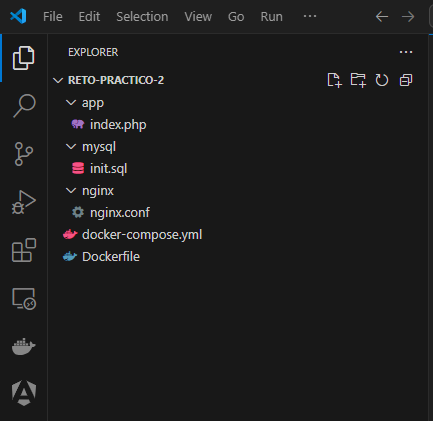
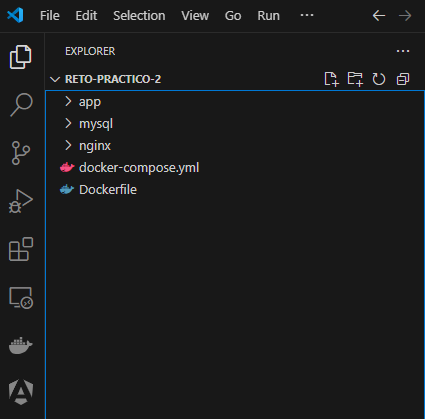
**Reto Practico 2: Hugo Ricardo Tello Pérez**

**Objetivo**: Implementar una arquitectura basada en contenedores utilizando Docker Compose, que incluya:

* Despliegue de una aplicación PHP.
* Persistencia de datos con MySQL.
* Balanceo de carga con Nginx.
* Monitoreo del estado de los servicios mediante healthchecks.

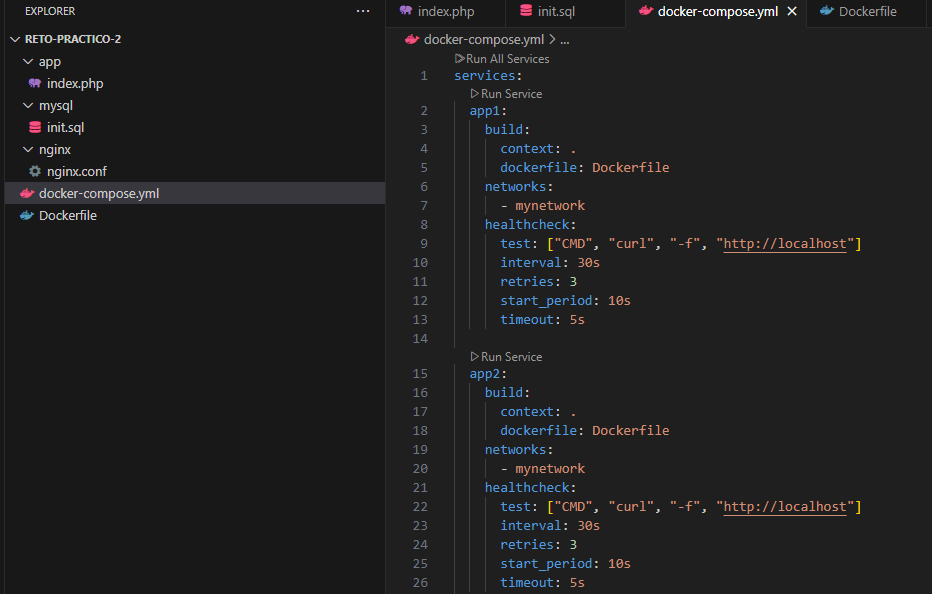
1. ***Preparar la aplicación web***

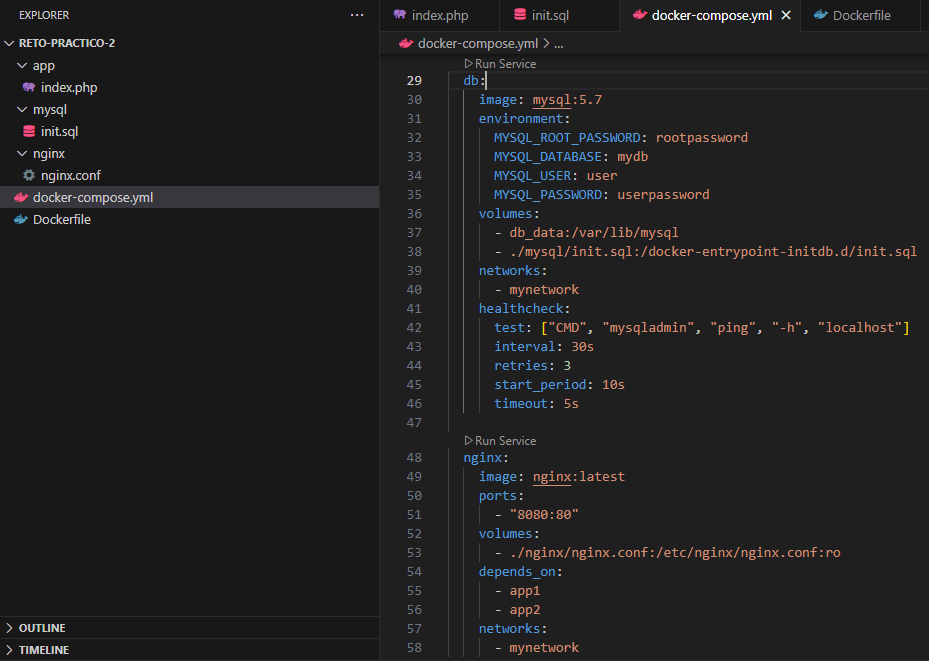
Estructura del proyecto

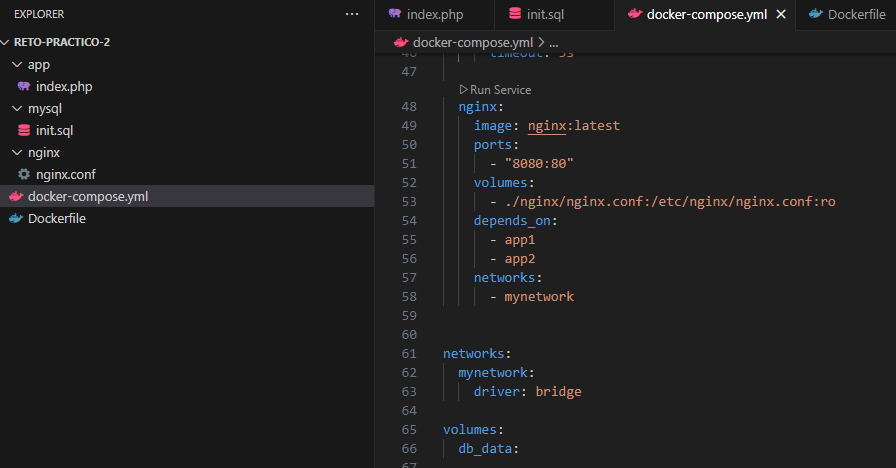


1. ***Crear un archivo docker-compose.yml que contenga los siguientes servicios:***
   * app

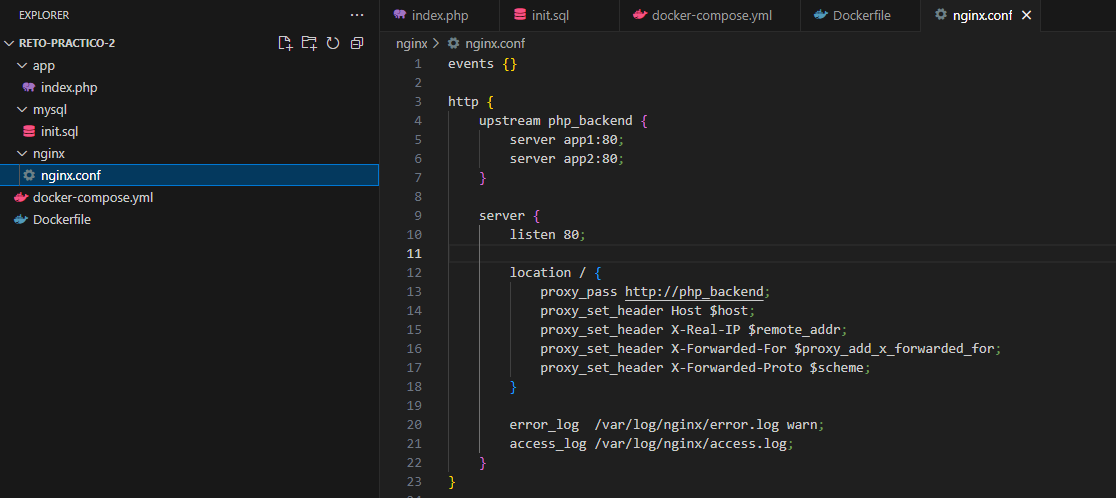
* Imagen base: php:apache
* Dos instancias para simular alta disponibilidad.
* Sirve el contenido de la carpeta app/.
* db
  + - Imagen: mysql:5.7
    - Configuración con usuario, contraseña y nombre de base de datos.
    - Volumen para persistencia de datos.
* nginx
  + - Actúa como balanceador de carga entre las dos instancias del servicio app.
    - Usa un archivo de configuración personalizado nginx.conf.
* Healthchecks
  + - Define verificaciones de estado para app y db en el archivo docker-compose.yml.





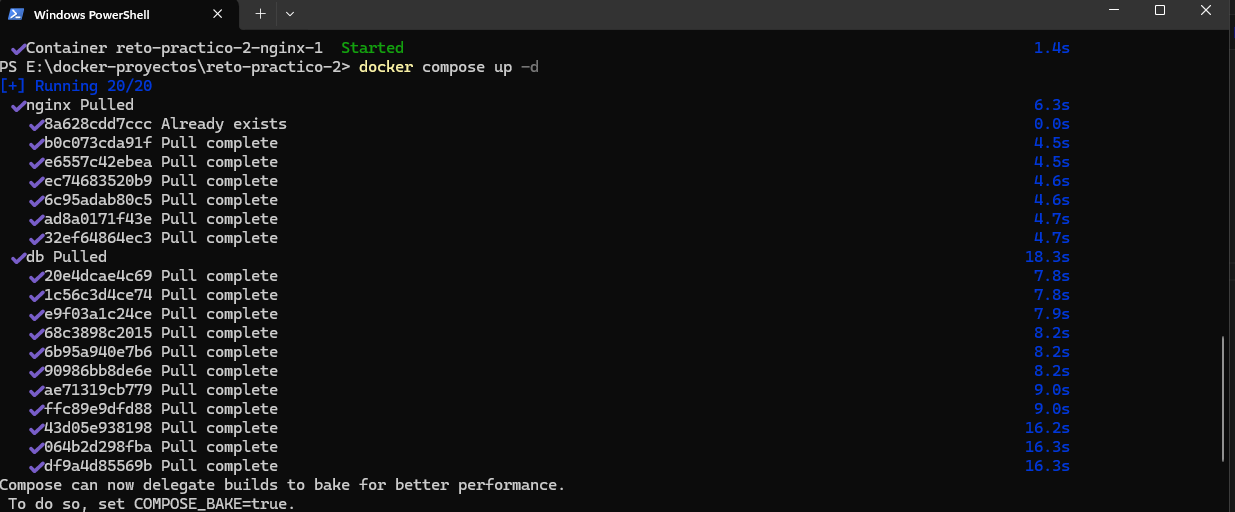


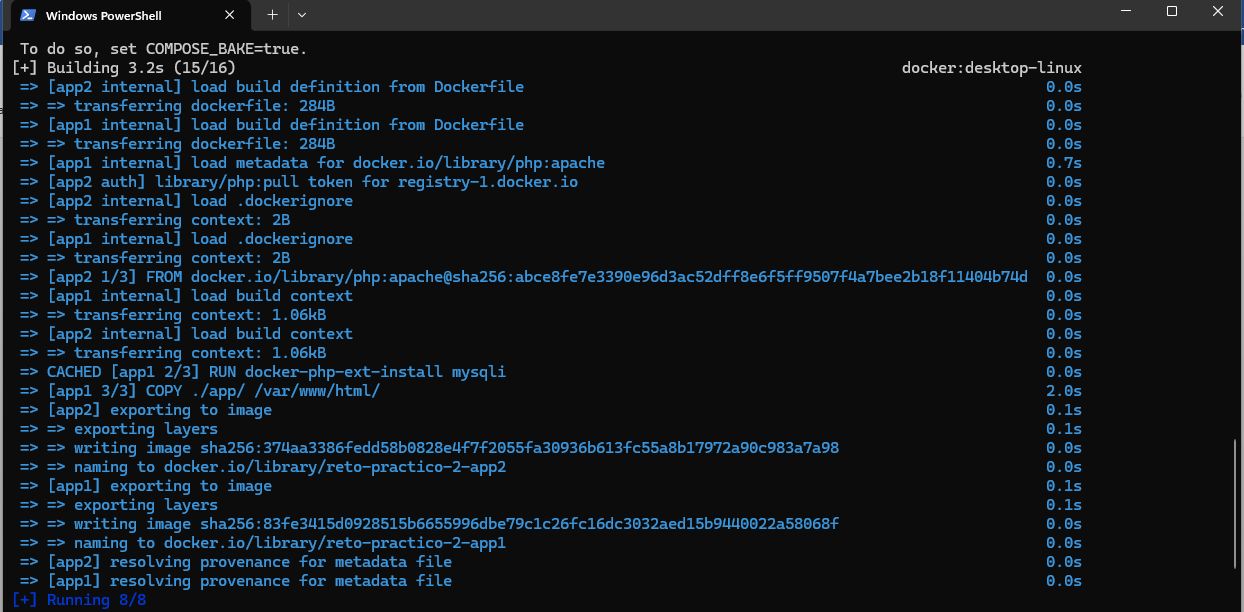
1. ***Configurar nginx.conf***
   * Debes configurar el archivo nginx.conf para distribuir el tráfico entrante entre ambas instancias de app.
   * Asegúrate de que cada petición se reparta de forma equilibrada.

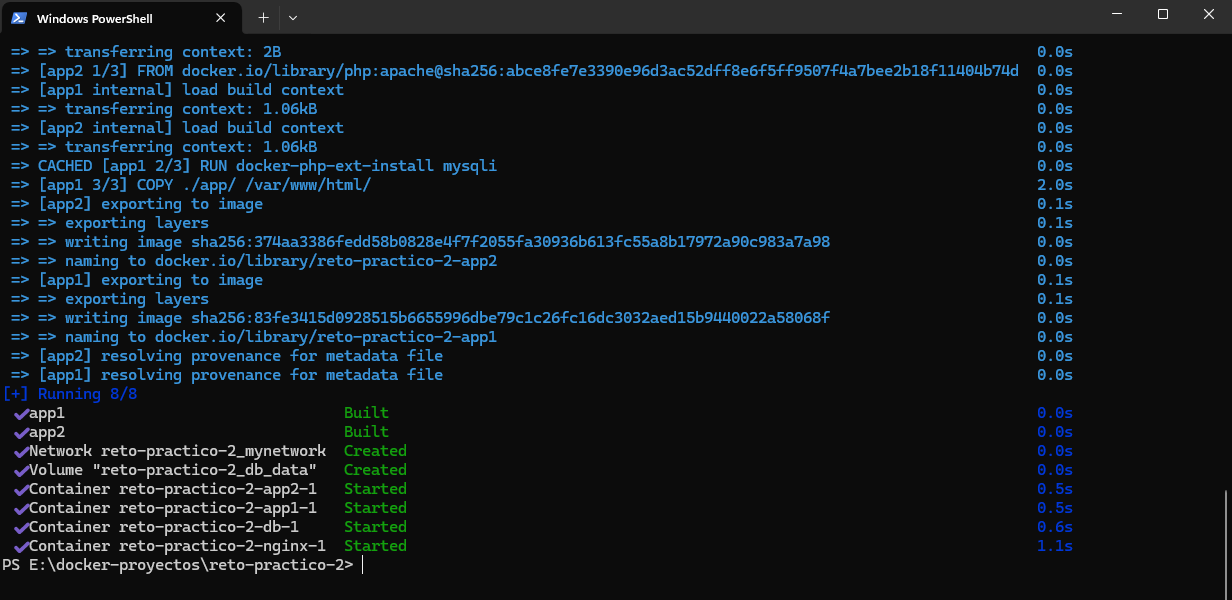


1. ***Desplegar y verificar el entorno***

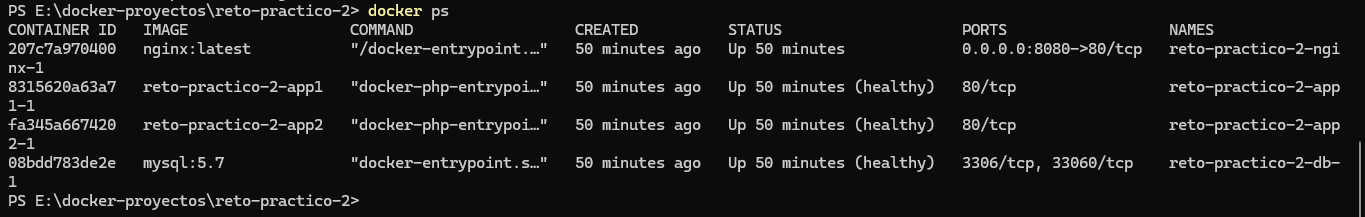
Ejecución del proyecto con el comando: docker compose up -d



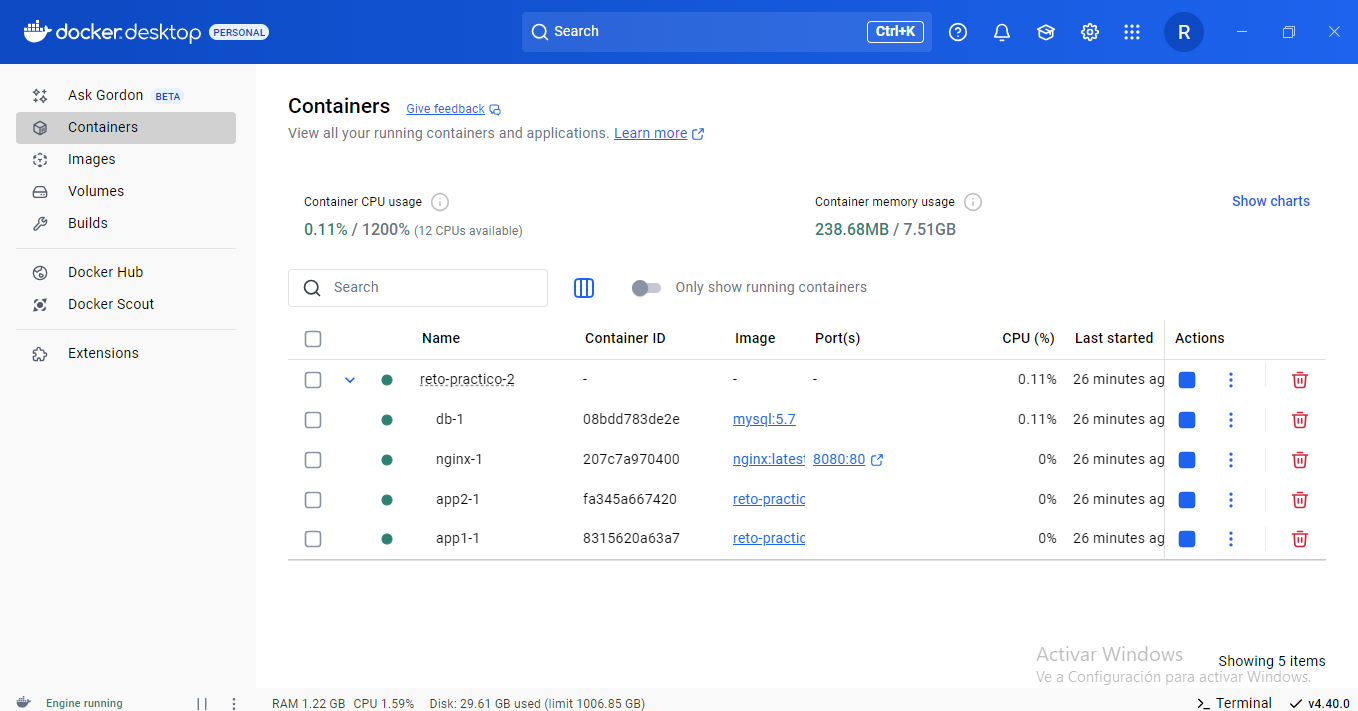




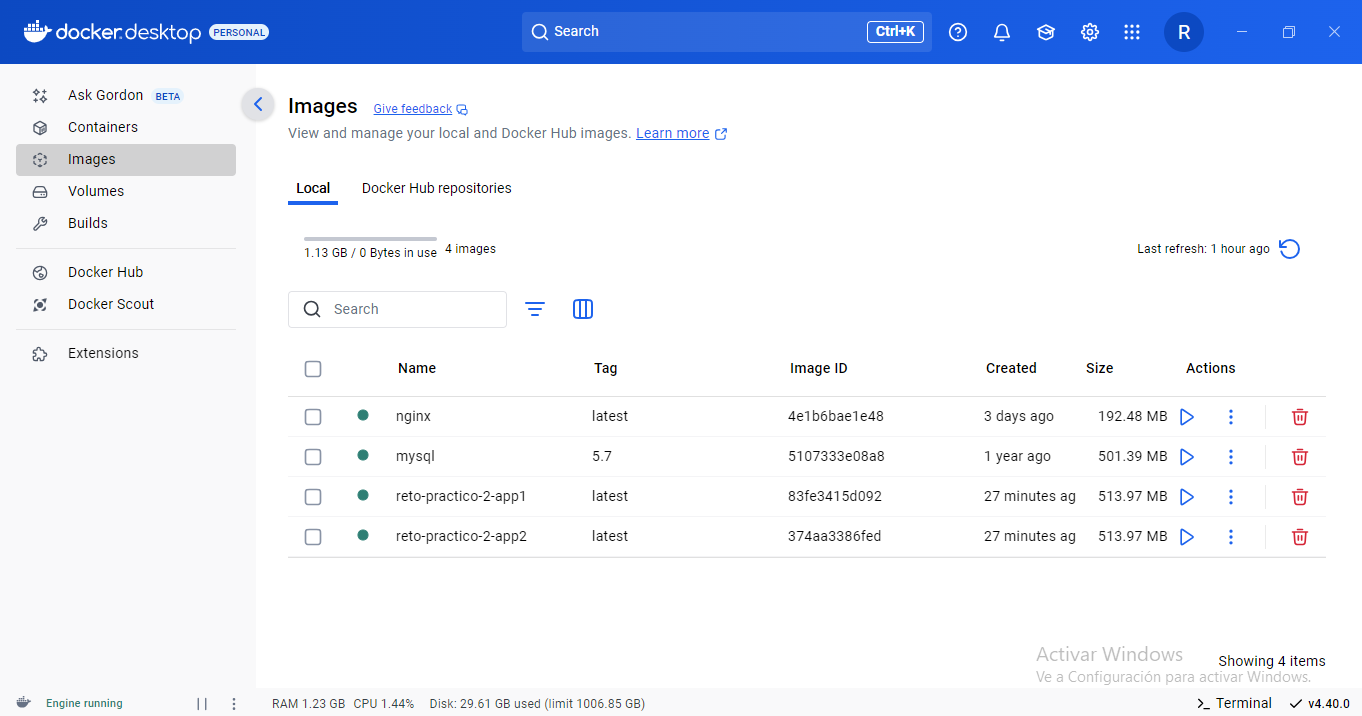
Listado de los contenedores creados: nginx, app1, app2, mysql



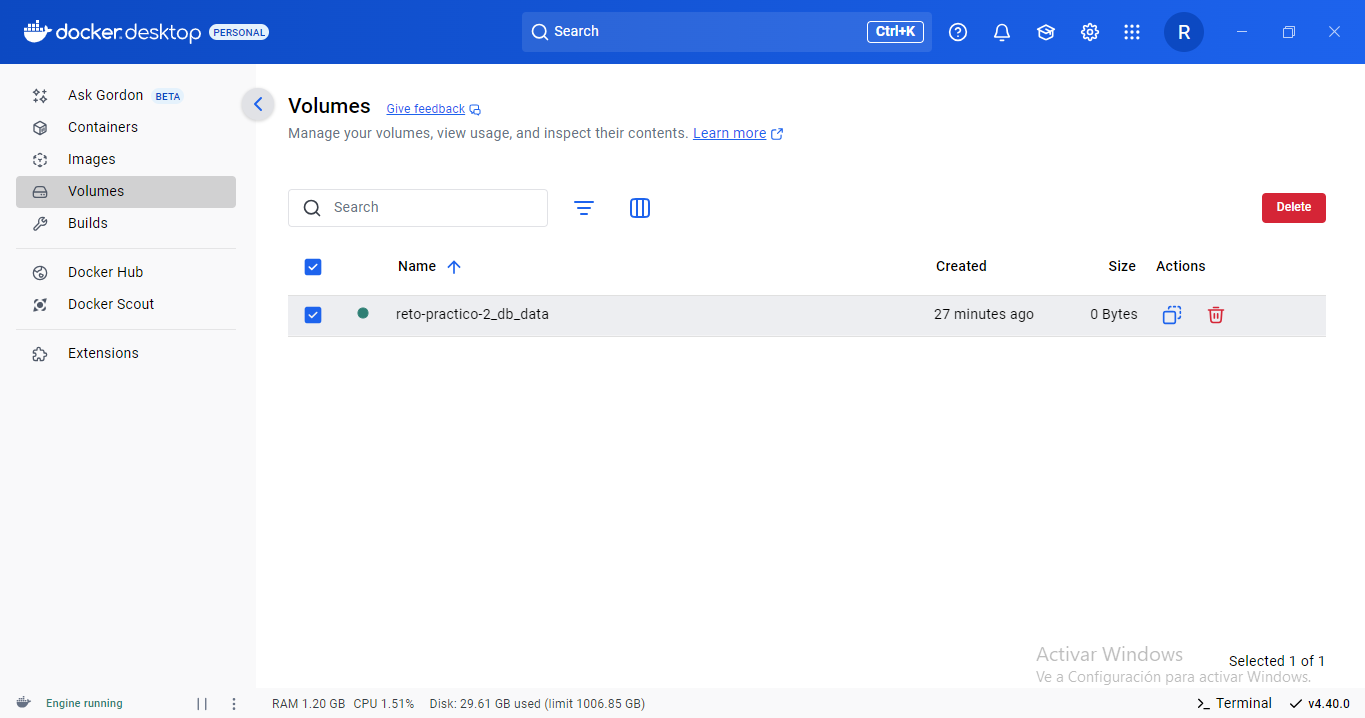
Docker desktop: Containers



Docker desktop: Images

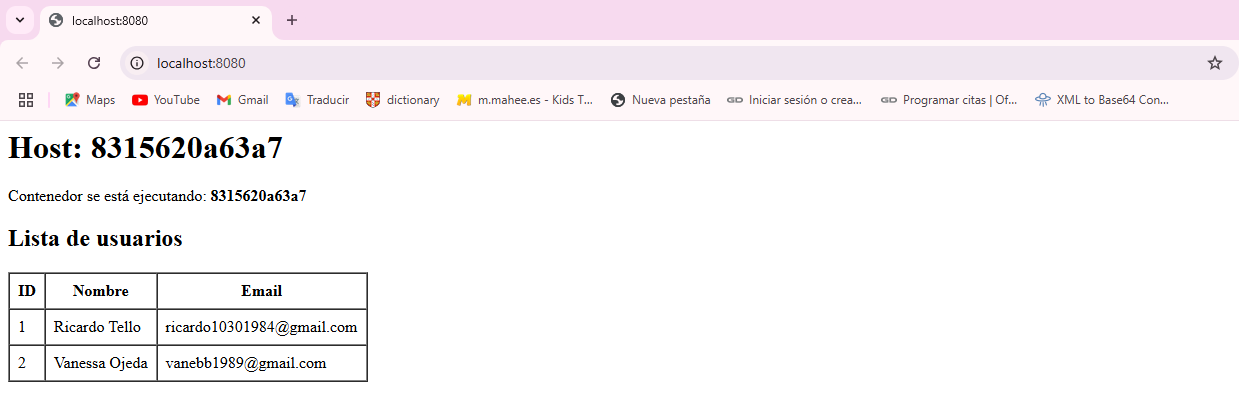


Docker desktop: Volumes

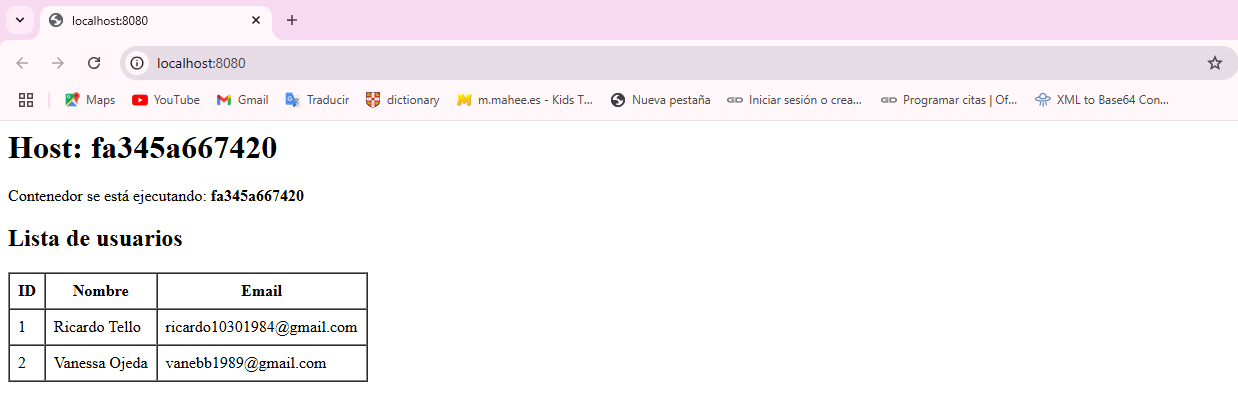


Ejecución de la aplicación en: <http://localhost:8080/>

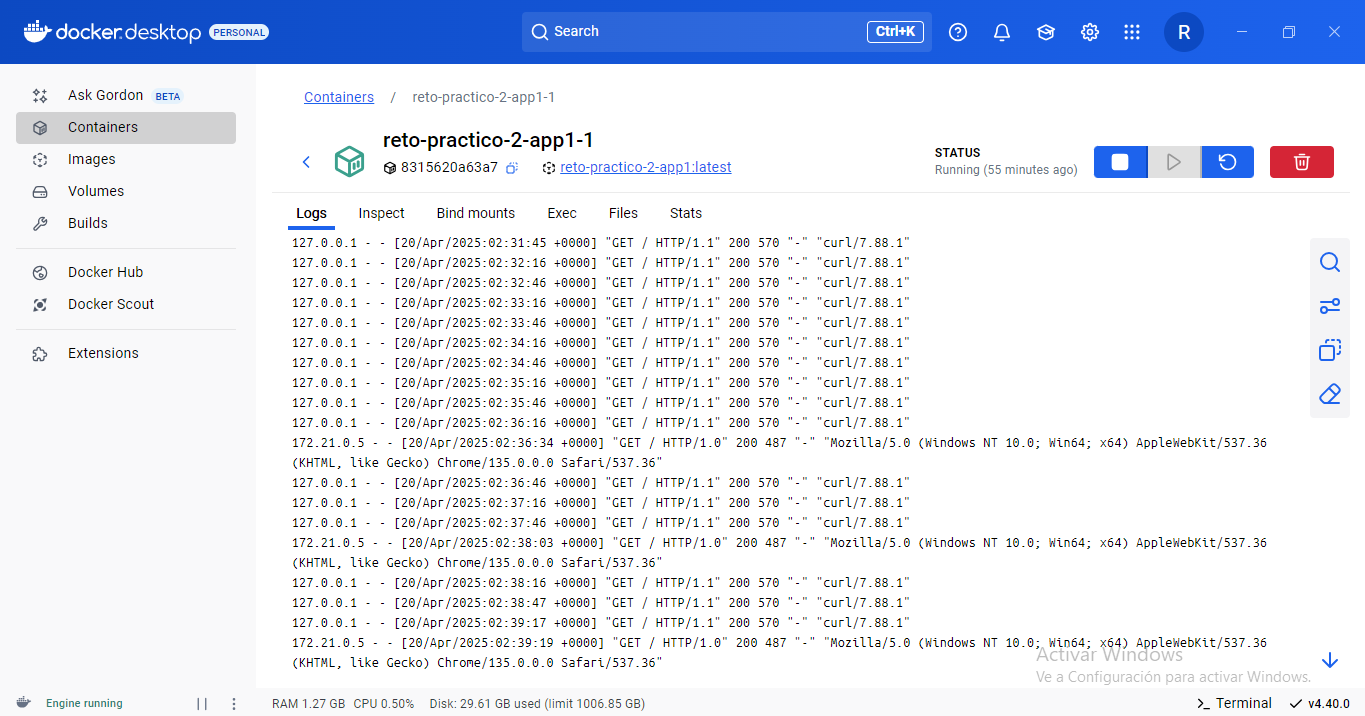
Contenedor app-1: 8315620a63a7

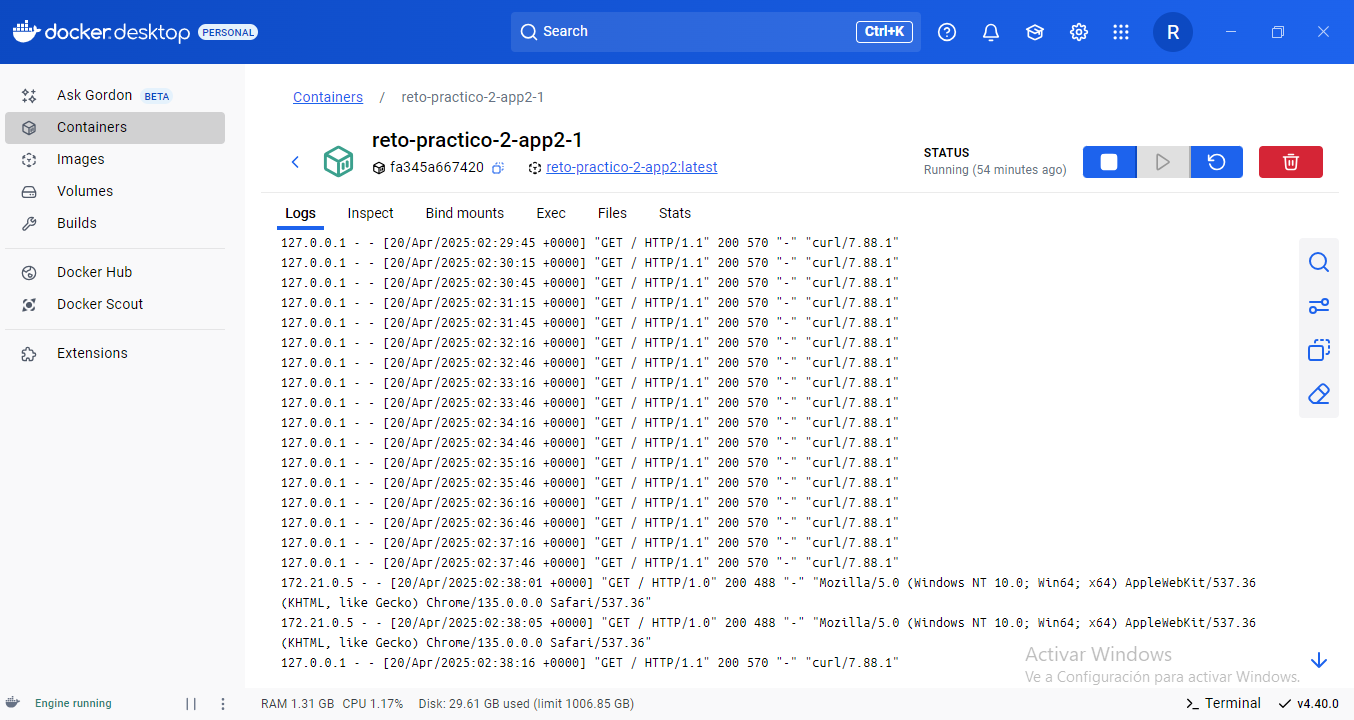


Contenedor app-2: fa345a667420

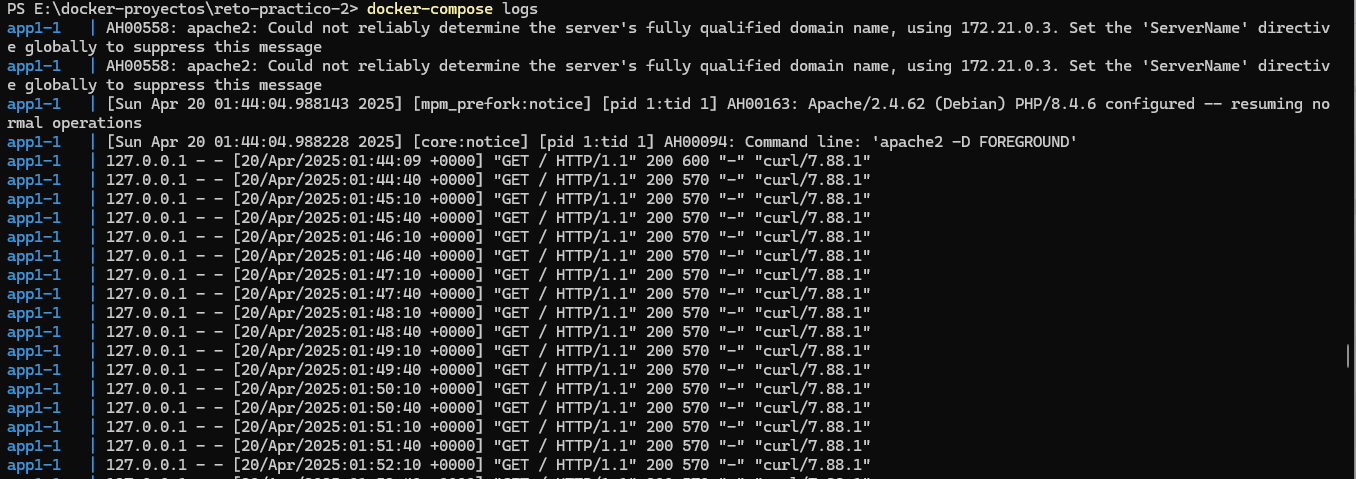


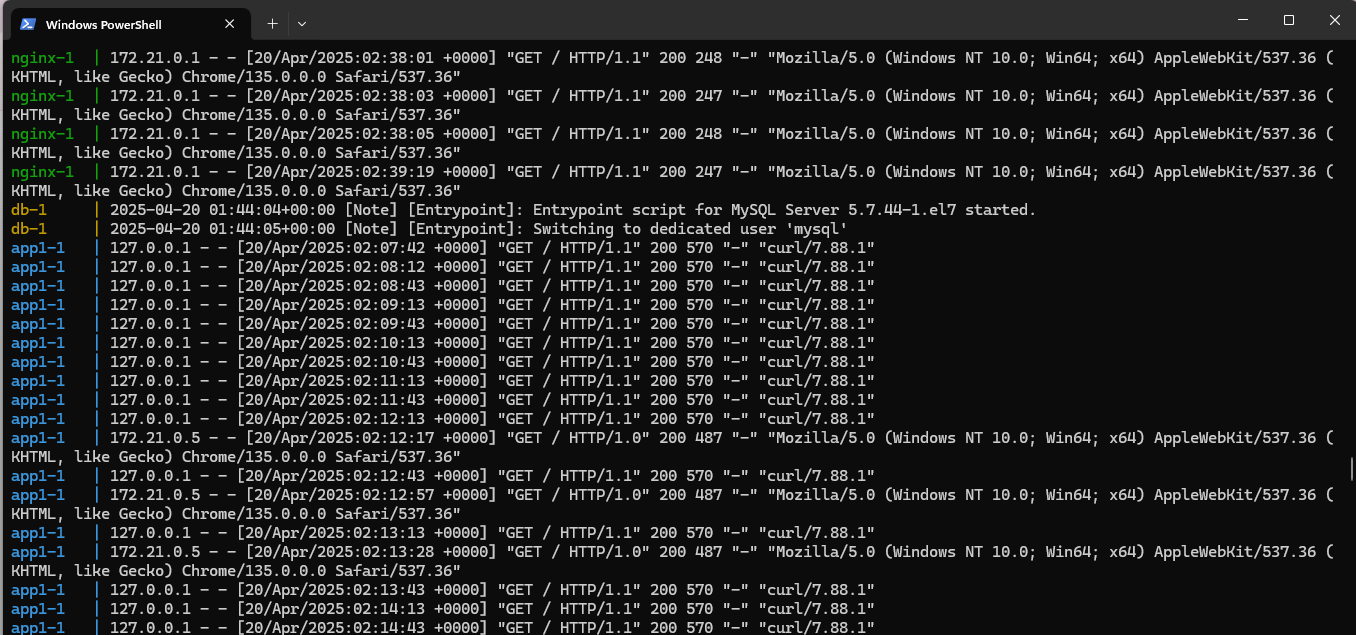
Logs de ejecución app1, app2: docker dsektop

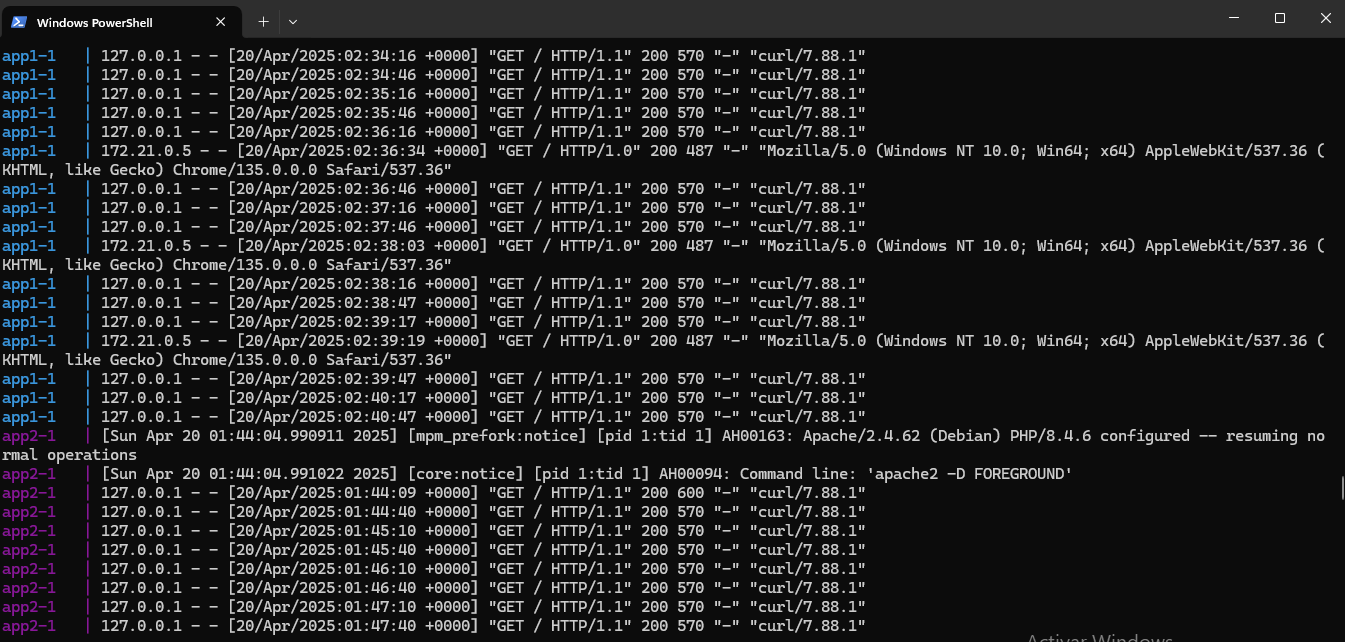


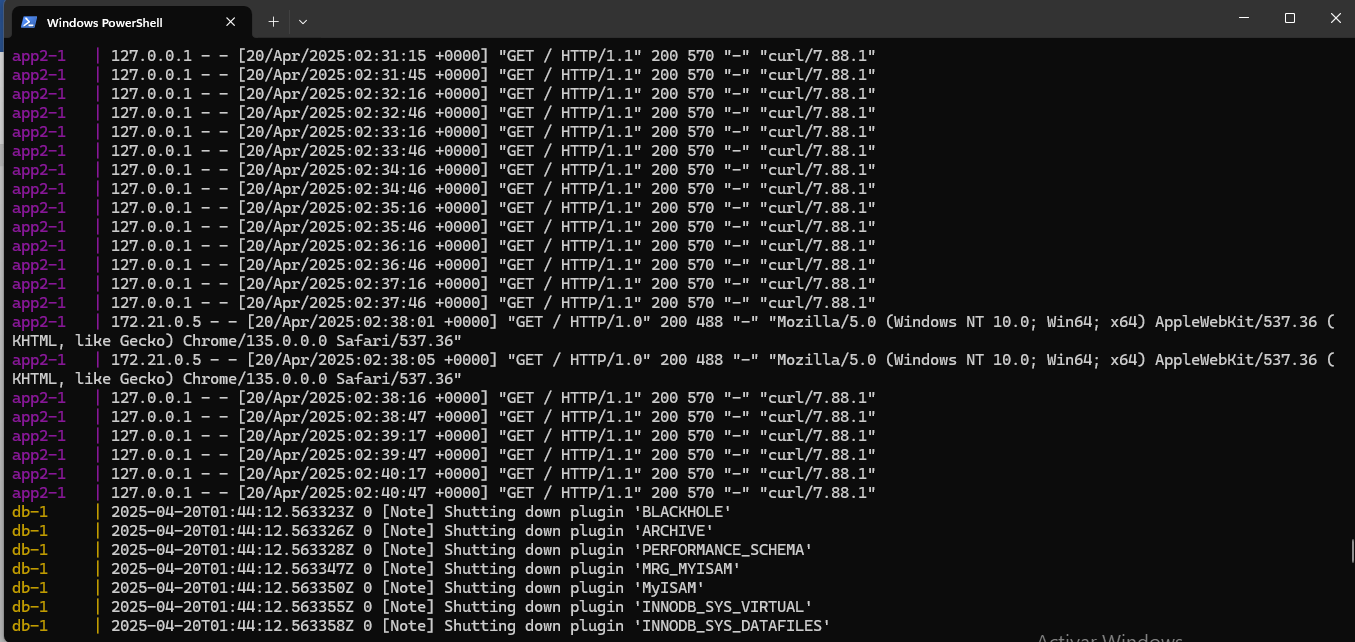


Logs de ejecución app1, app2: PowerShell

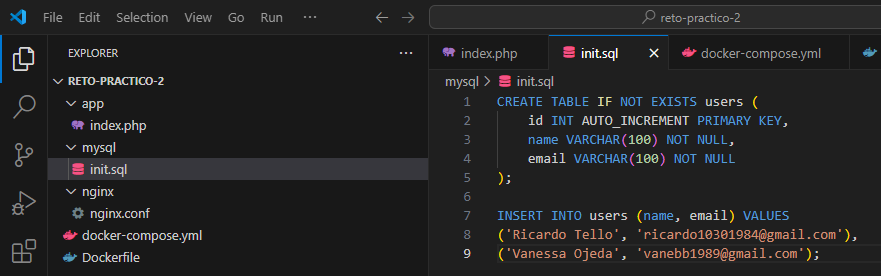


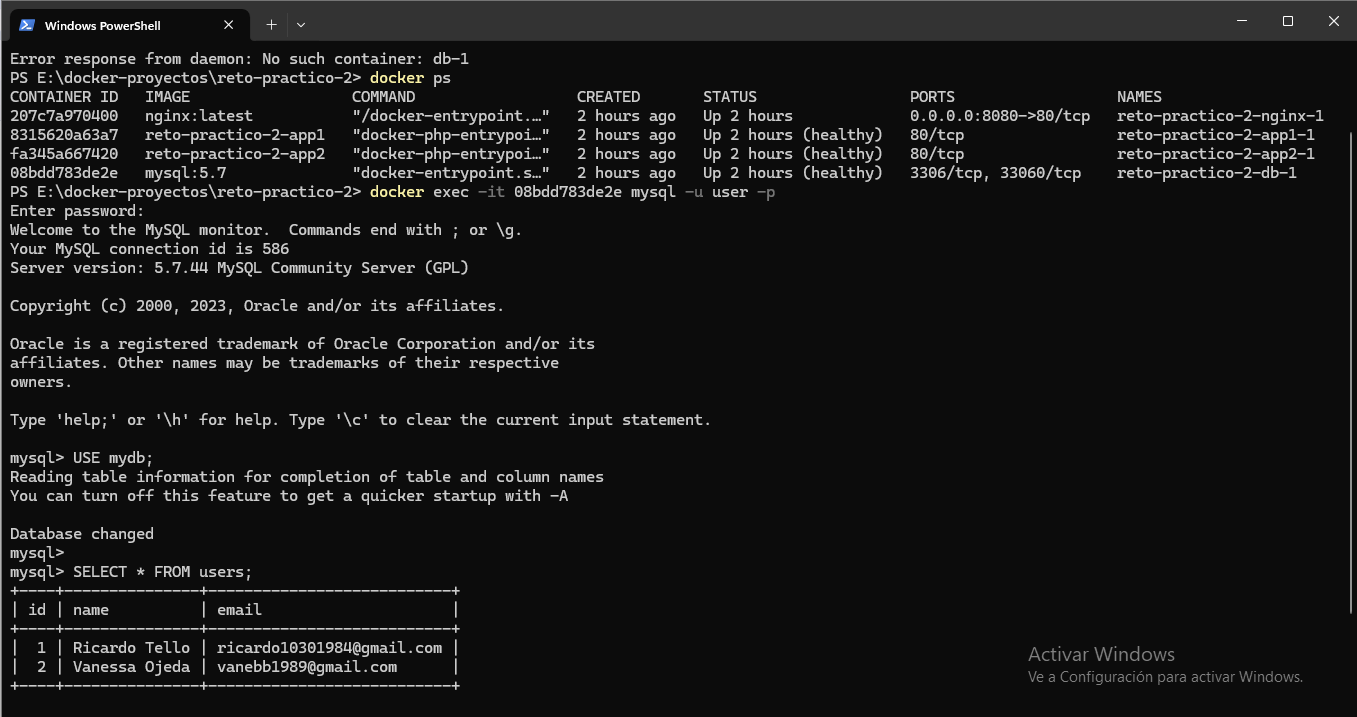




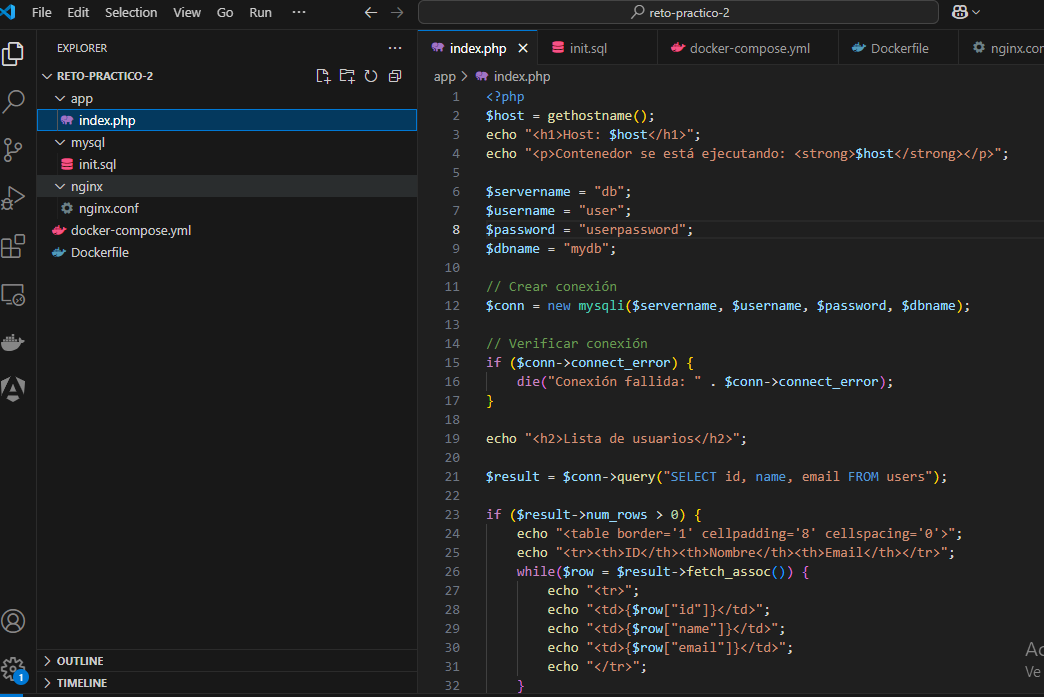


**Archivo init.sql:** se usa para ejecutar automáticamente comandos SQL cuando se inicia el contenedor de MySQL por primera vez.





**Archivo index.php:** es la página principal de la aplicación web, se valida o se prueba que todos los servicios de los contenedores están en funcionamiento.



**Comentarios:**

El objetivo logrado es haber implementado una arquitectura en contenedores utilizando Docker Compose, el mismo que ayuda a simular un entorno de producción en la vida real con la separación de servicios:

* PHP: aplicación web, alta disponibilidad con dos contenedores app1, app2
* MySQL: persistencia de datos, se cargan inicialmente desde el archivo init.sql
* Nginx: es el balanceador de carga que distribuye el trafico de las 2 instancias de la aplicación PHP
* Healthchecks: nos ayuda a verificar que la aplicación y la base de datos mysql estén en funcionamiento