

Ejecución mediante aplicación Docker

Prerrequisitos:

- Clonar el repositorio del proyecto
- Tener instalado Docker
- Tener puertos 3000, 4000 y 3306 **DISPONIBLES**

Opcional:

- Tener instalado docker desktop

1. Crear archivo de variables de entorno. env para el frontend y el backend

a. Variables frontend

```
VITE_HOST=0.0.0.0  
VITE_PORT=3000  
VITE_BACKEND_URL="http://localhost:4000"
```

VITE_HOST=0.0.0.0 significa que el servidor Vite estará escuchando en todas las interfaces de red de la máquina.

VITE_PORT=3000 significa que el servidor Vite estará escuchando en el puerto 3000, puede cambiar este puerto por otro si su máquina tiene este puerto ocupado.

VITE_BACKEND_URL se utiliza para especificar la URL del servidor backend. En este caso, el servidor backend está corriendo en <http://localhost:4000>, debe ser cambiado a la dirección IP de su máquina.

NINGÚN DATO DE VARIABLE DE ENTORNO PUEDE ESTAR VACÍO.

b. Variables Backend (Necesarias)

```
# DB
DB_NAME="NOMBRE_PRUEBA"
DB_USER="USUARIO_PRUEBA"
DB_PASSWORD="CLAVE_PRUEBA"
DB_HOST="iniciativaser-db_test-1"
DB_PORT=3308
DB_DRIVER="mariadb"

#JWT
JWT_SECRET="SECRETO_PRUEBA"

# configuración cors
ORIGIN="http://localhost:3000"

# configuración server
IP="0.0.0.0"
PORT=4000
```

Las primeras seis variables (DB_NAME, DB_USER, DB_PASSWORD, DB_HOST, DB_PORT, DB_DRIVER) están relacionadas con la configuración de la base de datos. Estas variables se utilizan para definir el nombre de la base de datos (DB_NAME), el usuario (DB_USER), la contraseña (DB_PASSWORD), el host (DB_HOST), el puerto (DB_PORT) y el controlador de la base de datos (DB_DRIVER) que tu aplicación utilizará para conectarse a la base de datos.

La variable JWT_SECRET es utilizada para firmar y verificar los tokens JWT en tu aplicación. JWT significa "JSON Web Token", que es un estándar de la industria para la creación de tokens de acceso que permiten la autenticación de usuarios y la transmisión segura de información.

La variable ORIGIN se utiliza para configurar el CORS (Cross-Origin Resource Sharing) en tu aplicación. CORS es una política de seguridad que permite o restringe las solicitudes de recursos a tu servidor desde un origen diferente.

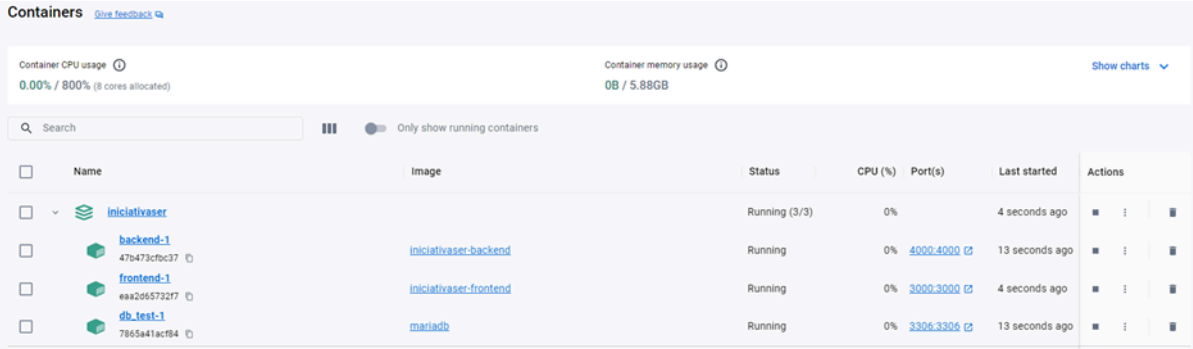
Finalmente, las variables IP y PORT se utilizan para configurar la dirección IP y el puerto en el que tu servidor estará escuchando las solicitudes entrantes.

NINGÚN DATO DE VARIABLE DE ENTORNO PUEDE ESTAR VACÍO.

2. Para ejecutar el proyecto, primero ejecutamos o abrimos la aplicación Docker Desktop. Para que la página se suba en la aplicación, es necesario ejecutar el comando `"docker-compose up"` o `"docker compose up"` en la carpeta del proyecto.

```
IniciativaSer> docker-compose up
```

Una vez levantado el proyecto, en Docker Desktop debería verse los siguientes elementos:



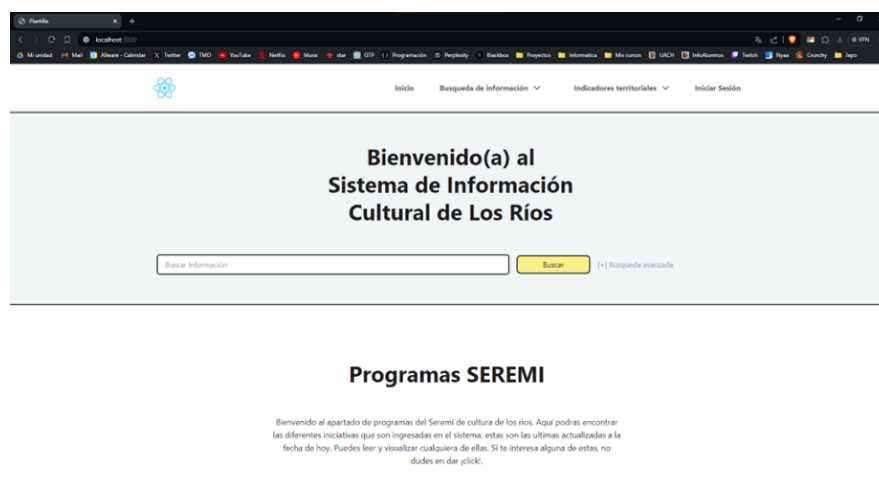
| Name | Image | Status | CPU (%) | Port(s) | Last started | Actions |
|----------------------------|------------------------|---------------|---------|-----------|----------------|---------|
| iniciativaser | | Running (3/3) | 0% | | 4 seconds ago | |
| backend-1 47b473cfc37 | iniciativaser:backend | Running | 0% | 4000:4000 | 13 seconds ago | |
| frontend-1 eaa2d55732f7 | iniciativaser:frontend | Running | 0% | 3000:3000 | 4 seconds ago | |
| db_test-1 7865a41act94 | mysql:8.0 | Running | 0% | 3306:3306 | 13 seconds ago | |

Esto significa que las imágenes se subieron correctamente y están en ejecución.

O si no tiene docker desktop puede verificar con el siguiente mensaje final en Visual Studio code o en la terminal en la que haya ejecutado el comando `"docker compose up"`

```
backend-1 | Executing (default): SHOW INDEX FROM `programa_iniciativa`
backend-1 | Executing (default): SELECT TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_TYPE = 'BASE TABLE' AND TABLE_NAME = 'iniciativa_comuna' AND TABLE_SCHEMA = 'test'
backend-1 | Executing (default): CREATE TABLE IF NOT EXISTS `iniciativa_comuna` (`iniciativaId` INTEGER NOT NULL, `comunaId` INTEGER NOT NULL, `createdAt` DATETIME NOT NULL, `updatedAt` DATETIME NOT NULL, PRIMARY KEY (`iniciativaId`, `comunaId`), FOREIGN KEY (`iniciativaId`) REFERENCES `iniciativa` (`id`) ON DELETE NO ACTION ON UPDATE CASCADE, FOREIGN KEY (`comunaId`) REFERENCES `comuna` (`id`) ON DELETE NO ACTION ON UPDATE CASCADE) ENGINE=InnoDB;
backend-1 | Executing (default): SHOW INDEX FROM `iniciativa_comuna`
backend-1 | Server is running on http://0.0.0.0:4000
```

Finalmente, el proyecto se ejecutará en la ruta local definida `http://localhost:3000/`



Observaciones:

- El manual de despliegue no enseña como configurar los puertos de la aplicación, sino más bien como desplegar la aplicación.
- Esta aplicación correrá en los puertos **4000**, **3000** y **3306**, por ende, si no están disponibles no podrá ejecutar el sistema, y deberá pedir un manual de configuración de despliegue.