogs": false
```

```
},
```

El cual posee la dirección a la base de datos no relacional Elasticsearch