Guía Completa de Laboratorio: Frappe + Docker + NGINX + HTTPS

# Índice de Contenido

1. 1. Introducción
2. 2. Requisitos Previos
3. 3. Estructura del Proyecto
4. 4. Paso a Paso de Despliegue
5. 5. Verificación de Servicios y Redirección
6. 6. Configuración de HTTPS con Let's Encrypt
7. 7. Pruebas de Funcionamiento
8. 8. Buenas Prácticas Aplicadas
9. 9. Conclusión

# 1. Introducción

Este documento describe el procedimiento completo para desplegar un entorno de desarrollo con Frappe ERP utilizando Docker en un sistema Windows 11 con WSL2 o en una VM Ubuntu. El entorno incluye servicios como Redis, MariaDB, Frappe y NGINX, con redirección del puerto 80 al 9000 y soporte para HTTPS mediante Let's Encrypt.

# 2. Requisitos Previos

- Windows 11 con WSL2 habilitado o una VM Ubuntu 22.04  
- Docker y Docker Compose instalados  
- Acceso a terminal Bash  
- Opcional: dominio apuntando a la IP pública para HTTPS

# 3. Estructura del Proyecto

El proyecto debe tener los siguientes archivos y estructura:  
- docker-compose.yml  
- nginx.conf  
- init.sh  
- directorio certbot/ con subdirectorios conf/ y www/  
- README\_EXTENDIDO.md (opcional para documentación interna)

# 4. Paso a Paso de Despliegue

1. Crear directorio de trabajo:  
  
```bash  
mkdir ~/frappe\_lab && cd ~/frappe\_lab  
```  
  
2. Copiar los archivos necesarios:  
  
```bash  
cp /mnt/c/Users/TU\_USUARIO/Downloads/\* .  
```  
  
3. Ejecutar los servicios:  
  
```bash  
docker-compose down  
docker-compose up --build -d  
```  
  
4. Ejecutar el script de inicialización:  
  
```bash  
docker exec -it frappe bash  
./init.sh  
```

# 5. Verificación de Servicios y Redirección

1. Ver contenedores activos:  
  
```bash  
docker ps  
```  
  
2. Validar acceso:  
  
```bash  
curl http://localhost  
```  
  
3. Ver logs de NGINX:  
  
```bash  
docker logs nginx  
```  
  
El navegador debe redirigir correctamente de `http://localhost` al puerto 9000.

# 6. Configuración de HTTPS con Let’s Encrypt

1. Configurar dominio y DNS  
  
2. Añadir servicio certbot en `docker-compose.yml`  
  
3. Ejecutar el contenedor certbot:  
  
```bash  
docker-compose run certbot  
```  
  
4. Verificar certificados SSL emitidos y disponibles en `/etc/letsencrypt/live/`

# 7. Pruebas de Funcionamiento

- Acceso a http://localhost  
- Redirección por NGINX al puerto 9000  
- Acceso HTTPS si el dominio está configurado  
- Validación de logs de Frappe y NGINX

# 8. Buenas Prácticas Aplicadas

- Servicios separados en contenedores  
- Uso de nombres DNS en red Docker  
- Sin uso de servicios del host como chpasswd o supervisor  
- Soporte para HTTPS sin intervención del sistema host

# 9. Conclusión

Con esta guía se puede desplegar de forma segura y aislada un entorno de Frappe ERP completo, incluyendo acceso web, redirección y soporte para HTTPS. Todos los pasos fueron validados y documentados para uso profesional y educativo.

# 10. Opcional: Configuración de HTTPS con Renovación Automática

Esta sección es opcional y puede aplicarse después de levantar correctamente el laboratorio en Windows 11 o Ubuntu con WSL2. Requiere tener un dominio registrado y apuntando a la IP pública del entorno.  
  
Se utilizará Let's Encrypt para emitir los certificados SSL válidos y gratuitos, y se configurará la renovación automática usando `cron`.  
  
Pasos:

1. Verifica que tu dominio (ej: midominio.com) apunte al servidor donde está desplegado Frappe.  
  
2. Añade los siguientes servicios en `docker-compose.yml` si no están:  
  
```yaml  
 certbot:  
 image: certbot/certbot  
 volumes:  
 - ./certbot/www:/var/www/certbot  
 - ./certbot/conf:/etc/letsencrypt  
 command: certonly --webroot --webroot-path=/var/www/certbot --email tu-email@dominio.com --agree-tos --no-eff-email -d midominio.com  
```  
  
3. Corre Certbot manualmente una vez para emitir el certificado:  
  
```bash  
docker-compose run certbot  
```  
  
4. Crea un archivo `renew\_ssl.sh` con el siguiente contenido:

```bash  
#!/bin/bash  
docker-compose run --rm certbot renew --webroot --webroot-path=/var/www/certbot  
docker-compose kill -s HUP nginx  
```

5. Otorga permisos de ejecución:  
  
```bash  
chmod +x renew\_ssl.sh  
```  
  
6. Agrega una tarea en crontab (dentro de tu entorno WSL o Linux VM):  
  
```bash  
crontab -e  
```  
  
Y añade la línea:  
  
```bash  
0 3 \* \* 1 /home/tu\_usuario/frappe\_lab/renew\_ssl.sh >> /home/tu\_usuario/frappe\_lab/renewal.log 2>&1  
```  
  
Esto renovará automáticamente el certificado cada lunes a las 3 AM.

⚠️ Importante: el contenedor nginx debe tener el volumen mapeado con `/etc/letsencrypt` y reiniciarse para aplicar el nuevo certificado.