|  |  |
| --- | --- |
|  | Trabajo individual |

## Identificación del trabajo

|  |  |
| --- | --- |
| 1. **Módulo:** | 1 |
| 1. **Asignatura:** | Bases de Datos |
| 1. **RA:** | Poder Descargar e instalar motor de bases de datos PSQL y crear una base de datos |
| 1. **Docente Online:** | Nombre del docente |
| 1. **Fecha de entrega**: | Fecha de entrega del trabajo |
|  |  |
|  |  |

## Identificación del estudiante

|  |  |
| --- | --- |
| **Nombre y apellido** | **Carrera** |
| Christian Aránguiz | Técnico en Informática |

## Introducción

## La Universidad PULS-AR requiere de un sistema de información integral que permita llevar un control exhaustivo y actualizado de los libros disponibles en su biblioteca. Este sistema debe no solo registrar los detalles fundamentales, como el título de los libros, los autores, los lectores y las editoriales, sino también gestionar el préstamo de libros a los estudiantes, permitiendo un seguimiento eficiente de la disponibilidad y ubicación de cada ejemplar.

## En esta primera Actividad Práctica Integradora, se busca establecer las bases técnicas necesarias para el desarrollo de este sistema de gestión bibliotecaria. El enfoque de esta actividad inicial consiste en instalar y configurar el motor de base de datos PostgreSQL. Aunque inicialmente se sugiere la versión 10, se ha decidido utilizar PostgreSQL 12 Alpine. Esta versión cumple con los requisitos mínimos para ser compatible con Docker, permitiendo la creación de un entorno de base de datos en contenedores. Docker facilita la instalación, portabilidad y administración de PostgreSQL, optimizando el desarrollo del sistema.

## Tras la configuración del entorno, se procederá a la creación de la nueva base de datos de la mencionada biblioteca, que será la estructura central del sistema. Esta base de datos servirá como repositorio de la información clave sobre el inventario de libros, los autores y los préstamos realizados a los lectores.

## Desarrollo

Para emular un entorno de desarrollo más realista y eficiente, se utilizará Docker junto con Docker Compose para servir la base de datos. Esta configuración permite administrar PostgreSQL en un contenedor aislado, facilitando la portabilidad y la replicabilidad del entorno. Además, Docker Compose simplifica la configuración mediante un archivo de configuración, permitiendo el despliegue rápido de la base de datos con los parámetros necesarios para la actividad. Se ha optado por PostgreSQL 12 Alpine, ya que es la versión mínima compatible con Docker, optimizando el uso de recursos y garantizando estabilidad en el entorno de contenedores.

1. **Instalación y Configuración de PostgreSQL 12 Alpine**  
    La instalación se llevó a cabo utilizando Docker, con una imagen ligera de Alpine para optimizar el uso de recursos. Se configuró un archivo docker-compose.yml junto con un archivo .env que contiene las variables de entorno necesarias, tales como el usuario, contraseña y nombre de la base de datos. Esto permite administrar la configuración de PostgreSQL de forma flexible y segura.

A screen shot of a computer

Description automatically generated

Imagen 1: Muestra el archivo docker-compose.yml junto con el archivo .env, donde se especifican los valores de configuración para el contenedor Docker que ejecuta PostgreSQL.

Una vez configurado Docker y PostgreSQL, se puso en marcha el contenedor y se verificó su funcionamiento.

A screenshot of a computer

Description automatically generated

Imagen 2: Evidencia el estado del contenedor en Docker, indicando que PostgreSQL está en ejecución y listo para aceptar conexiones.

1. **Conexión y Creación de la Base de Datos "biblioteca"**  
    Con el contenedor de PostgreSQL operativo, se estableció una conexión utilizando la interfaz gráfica. Se configuraron los parámetros de conexión, incluyendo el nombre de la conexión, host, usuario, y nombre de la base de datos.

A screenshot of a computer

Description automatically generated

Imagen 3: Muestra la configuración de conexión a la base de datos "biblioteca", asegurando que todos los parámetros están correctamente establecidos para acceder al sistema.

Luego, se creó la base de datos llamada **biblioteca** dentro del entorno de PostgreSQL. Una vez creada, se verificó su existencia en el árbol de bases de datos.

A screenshot of a computer

Description automatically generated

Imagen 4: Muestra el árbol de bases de datos en el cliente PostgreSQL, confirmando la creación exitosa de la base de datos "biblioteca".

**Extra: Creación de la Tabla "libros"**  
Para avanzar en la estructura del sistema bibliotecario, se creó una tabla denominada libros dentro de la base de datos "biblioteca". Esta tabla incluye los campos necesarios para almacenar la información básica de cada libro, como el nombre del libro, autor, ISBN, editorial y año de publicación:

A screenshot of a computer program

Description automatically generated

Imagen 5: Muestra la estructura de la tabla libros, con cada columna y su tipo de dato, confirmando la correcta creación de la tabla en la base de datos.

## Conclusión

La implementación de este sistema de información para la biblioteca de la Universidad PULS-AR mediante PostgreSQL y Docker ha demostrado ser una solución robusta y eficiente para gestionar datos en un entorno de desarrollo controlado. A través del uso de Docker y Docker Compose, se facilitó la configuración y administración de la base de datos, logrando un entorno replicable y portable que puede adaptarse a futuros desarrollos o modificaciones.

La elección de PostgreSQL 12 Alpine permitió optimizar los recursos del sistema y asegurar la compatibilidad con contenedores, garantizando un rendimiento estable. Además, se establecieron las bases para la estructura de datos mediante la creación de la base de datos "biblioteca" y la tabla "libros", que servirán como núcleo del sistema de gestión bibliotecaria.

Este proyecto no solo cumplió con los requisitos iniciales de la actividad, sino que también proporciona una base sólida para futuras expansiones, donde se podrán añadir funcionalidades adicionales para mejorar la gestión de la información bibliográfica y los préstamos de libros.

## Bibliografía

* Docker, Inc. (n.d.). *Docker Documentation*. Docker Hub. Recuperado de https://hub.docker.com
* TablePlus. (n.d.). *Documentation*. Recuperado de https://docs.tableplus.com/ PostgreSQL Global Development Group. (n.d.). *PostgreSQL Documentation*. Recuperado de <https://www.postgresql.org/docs>