UNIVERSIDAD TECNOLÓGICA DE SANTIAGO, UTESA.

Asignatura:

INF-025-001 ALGORITMOS PARALELOS

Tema:

Actividad Semana 3

Presentado por:

Georges de Jesús Gil Pichardo

Matricula:

1-18-2363

Presentado a:

M.A. IVAN MENDOZA

Docker y Docker Compose:

¿Qué es Docker?

Docker es una plataforma de código abierto que permite automatizar el despliegue de aplicaciones dentro de contenedores de software. Un contenedor es una unidad estándar que puede contener cualquier aplicación junto con sus dependencias y ejecutarla en cualquier entorno. Docker proporciona una forma de empaquetar aplicaciones en contenedores, lo que facilita su distribución y ejecución en diferentes entornos.

¿Para qué nos sirve en desarrollo?

En desarrollo, Docker es invaluable porque:

Portabilidad: Las aplicaciones empacadas en contenedores Docker son portátiles y pueden ejecutarse en cualquier lugar, desde el entorno de desarrollo hasta la producción, sin preocuparse por las diferencias en los entornos.

Aislamiento: Los contenedores proporcionan aislamiento, lo que significa que una aplicación y sus dependencias pueden ejecutarse sin interferir con otras aplicaciones en el mismo sistema.

Rapidez: Docker permite la creación rápida y la eliminación de contenedores, lo que facilita las pruebas y el desarrollo iterativo.

Escalabilidad: Docker facilita la creación y gestión de aplicaciones escalables mediante el uso de contenedores y orquestación de contenedores en clústeres.

¿Cómo se utiliza Docker?

Imágenes de Docker:

Una imagen de Docker es como una plantilla que contiene todo lo necesario para ejecutar una aplicación. Puedes buscar imágenes en el Docker Hub, que es un repositorio público de imágenes de Docker, o puedes crear tus propias imágenes personalizadas utilizando un

archivo de configuración llamado Dockerfile. Para descargar una imagen, utiliza el comando docker pull nombre_de_la_imagen. Para crear tu propia imagen, crea un Dockerfile que especifique las instrucciones para construir la imagen y utiliza el comando docker build -t nombre_de_la_imagen . en el mismo directorio que el Dockerfile.

Contenedores:

Un contenedor es una instancia en ejecución de una imagen de Docker. Puedes crear y ejecutar un contenedor a partir de una imagen utilizando el comando docker run nombre_de_la_imagen. Puedes especificar opciones adicionales, como puertos, volúmenes y variables de entorno, para personalizar el comportamiento del contenedor.

Comandos Básicos:

- docker pull nombre_de_la_imagen: Descargar una imagen de Docker desde Docker Hub.
- docker run nombre_de_la_imagen: Ejecutar un contenedor a partir de una imagen.
- docker ps -a: Listar todos los contenedores (incluso los detenidos).
- docker stop ID_del_contenedor: Detener un contenedor en ejecución.
- docker rm ID_del_contenedor: Eliminar un contenedor.
- docker images: Listar las imágenes de Docker en el sistema.
- docker rmi nombre_de_la_imagen: Eliminar una imagen de Docker.

Docker Compose:

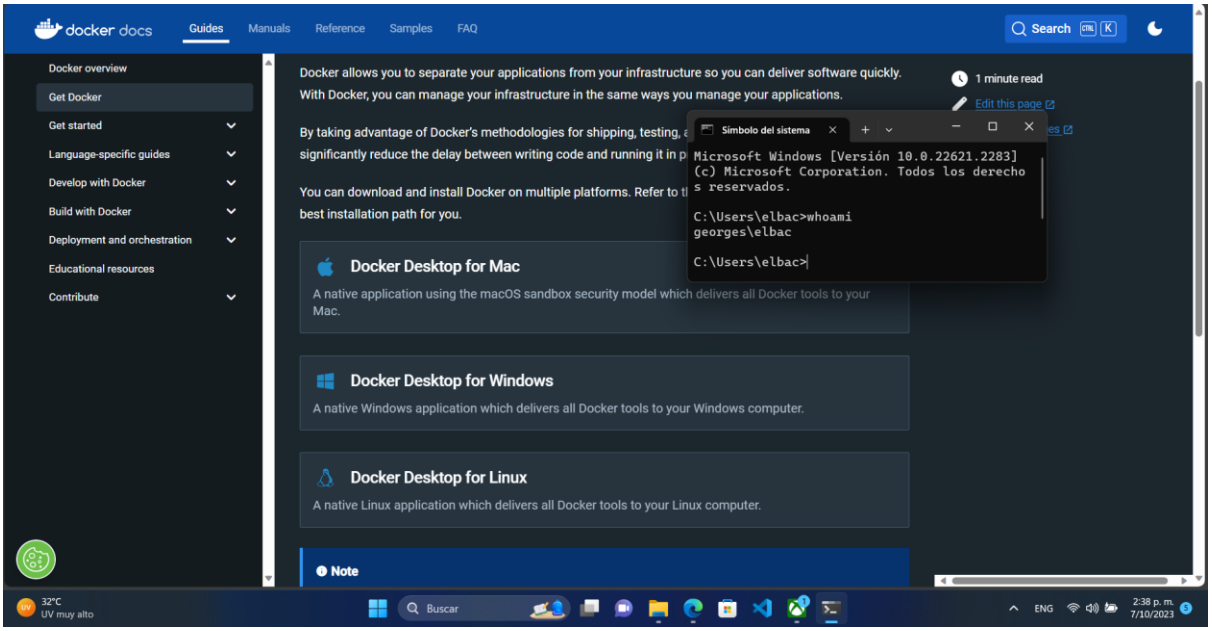
Docker Compose es una herramienta para definir y gestionar aplicaciones multi-contenedor. Permite describir las aplicaciones, sus dependencias, redes y volúmenes en un archivo docker-compose.yml.

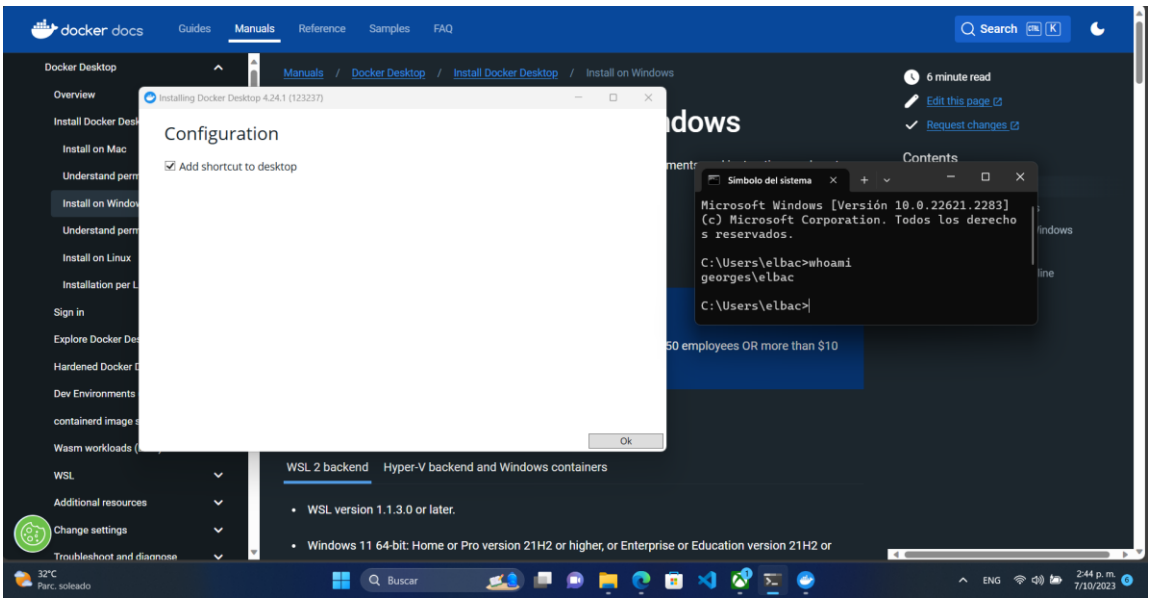
Cluster o Balanceo de carga

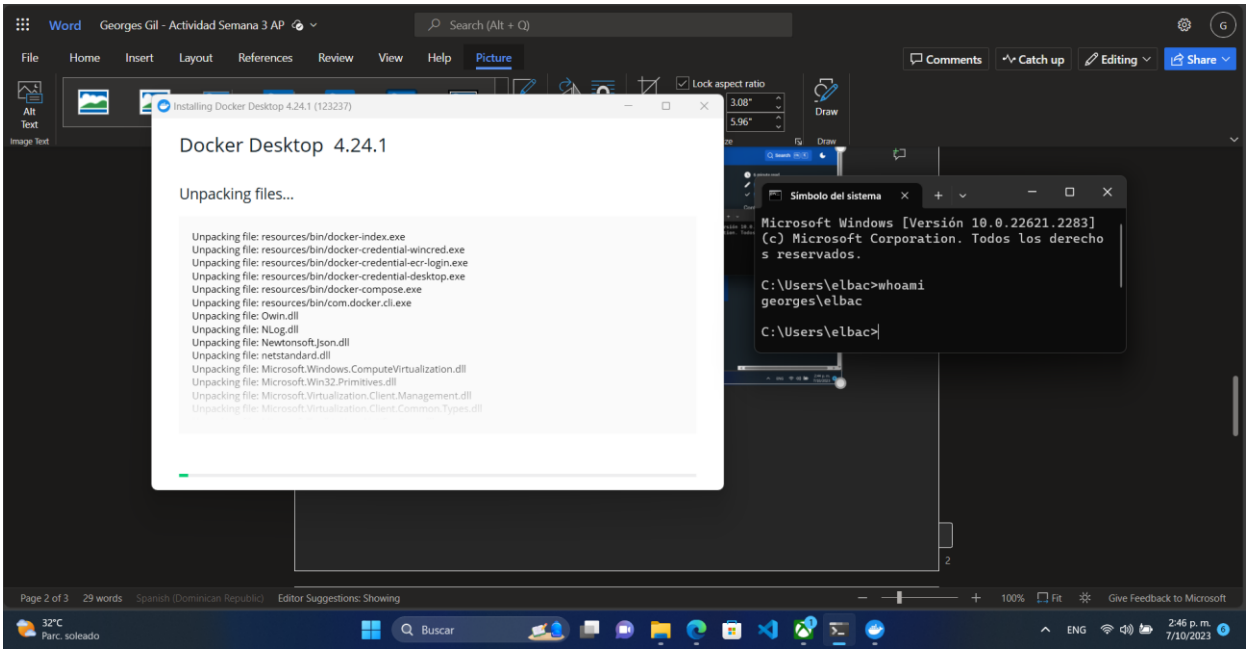
Consiste de un Front-1 (balanceador de carga), tres servidores web y un servidor GFS (Global File System) donde se almacenará la información. Para esto, se usa Swarm, un protocolo nativo de Docker. Convierte un conjunto de hosts en un solo host virtualizado de

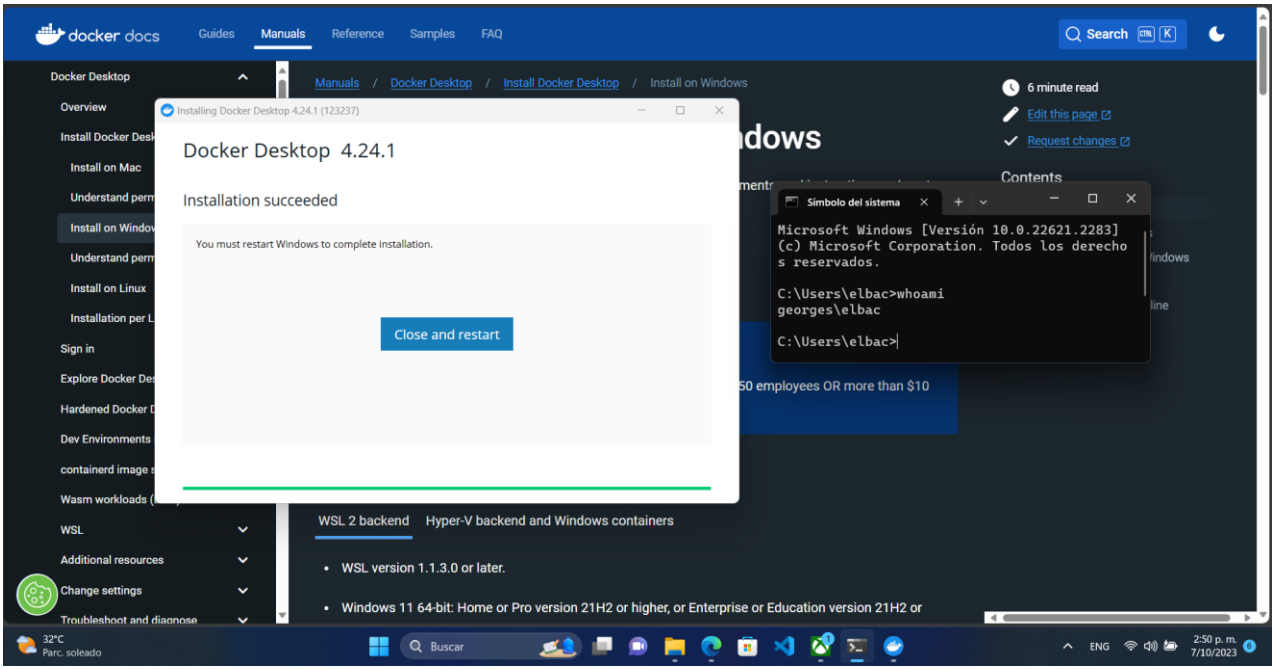
Docker. Algunas de sus características son: • Alta disponibilidad. • Balanceo de carga. • Descentralizado. • Escalabilidad

Instalación Docker









Ejemplos de Docker-Compose

```yaml
version: '3'
services:
  web:
    image: nginx
    ports:
      - "80:80"
  app:
    image: node:alpine
    volumes:
      - ./app:/app
    working_dir: /app
    command: "npm start"
    depends_on:
      - db
  db:
    image: mariadb
    environment:
      MYSQL_ROOT_PASSWORD: example
      MYSQL_DATABASE: mydb
```

Node.js, Nginx y MariaDB con Docker Compose

Node.js: Este servicio utiliza una imagen de Node.js para ejecutar una aplicación Node.js. Monta el código fuente de la aplicación desde el directorio local ./app al contenedor y ejecuta el comando npm start.

Nginx: El servicio de Nginx utiliza la imagen oficial de Nginx y redirige el tráfico del puerto 80 del host al puerto 80 del contenedor.

MariaDB: Utiliza la imagen oficial de MariaDB y configura las variables de entorno para establecer la contraseña de root y el nombre de la base de datos.

Ejercicio:

Crear una imagen de una aplicación que se conecte a una base de datos(Otra imagen de docker), con lo siguiente criterios de tecnologías por rango de matrículas:

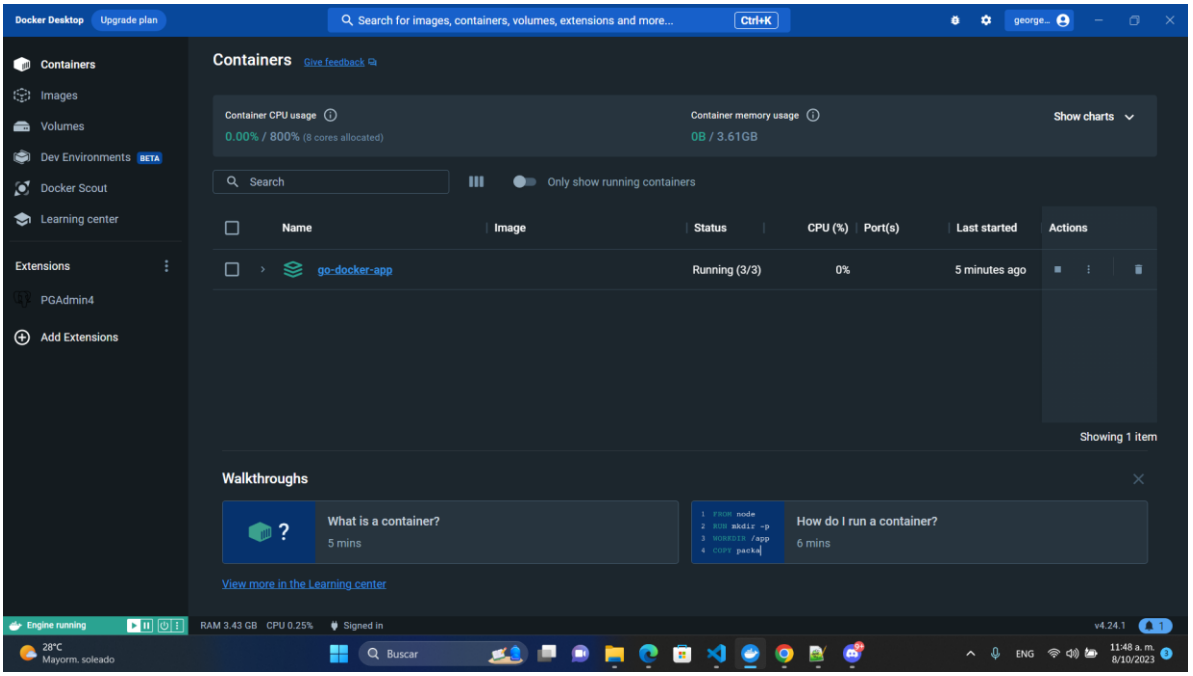
Matriculas (1-18-*)

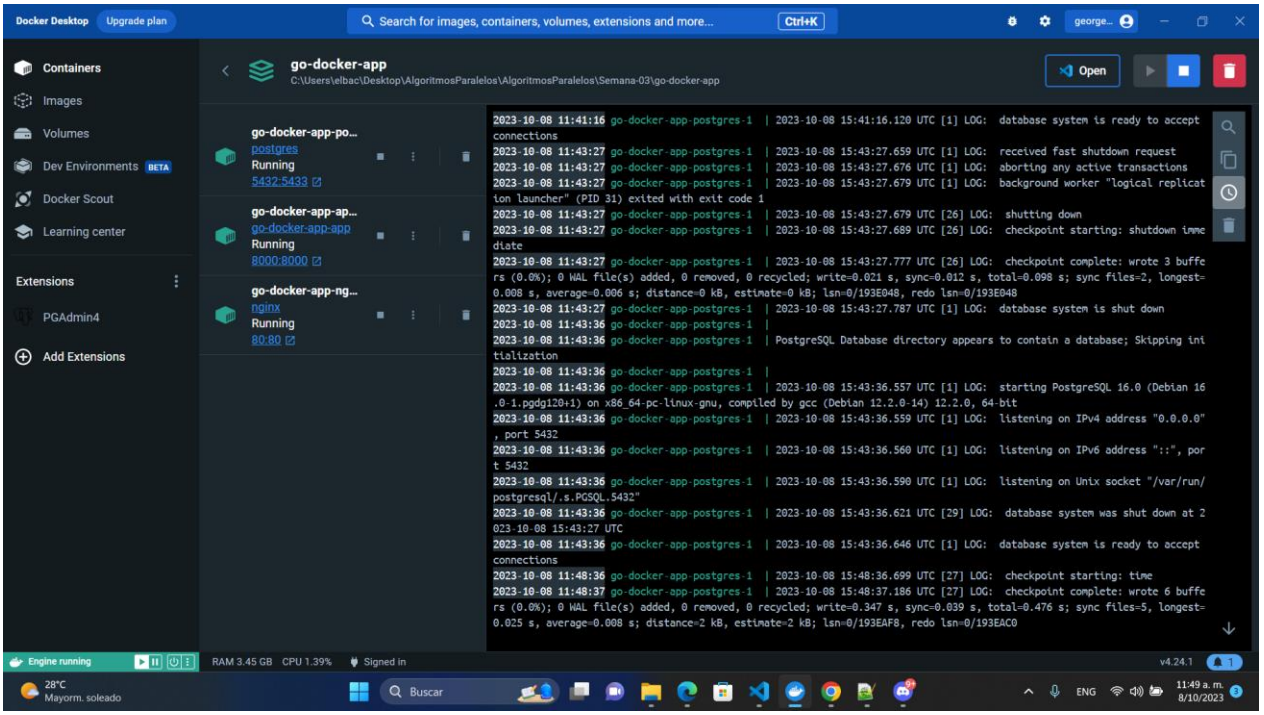
- GOLANG (APP)
- NGINX (Alojamiento)
- POSTGRESQL (BD)

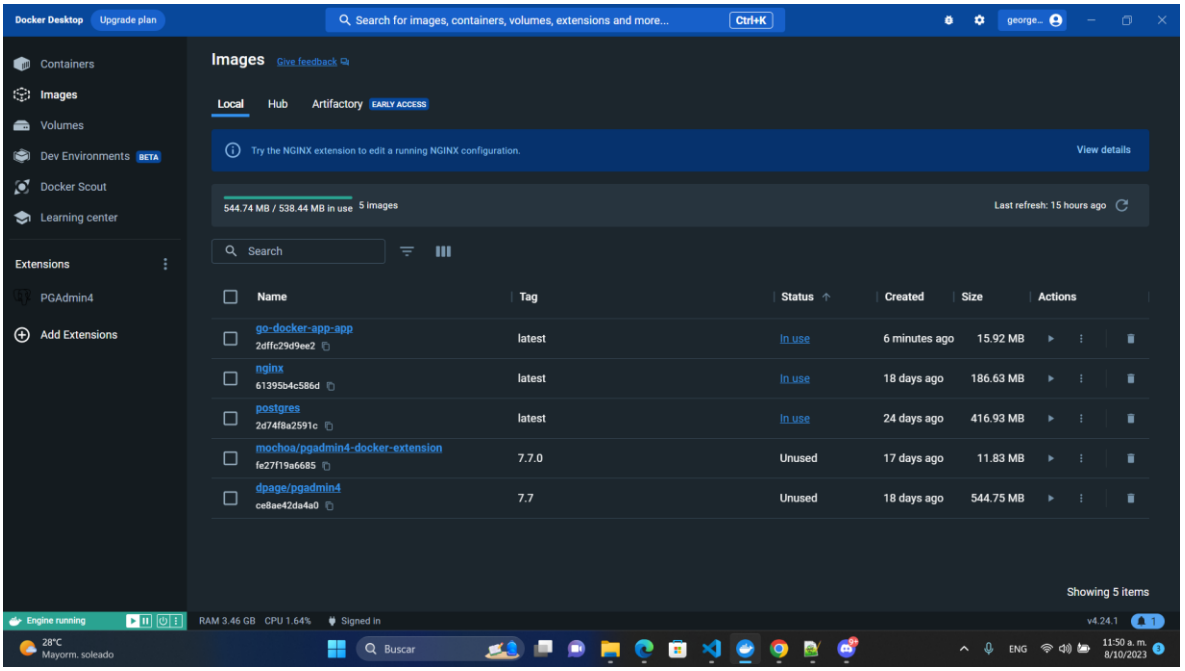
1. Aplicación que haga uso de un **CRUD** de una tabla en su BD.
2. Alojamiento del servicio.
3. Base de datos con su usuario y contraseña es su matricula y nombre del alumno.
4. Subir código de su app a Github.
5. Mostrar aplicación en clases.

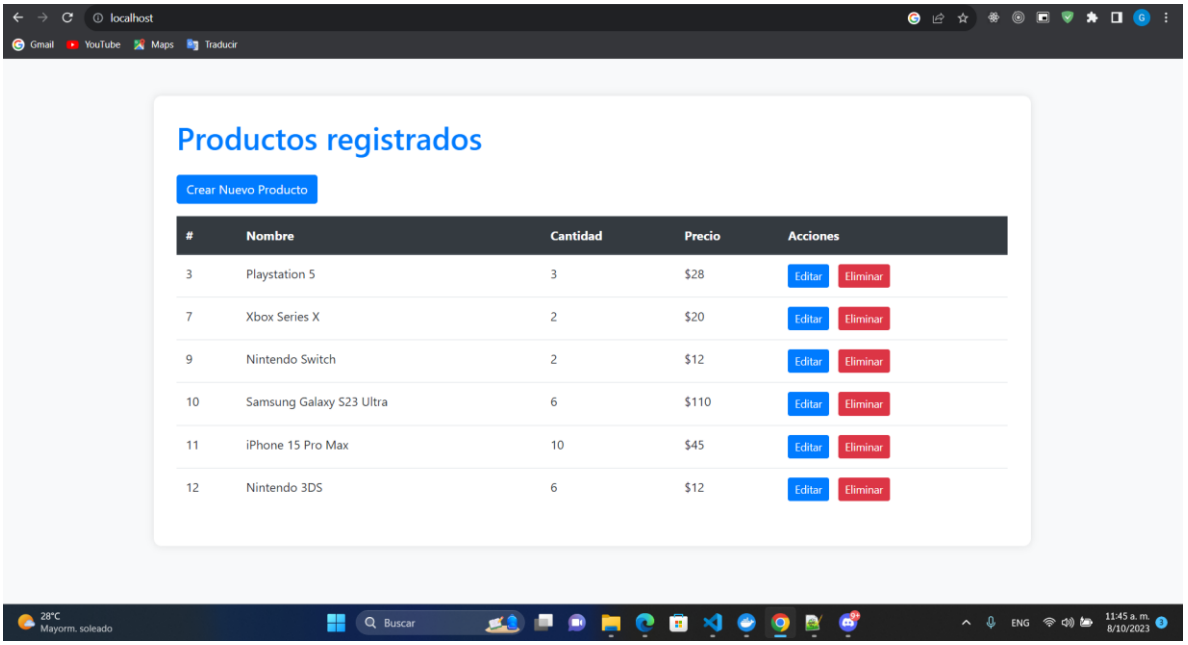
Nota: Uso de variables de entornos con Docker, uso de docker-compose y crear una imagen de su aplicación.

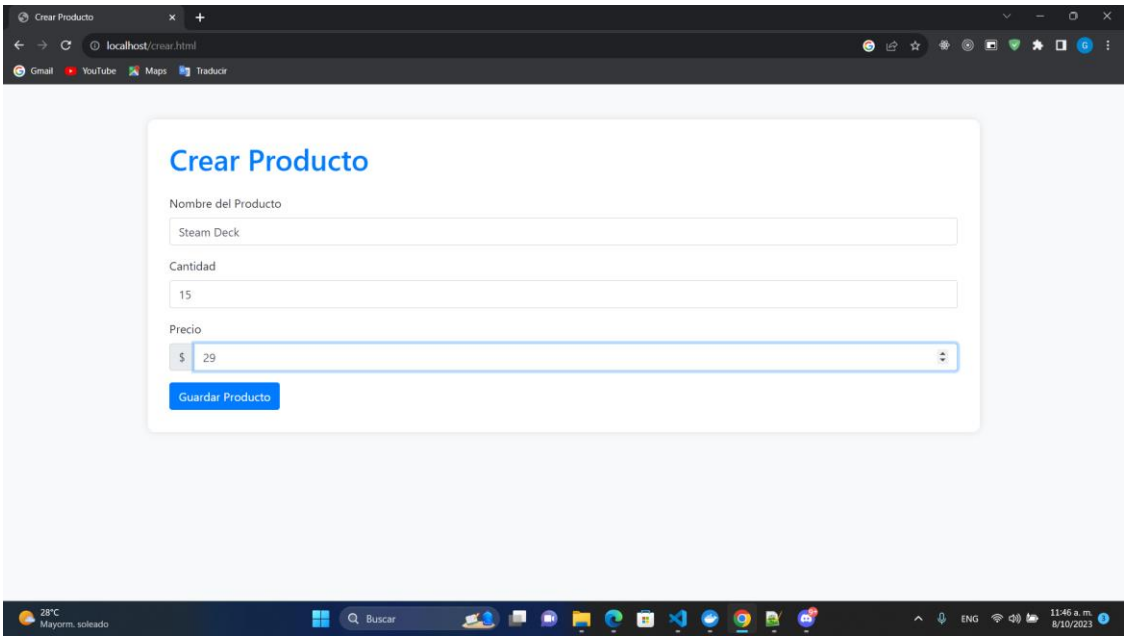
Link Github: [AlgoritmosParalelos/Semana-03 at main · GeorgesGil/AlgoritmosParalelos (github.com)]




Docker Desktop
Upgrade plan
Search for images, containers, volumes, extensions and more...
Ctrl+K
george...
Containers
Images
Volumes
Dev Environments
BETA
Docker Scout
Learning center
Extensions
PGAdmin4
Add Extensions
go-docker-app
C:\Users\elbac\Desktop\AlgoritmosParalelos\AlgoritmosParalelos\Semana-03\go-docker-app
Open
go-docker-app-po...
postgres
Running
5432:5433
go-docker-app-ap...
go-docker-app-app
Running
8000:8000
go-docker-app-ng...
nginx
Running
80:80
2023-10-08 11:41:16 go-docker-app-postgres-1 | 2023-10-08 15:41:16.120 UTC [1] LOG: database system is ready to accept connections
2023-10-08 11:43:27 go-docker-app-postgres-1 | 2023-10-08 15:43:27.659 UTC [1] LOG: received fast shutdown request
2023-10-08 11:43:27 go-docker-app-postgres-1 | 2023-10-08 15:43:27.676 UTC [1] LOG: aborting any active transactions
2023-10-08 11:43:27 go-docker-app-postgres-1 | 2023-10-08 15:43:27.679 UTC [1] LOG: background worker "logical replication launcher" (PID 31) exited with exit code 1
2023-10-08 11:43:27 go-docker-app-postgres-1 | 2023-10-08 15:43:27.679 UTC [26] LOG: shutting down
2023-10-08 11:43:27 go-docker-app-postgres-1 | 2023-10-08 15:43:27.689 UTC [26] LOG: checkpoint starting: shutdown immediate
2023-10-08 11:43:27 go-docker-app-postgres-1 | 2023-10-08 15:43:27.777 UTC [26] LOG: checkpoint complete: wrote 3 buffers (0.0%); 0 WAL file(s) added, 0 removed, 0 recycled; write=0.021 s, sync=0.012 s, total=0.098 s; sync files=2, longest=0.008 s, average=0.006 s; distance=0 kB, estimate=0 kB; lsn=0/193E048, redo lsn=0/193E048
2023-10-08 11:43:27 go-docker-app-postgres-1 | 2023-10-08 15:43:27.787 UTC [1] LOG: database system is shut down
2023-10-08 11:43:36 go-docker-app-postgres-1 |
2023-10-08 11:43:36 go-docker-app-postgres-1 | PostgreSQL Database directory appears to contain a database; Skipping initialization
2023-10-08 11:43:36 go-docker-app-postgres-1 |
2023-10-08 11:43:36 go-docker-app-postgres-1 | 2023-10-08 15:43:36.557 UTC [1] LOG: starting PostgreSQL 16.0 (Debian 16.0-1.pgdg120+1) on x86_64-pc-linux-gnu, compiled by gcc (Debian 12.2.0-14) 12.2.0, 64-bit
2023-10-08 11:43:36 go-docker-app-postgres-1 | 2023-10-08 15:43:36.559 UTC [1] LOG: listening on IPv4 address "0.0.0.0", port 5432
2023-10-08 11:43:36 go-docker-app-postgres-1 | 2023-10-08 15:43:36.560 UTC [1] LOG: listening on IPv6 address "::", port 5432
2023-10-08 11:43:36 go-docker-app-postgres-1 | 2023-10-08 15:43:36.590 UTC [1] LOG: listening on Unix socket "/var/run/postgresql/.s.PGSQL.5432"
2023-10-08 11:43:36 go-docker-app-postgres-1 | 2023-10-08 15:43:36.621 UTC [29] LOG: database system was shut down at 2023-10-08 15:43:27 UTC
2023-10-08 11:43:36 go-docker-app-postgres-1 | 2023-10-08 15:43:36.646 UTC [1] LOG: database system is ready to accept connections
2023-10-08 11:48:36 go-docker-app-postgres-1 | 2023-10-08 15:48:36.699 UTC [27] LOG: checkpoint starting: time
2023-10-08 11:48:37 go-docker-app-postgres-1 | 2023-10-08 15:48:37.186 UTC [27] LOG: checkpoint complete: wrote 6 buffers (0.0%); 0 WAL file(s) added, 0 removed, 0 recycled; write=0.347 s, sync=0.039 s, total=0.476 s; sync files=5, longest=0.025 s, average=0.008 s; distance=2 kB, estimate=2 kB; lsn=0/193EAF8, redo lsn=0/193EAC0
Engine running
RAM 3.45 GB CPU 1.39%
Signed in
v4.24.1
28°C
Mayorm. soleado
Buscar
ENG
11:49 a. m.
8/10/2023


Docker Desktop
Upgrade plan
Search for images, containers, volumes, extensions and more...
Ctrl+K
george...
Containers
Images
Volumes
Dev Environments
BETA
Docker Scout
Learning center
Extensions
PGAdmin4
Add Extensions
Images
Give feedback
Local
Hub
Artifactory
EARLY ACCESS
Try the NGINX extension to edit a running NGINX configuration.
View details
544.74 MB / 538.44 MB in use
5 images
Last refresh: 15 hours ago
Search
Name
Tag
Status
Created
Size
Actions
go-docker-app-app
2dffc29d9ee2
latest
In use
6 minutes ago
15.92 MB
nginx
61395b4c586d
latest
In use
18 days ago
186.63 MB
postgres
2d74f8a2591c
latest
In use
24 days ago
416.93 MB
mochoa/pgadmin4-docker-extension
fe27f19a6685
7.7.0
Unused
17 days ago
11.83 MB
dpage/pgadmin4
ce8ae42da4a0
7.7
Unused
18 days ago
544.75 MB
Showing 5 items
Engine running
RAM 3.46 GB CPU 1.64%
Signed in
v4.24.1
28°C
Mayorm. soleado
Buscar
ENG
11:50 a. m.
8/10/2023

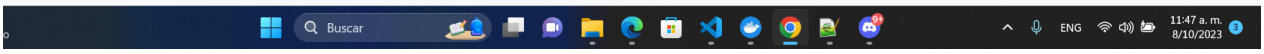

localhost
Gmail
YouTube
Maps
Traducir
Productos registrados
Crear Nuevo Producto
Nombre Cantidad Precio Acciones
3 Playstation 5 3 $28 Editar Eliminar
7 Xbox Series X 2 $20 Editar Eliminar
9 Nintendo Switch 2 $12 Editar Eliminar
10 Samsung Galaxy S23 Ultra 6 $110 Editar Eliminar
11 iPhone 15 Pro Max 10 $45 Editar Eliminar
12 Nintendo 3DS 6 $12 Editar Eliminar
28°C
Mayorm. soleado
Buscar
ENG
11:45 a. m.
8/10/2023


Crear Producto
localhost/crear.html
Gmail
YouTube
Maps
Traducir
Crear Producto
Nombre del Producto
Steam Deck
Cantidad
15
Precio
$
29
Guardar Producto
28°C
Mayorm. soleado
Buscar
ENG
11:46 a. m.
8/10/2023

Productos registrados

Crear Nuevo Producto

#	Nombre	Cantidad	Precio	Acciones
3	Playstation 5	3	$28	Editar Eliminar
7	Xbox Series X	2	$20	Editar Eliminar
9	Nintendo Switch	2	$12	Editar Eliminar
10	Samsung Galaxy S23 Ultra	6	$110	Editar Eliminar
11	iPhone 15 Pro Max	10	$45	Editar Eliminar
12	Nintendo 3DS	6	$12	Editar Eliminar
15	Steam Deck	15	$29	Editar Eliminar



Actualizar Producto

Nombre del Producto

Steam DOCK

Cantidad

15

Precio

$ 29

Actualizar Producto