

Taller Bases de Datos

Juan Sebastian Dueñas Robayo

833539

Corporación Universitaria Minuto de Dios

Ing. William Alexander Matallana Porras

10-66167: Arquitectura de Software

2025

1. Archivos .yaml

Los archivos YAML son usados por desarrolladores, pues contienen código fuente escrito en el lenguaje de programación YAML (YAML no es un lenguaje de marcas). El código contenido en un archivo YAML es legible por el ser humano y generalmente se utiliza para serializar datos. Con la ayuda de los archivos YAML, se pueden leer y escribir datos con independencia completa de los lenguajes de programación. Por tanto, los archivos YAML pueden utilizarse junto con otros muchos archivos contenedores de código fuente en distintos lenguajes, como C, C#, C++, Java, PHP y muchos más.

2. Archivos .json

Un archivo JSON (JavaScript Object Notation) es un formato de almacenamiento e intercambio de datos que utiliza una estructura de clave-valor para representar información de manera legible tanto para humanos como para máquinas. Se basa en la sintaxis de los objetos de JavaScript, aunque es compatible con muchos otros lenguajes de programación.

3. Archivos .yaml vs .json

YAML y JSON son dos formatos de serialización de datos que proporcionan un mecanismo de intercambio de datos legible por humanos. Dan formato a los datos de manera estandarizada para su intercambio entre aplicaciones de software, y su texto también es legible por humanos. Ambos representan datos como pares de clave y valor. JSON admite objetos de datos como valores, mientras que YAML no. Sin embargo, YAML admite más tipos de datos y utiliza de forma más próxima el lenguaje natural para que los desarrolladores puedan utilizarlo.

4. Uso Docker Compose

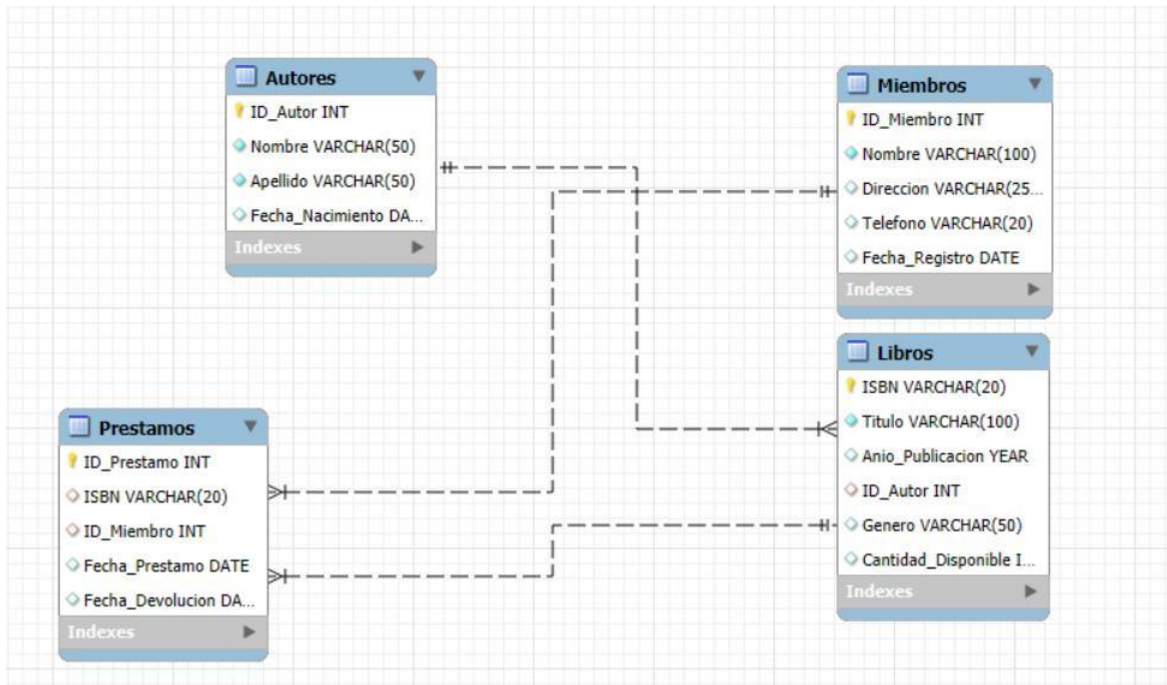
Docker Compose es una herramienta versátil que te permite definir y gestionar aplicaciones multi-contenedor de forma sencilla. Con Docker Compose, puedes describir la configuración de tu entorno de desarrollo en un archivo YAML, especificando los servicios, volúmenes y redes necesarios para tu aplicación. Luego, con un solo comando, puedes crear y ejecutar todos los contenedores definidos en tu archivo de configuración.

5. Creación de un contenedor usando Docker Compose

Dentro de la sección `services` del archivo `docker-compose.yml`, podemos definir los diferentes servicios y contenedores que formarán parte de nuestro entorno de desarrollo. Cada servicio se define con un nombre único y se especifica su configuración. A continuación, se muestra un ejemplo de definición de servicios:

```
1services:
2  service1:
3    image: nginx:latest
4    ports:
5      - 8080:80
6    volumes:
7      - ./app:/usr/share/nginx/html
8
9  service2:
10   image: mysql:latest
11   environment:
12     - MYSQL_ROOT_PASSWORD=password
13     - MYSQL_DATABASE=mydb
```

6. Ingeniería Inversa



Referencias

- Amazon Web Services. (s.f.). *The difference between YAML and JSON*. AWS. <https://aws.amazon.com/es/compare/the-difference-between-yaml-and-json/>
- Arsys. (s.f.). *Archivo JSON: qué es y para qué sirve*. Arsys. <https://www.arsys.es/blog/archivo-json-que-es-y-para-que-sirve>
- Imagina Formación. (s.f.). *¿Qué es Docker Compose?*. Imagina Formación. <https://imaginaformacion.com/tutoriales/que-es-docker-compose>
- Online Convert. (s.f.). *Convertidor de archivos en formato YML*. Online Convert. <https://www.online-convert.com/es/formato-de-archivo/yml>

