Enunciado de Pedido Empresarial

Empresa solicitante: Editorial Global Books S.A.

Consultora: Twooweb

Contexto:

Nuestra compañía gestiona un amplio catálogo de libros junto con la interacción de miles de usuarios que generan valoraciones. Actualmente almacenamos estos datos en múltiples ficheros CSV y necesitamos migrar y optimizar nuestro sistema hacia PostgreSQL, garantizando integridad, escalabilidad y capacidad analítica.

Requerimiento a Twooweb:

Se solicita el diseño, implementación y validación de un entorno de base de datos en **PostgreSQL**, desplegado mediante **Docker y pgAdmin**, que cumpla con los siguientes objetivos:

1. Preparación del entorno:

- Montaje de PostgreSQL y pgAdmin mediante docker-compose.yml con persistencia de datos.
- o Acceso y registro del servidor desde el navegador a través de pgAdmin.

2. Carga de datasets:

- Integración de los archivos CSV (libros, usuarios, valoraciones) en el contenedor.
- Creación de tablas en PostgreSQL con tipado correcto y validaciones mediante PRIMARY KEY y CHECK.

3. Importación y limpieza:

- o Carga masiva de datos con COPY FROM.
- Detección y corrección de inconsistencias: registros huérfanos, datos inválidos o incompletos.

4. Modelo relacional:

- Establecimiento de relaciones con claves primarias y foráneas (ON DELETE CASCADE).
- Verificación de la integridad referencial mediante inserción de datos de prueba.

5. Optimización y rendimiento:

- Creación de distintos tipos de índices (B-Tree, Hash, GIN, BRIN y compuestos).
- Medición de rendimiento con EXPLAIN ANALYZE, comparando consultas antes y después de aplicar los índices.

6. Consultas analíticas:

- Ejecución de una batería de consultas que abarque niveles básico, intermedio, avanzado y experto (JOINs, agregaciones, subconsultas, CTEs y funciones ventana).
- Desarrollo de un reto final de filtrado colaborativo (uso de la función CORR()
 para encontrar usuarios con gustos similares).

Competencias esperadas de la consultoría Twooweb al entregar el proyecto:

- Entorno funcional de PostgreSQL en Docker con persistencia de datos.
- Modelo relacional robusto, limpio y validado.
- Optimización de rendimiento con índices y consultas eficientes.
- Conjunto de consultas SQL que permitan explotar el valor analítico de los datos.

Consultas:

Bloque 1 – Nivel Básico (lectura simple)

Ejercicio 1: Mostrar todos los usuarios

Enunciado: Obtén todos los registros de la tabla usuarios.

Ejercicio 2: Mostrar los primeros 10 libros

Enunciado: Obtén los primeros 10 libros registrados en la tabla libros.

Ejercicio 3: Buscar un usuario concreto

Enunciado: Encuentra el usuario con user_id = 12345.

Ejercicio 4: Contar usuarios

Enunciado: ¿Cuántos usuarios hay registrados?

Ejercicio 5: Libros publicados en 2000

Enunciado: Lista los títulos de libros publicados en el año 2000.

Bloque 2 – Nivel Intermedio (JOINs y agregaciones)

Ejercicio 6: Top 10 libros con más ratings

Enunciado: Encuentra los 10 libros que más valoraciones tienen.

Ejercicio 7: Promedio de rating por libro

Enunciado: Calcula el promedio de rating de cada libro.

Ejercicio 8: Edad promedio de usuarios que valoran

Enunciado: ¿Cuál es la edad promedio de los usuarios que dieron ratings?

Ejercicio 9: Autores más valorados

Enunciado: Muestra los 5 autores con más valoraciones.

Ejercicio 10: Usuarios de USA

Enunciado: Muestra los usuarios cuya location contenga "usa".

Bloque 3 – Nivel Avanzado (subconsultas y HAVING)

Ejercicio 11: Mejor libro (mínimo 50 ratings)

Ñ

Ejercicio 12: Usuarios con más de 100 valoraciones

Ejercicio 13: Libros más populares entre <25 años

Ejercicio 14: Autores con promedio > 8

Ejercicio 15: Usuarios sin ratings

● Bloque 4 – Nivel Experto (CTE, funciones ventana)

Ejercicio 16: Ranking de usuarios por cantidad de ratings

Ejercicio 17: Promedio de ratings por década

Ejercicio 18: Usuario más joven y más viejo

Ejercicio 19: Top usuarios por autor

Ejercicio 20: Promedio de ratings por país

Reto Final – Difícil (colaborative filtering)

Enunciado: Encuentra los 3 pares de usuarios más similares según sus ratings de libros en común.

Solución: