Reporte de Decisiones Importantes del Proyecto.

Jorge Luis Guzmán, Catalina Gonzalez, Amelia Diaz, Úrsula Saez, Monserrat Zúñiga.
Departamento de física, Universidad de Santiago de Chile.

1. Diseño de la Base de Datos

Diagrama Relacional

Durante la etapa de diseño conceptual de la base de datos, se partió desde el diagrama relacional base propuesto en el enunciado del problema. Sin embargo, tras un análisis más detallado de las relaciones y entidades necesarias para responder adecuadamente a las consultas requeridas, se tomó la decisión de **eliminar la entidad VENDEDOR** del modelo.

Esta decisión se justificó debido a que la entidad **EMPLEADOR** cumplía correctamente el rol funcional necesario dentro del sistema, permitiendo representar de manera adecuada a las personas responsables de las ventas sin introducir redundancias innecesarias en el modelo. De esta forma, se logró un diseño más simple, coherente y fácil de mantener, respetando los principios de normalización.

El diagrama relacional final incluye únicamente las entidades y relaciones estrictamente necesarias para el funcionamiento del proyecto y para responder las consultas solicitadas.

2. Implementación de la Base de Datos

Herramientas Utilizadas

Para la implementación de la base de datos se utilizó el gestor MySQL, junto con la interfaz gráfica HeidiSQL, ya que era una herramienta previamente conocida por el grupo, lo que permitió agilizar el proceso de creación, modificación y prueba de la base de datos.

Población de Datos

Inicialmente, se decidió trabajar con un volumen moderado de datos, considerando:

- 30 galerías
- 1000 productos

No obstante, durante el proceso de implementación se detectaron errores que obligaron a repoblar la base de datos, lo que derivó en la creación de una nueva base dentro de HeidiSQL para evitar inconsistencias.

Posteriormente, se intentó aumentar significativamente el volumen de datos agregando dos ceros adicionales a cada cantidad, con el objetivo de acercarse al tamaño recomendado para el procesamiento con Apache Spark. Sin embargo, esta configuración resultó inviable,

ya que el código no lograba ejecutarse correctamente y los tiempos de carga se volvían excesivos.

Tras varias pruebas y evaluaciones, se optó por una solución intermedia, estableciendo las siguientes cantidades finales:

- CANTIDAD_GALERIAS = 300
- CANTIDAD_LOCALES = 600
- CANTIDAD_EMPLEADOS = 2000
- CANTIDAD_CLIENTES = 3000
- CANTIDAD_PRODUCTOS = 10000
- CANTIDAD_VENTAS = 6000

Esta configuración permite mantener la coherencia de los datos, asegurar tiempos de ejecución razonables y evitar errores durante la carga.

Los scripts de creación (`create_database.sql`), población (`populate_database.sql`) y consultas (`queries.sql`) se realizaron sin inconvenientes y cumplen con las restricciones de integridad definidas en el diseño.

Limitaciones de Tamaño de Datos

El enunciado del proyecto recomienda trabajar con un volumen de datos superior a 128 MB para el uso de múltiples particiones en Apache Spark. Sin embargo, debido a las limitaciones de hardware de los computadores del grupo, no fue posible generar ni procesar directamente una base de datos de ese tamaño dentro del gestor SQL.

Como solución alternativa, se decidió exportar los datos a archivos `.csv`, lo que permitió manejar volúmenes mayores de información sin comprometer la estabilidad del sistema.

3. Procesamiento de Datos con Apache Spark

Para cumplir con los requisitos del procesamiento distribuido, se creó una nueva base de datos independiente, diseñada exclusivamente para ser utilizada con Apache Spark, la cual alcanza un tamaño aproximado de 137 MB.

El aumento del tamaño de los datos se logró mediante la generación de archivos `.csv` con un mayor volumen de registros, lo que permitió simular un escenario realista de Big Data y cumplir con el requisito mínimo de tamaño solicitado.

No obstante, al momento de realizar el procesamiento con Apache Spark, surgieron dificultades técnicas debido a que el entorno de ejecución disponible corresponde a Windows, mientras que Apache Spark presenta un funcionamiento más estable y directo en sistemas Linux. Esta incompatibilidad impidió ejecutar correctamente el procesamiento distribuido, a pesar de contar con los datos y el código preparado.

Conclusión

A lo largo del desarrollo del proyecto se tomaron decisiones clave orientadas a garantizar la funcionalidad, coherencia y viabilidad técnica del sistema, adaptándose constantemente a las limitaciones del entorno de trabajo. El diseño de la base de datos fue refinado para evitar redundancias, la implementación se ajustó mediante pruebas iterativas, y se exploraron soluciones alternativas para cumplir con los requisitos de procesamiento de grandes volúmenes de datos.

A pesar de las restricciones técnicas encontradas, el proyecto permitió aplicar de forma práctica los conceptos aprendidos en el curso y comprender los desafíos reales asociados al diseño, implementación y procesamiento de bases de datos a gran escala.