



Memoria SAD

Descripción del proyecto

Especificación del servicio

Perspectiva general

Frontend

Implementación

Worker

Cola NATS

Limitaciones del trabajo

Dependencias Requeridas

Despliegue del servicio

Tests

Despliegue

Ejemplo login

Ejemplo asignación de tareas

Ejemplo peticiones múltiples

Estado de NATS

Anexos

Capturas Google Cloud Console

Capturas contenedor

Miembros del equipo de trabajo

Aceros Guerrero, Brayan Nicolás

Bengui Simao, Miguel

Páez, José Gregorio

Yépez Salinas, Nevil Rafael

Descripción del proyecto

El proyecto actual describe la implementación de la solución para un ejercicio planteado en la asignatura de Servicios y Aplicaciones Distribuidas (SAD), impartida en el curso 2023-2024 en el Máster Universitario de Ingeniería Informática de la Universidad Politécnica de Valencia.

Especificación del servicio

Perspectiva general

Construcción de microservicios para gestionar peticiones de clientes con tres componentes básicos:

- FRONTEND: encargado de recibir las peticiones REST de los clientes y colocarlas en una cola.
- WORKER: encargado de extraer una petición de la cola, ejecutar el trabajo asociado y volcar el resultado en otra cola.
- COLA: estructura que permite almacenar las peticiones para su posterior procesamiento.

Frontend

Componente que protege el servicio a través de peticiones autenticadas. Para realizar esta autenticación utiliza la API de Google OAuth2.

Consta de dos canales:

- Un punto de entrada, implementado con el express para manejar peticiones REST
- Una cola implementada con NATS

Implementación

- Servidor REST protegido por un portador de token, que asegura la identidad del que solicita el servicio. En el proyecto actual se ha configurado la API de Google para la autorización en Google API Console obteniendo un token de acceso del servidor de autorización de Google.
- API:

- Se identifican los envíos y recepciones de los trabajos mediante un id único haciendo uso de la librería `crypto`
- Se crea un enlace para acceder al resultado de dicho trabajo
 - En caso de situarse en procesamiento, se informa y se espera al reenvío de la petición.

Worker

Servicio que atiende las peticiones del FRONTEND y ejecuta los trabajos asignados. Recibe una petición que consta de una serie de campos que describen el trabajo:

- `URL:` que describe el URL del repositorio
- `PATH:` que describe el directorio donde se sitúa la función principal del programa
- `FILE:` nombre del fichero .js o .py
- `ARGUMENTS:` que describe el URL del repositorio

Además realiza las siguientes funciones:

- `clearEnviroment` : elimina los archivos y directorios utilizados durante el proceso de la petición para dejar el entorno como estaba antes de la petición.
- `downloadRepo` : descarga un repositorio Git en la carpeta "dir". Antes de la descarga, verifica si la petición tiene un token y ajusta la URL en consecuencia. Después de la descarga, llama a `codeExecution` para ejecutar el código según los parámetros especificados en la petición.
- `codeExecution` : ejecuta el código según los parámetros especificados en la petición. Admite archivos con extensiones ".js" y ".py". Instala las dependencias necesarias antes de ejecutar el código.
- `consume` : Establece una conexión NATS, se suscribe al tema "petition-topic" y consume mensajes. Cuando se recibe un mensaje, se procesa la petición, se descarga el repositorio y se ejecuta el código.
- `sendOutput` : Publica el resultado de la petición en el tema "result-topic" de NATS. El resultado incluye la clave y el resultado de la petición.

Para intercambiar los resultados de las peticiones de trabajo se ha optado por la creación de ficheros con el mismo nombre que el identificador único de la petición, para facilitar su identificación por parte del cliente. También se ha definido un temporizador para iniciar la eliminación de ese fichero transcurridos 30 minutos.

Cola NATS

Tecnología utilizada para establecer comunicación entre los servicios. Para poder garantizar que se produce un reparto de los trabajos entre diferentes workers, es necesario definir una cola de suscripción al grupo `'workers'` .

Se define una conexión en el FRONTEND y en cada WORKER con el servidor NATS a través de la definición de dos temas (subjects) `PET-TOPIC` Y `RESULT-TOPIC` .

- El frontend publica las peticiones bajo el tema pet-topic y se suscribe al tema pet-topic.
- Cada worker se suscribe a la cola workers del tema `pet-topic` y recibe las peticiones de los clientes.
- A su vez, cada worker publica los resultados bajo el tema result-topic.

Limitaciones del trabajo

- El servicio no es capaz de retomar los trabajos ejecutados por workers que dejan de ejecutarse en los contenedores.
- No existe una lista de trabajos suscritos, pendientes y finalizados.
- No se ha definido un cluster de NATS, que permite el reparto entre diferentes nodos y la disponibilidad del servicio.
- El servicio es capaz de ejecutar únicamente programas realizados en node.js.
- No existe una medida del tiempo de ejecución

Dependencias Requeridas

Para la creación del proyecto se necesitó instalar las siguientes dependencias

```
pip install nats.io
npm install nats@latest
npm install morgan --save
npm install morgan
npm install express-session --save
npm install express-session
npm install express
```

```
npm install nats --save
npm init
npm install shelljs --save
```

Despliegue del servicio

- Localización de las pruebas para la ejecución:

```
https://github.com/jpaez7/ProyectoFinal-SAD/Test.git
```

- Comandos para el despliegue del servicio:

```
docker-compose up
```

```
docker-compose up --scale worker=5
```

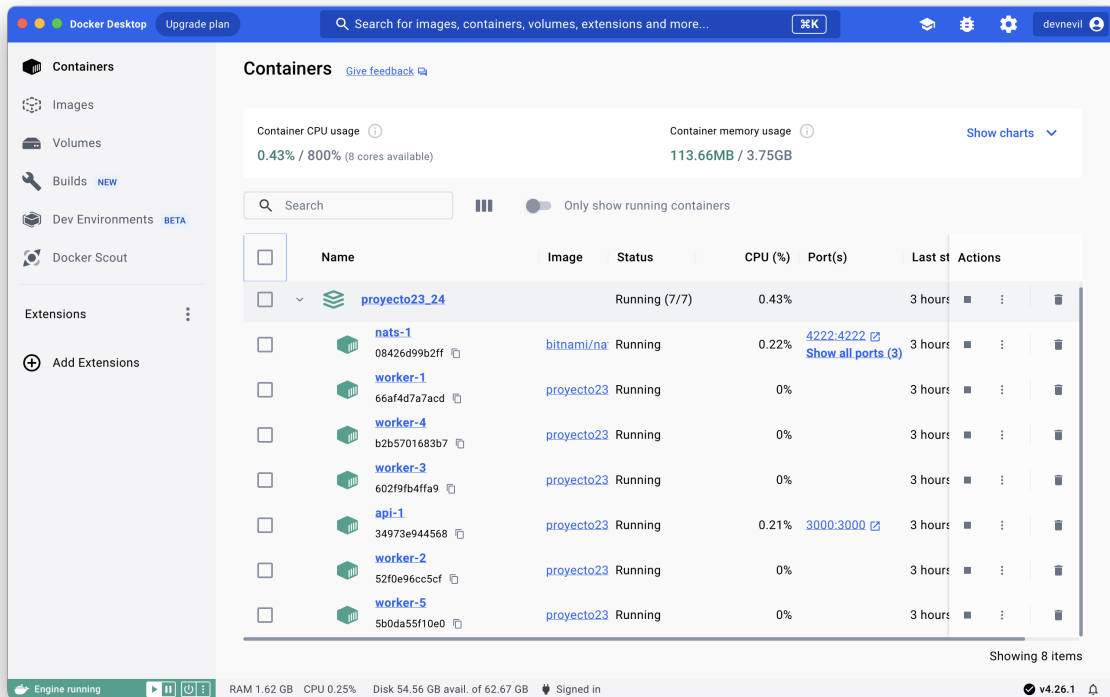
Tests

- El servidor que atiende las peticiones REST se encuentra en localhost:3000.
- Se muestra una pequeña interfaz para iniciar sesión haciendo uso de la cuenta de Google.
- Tras la autorización, se presenta la posibilidad de cerrar sesión. Tras esto, debe accederse nuevamente a localhost:3000
- Para realizar peticiones de trabajos se debe enviar las peticiones según el modelo siguiente:

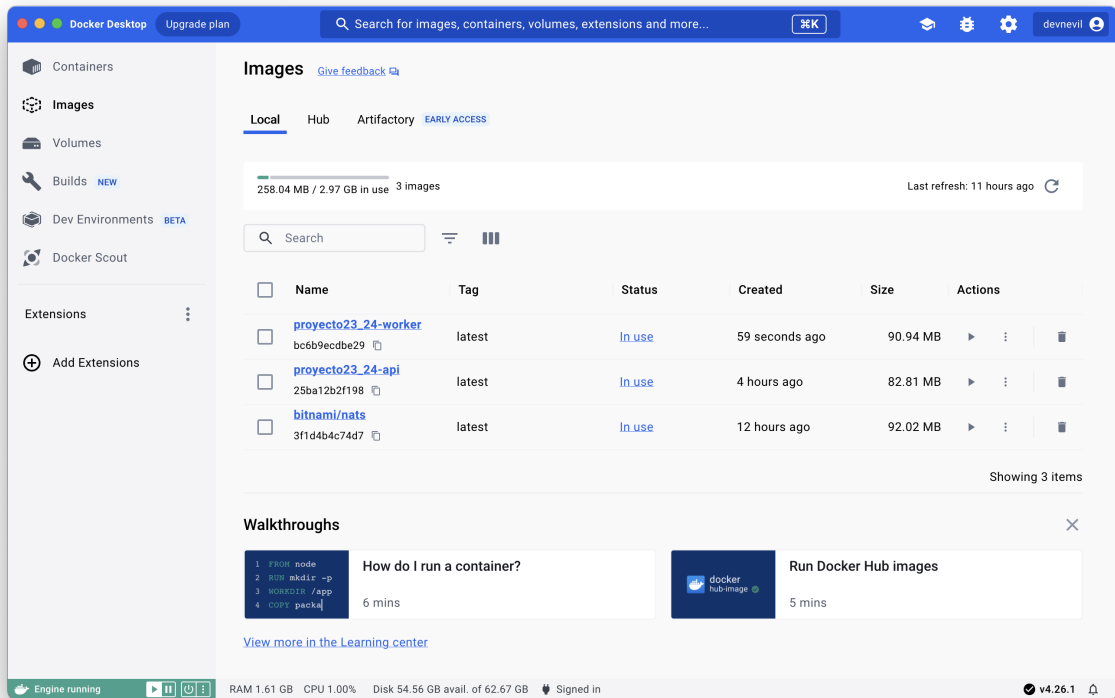
```
http://localhost:3000/petition?url=url&file=file&arguments=args&path=/
```

Despliegue

```
docker-compose up --scale worker=5
```

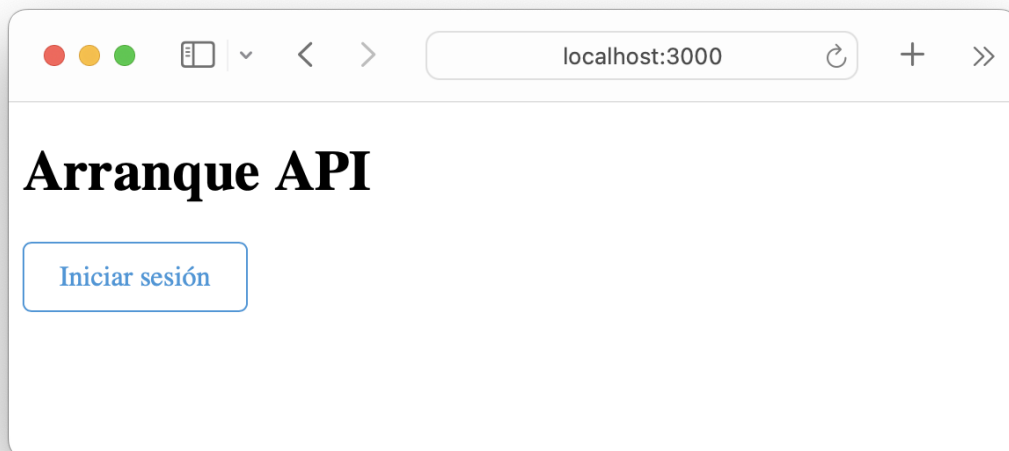


CONTENEDORES

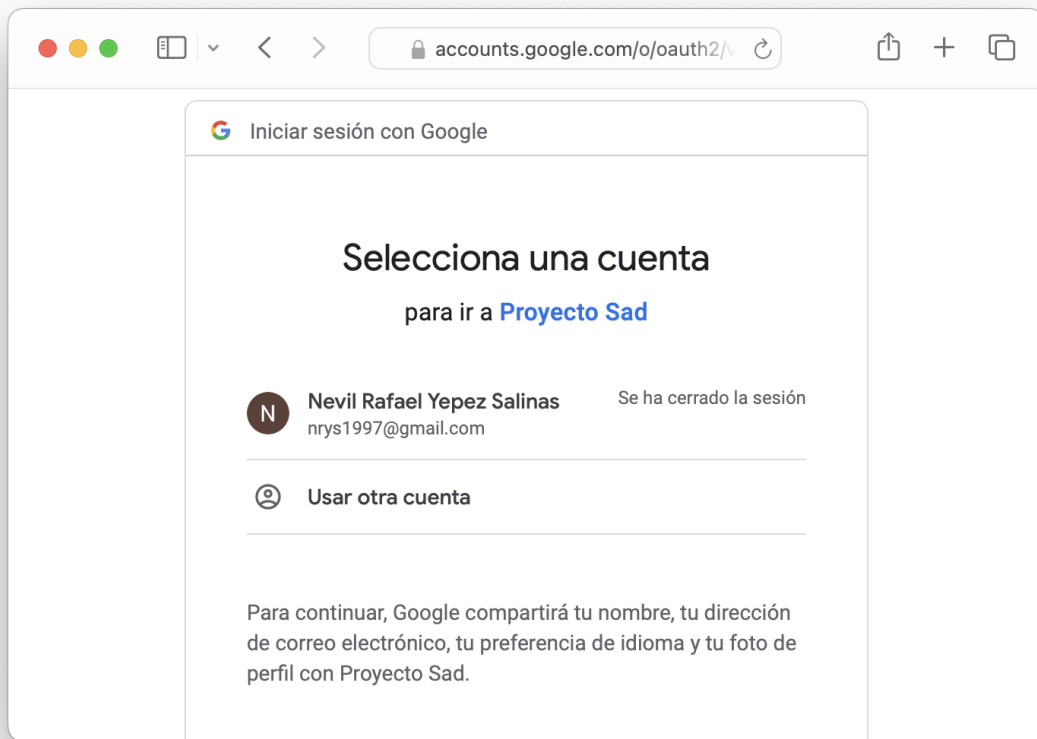


IMÁGENES

Ejemplo login



localhost:3000



Página de inicio de sesión con Google



Ejemplo asignación de tareas

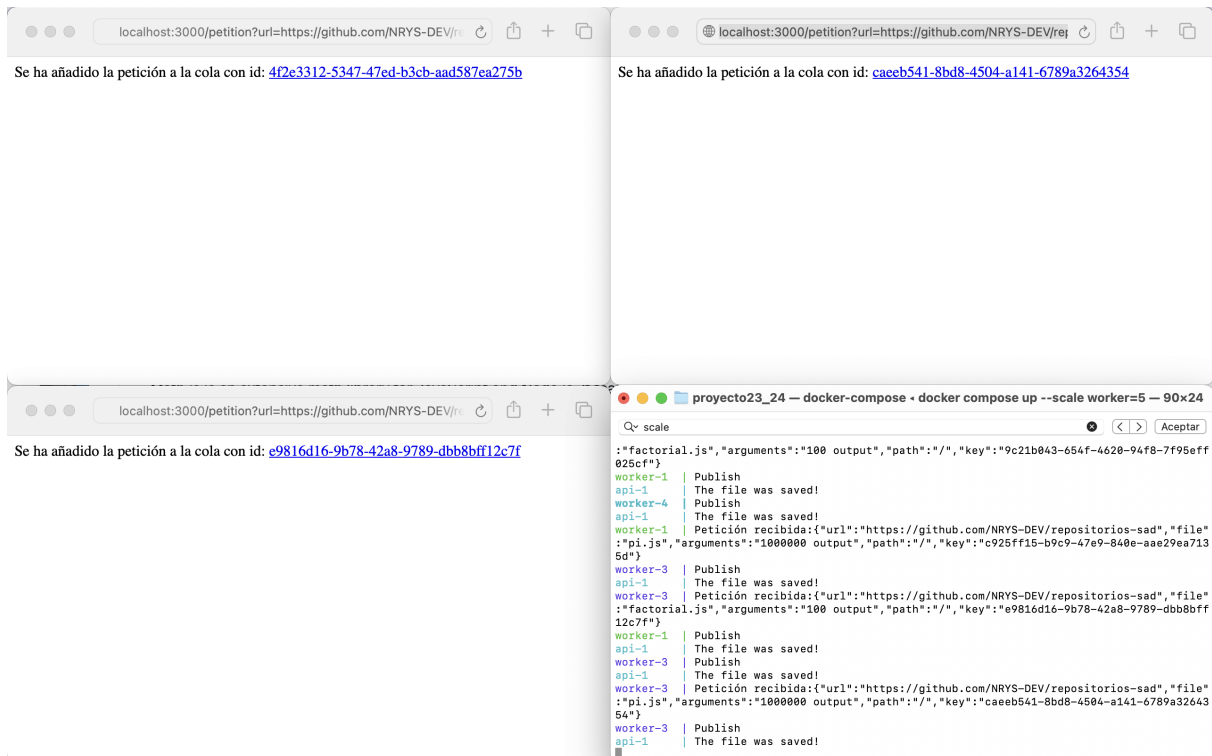


<http://localhost:3000/petition?url=https://github.com/NRYS-DEV/repositorios-sad&file=pi.js&arguments=1000000 output&path=/>



Resultado de la función tras realizar click al enlace del id del trabajo de la petición

Ejemplo peticiones múltiples



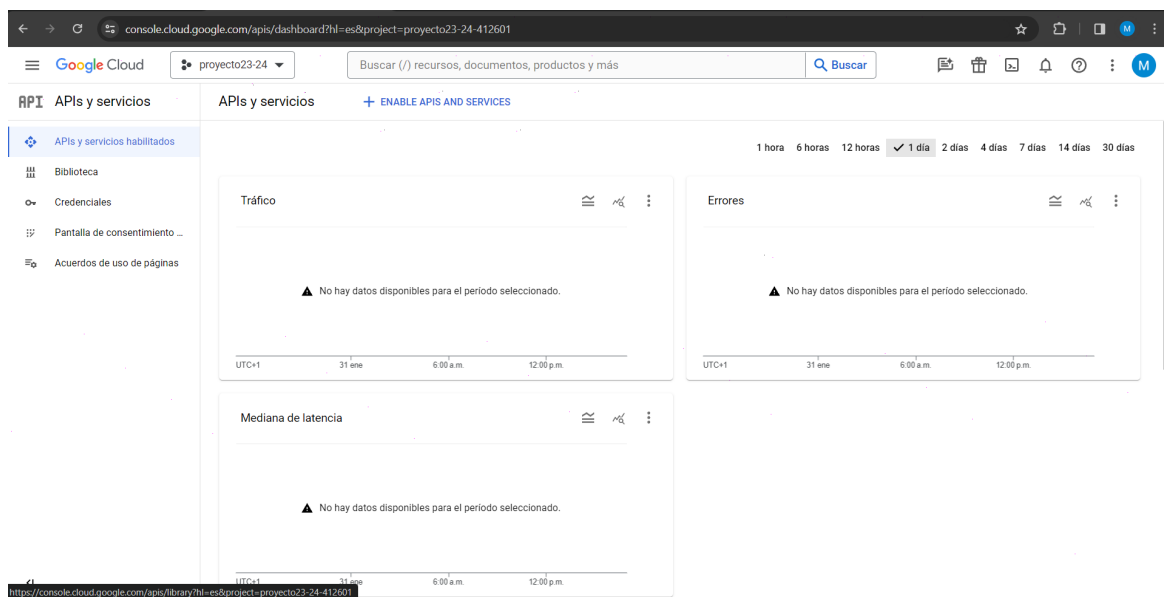
Los ejemplos propuestos se pueden acceder desde el siguiente repositorio

<https://github.com/NRYS-DEV/repositorios-sad/>

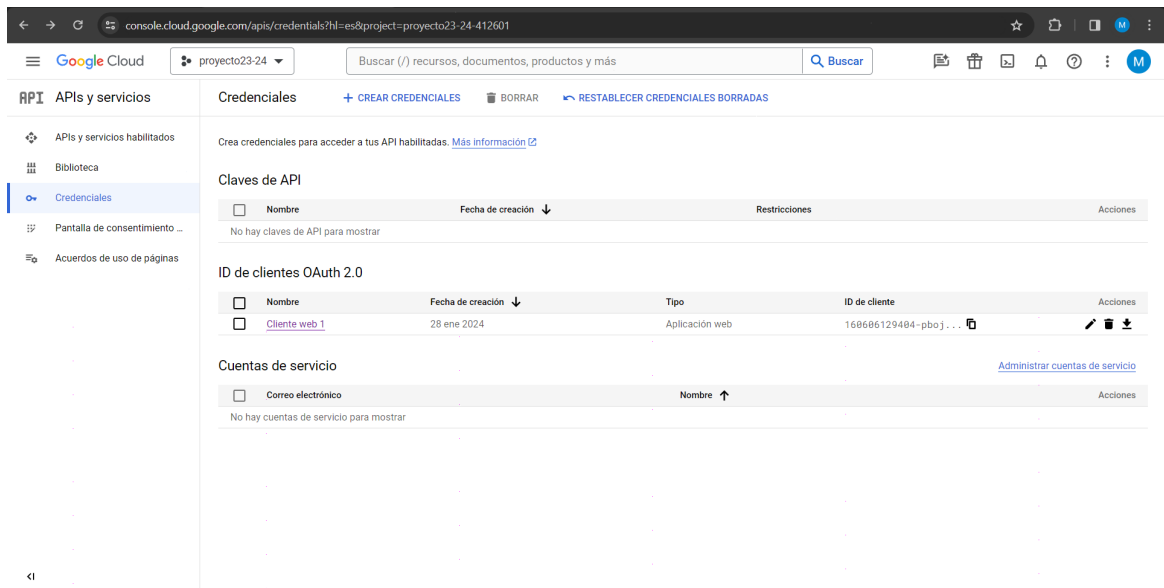
Anexos

▼ Capturas Google Cloud Console

Configuración del proyecto

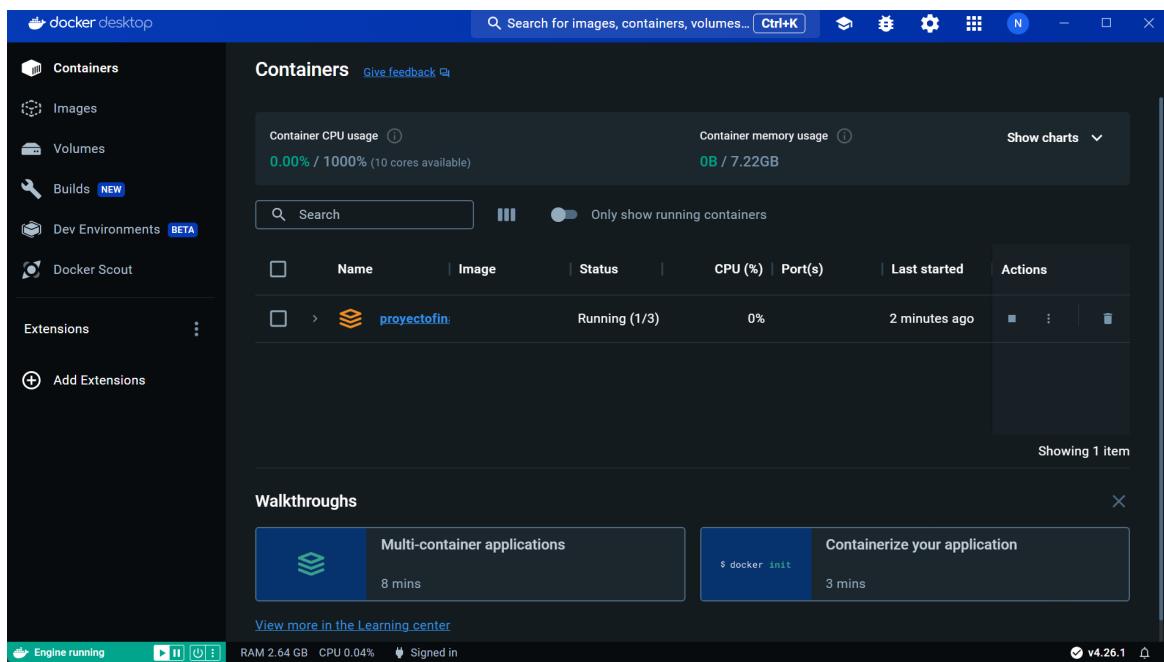


Configuración de la aplicación WEB



▼ Capturas contenedor

Contenedor docker



Imágenes docker

