

Proyecto de Bases de datos para Gestión de Riesgos Financieros

Dario Pino, Andrés Julián Barreto, John Aldemar Gonzalez

Universidad Central
Maestria en Analitica de datos
Curso de Bases de Datos
Bogotá, Colombia

dpinot@ucentral.edu.co, abarretop1@ucentral.edu.co, jgonzalezl8@ucentral.edu.co

7 de mayo de 2024

Índice

1. Introducción (Max 250 Palabras) - (<i>Primera entrega</i>)	3
2. Características del proyecto de investigación que hace uso de Bases de Datos (Max 500 Palabras) - (<i>Primera entrega</i>)	3
2.1. Titulo del proyecto de investigación (Max 100 Palabras) - (<i>Primera entrega</i>)	4
2.2. Objetivo general (Max 100 Palabras) - (<i>Primera entrega</i>)	4
2.2.1. Objetivos especificos (Max 100 Palabras) - (<i>Primera entrega</i>)	4
2.3. Alcance (Max 200 Palabras) - (<i>Primera entrega</i>)	4
2.4. Pregunta de investigación (Max 100 Palabras) - (<i>Primera entrega</i>)	5
2.5. Hipotesis (Max 100 Palabras) - (<i>Primera entrega</i>)	5
3. Reflexiones sobre el origen de datos e información (Max 400 Palabras) - (<i>Primera entrega</i>)	6
3.1. ¿Cual es el origen de los datos e información ? (Max 100 Palabras) - (<i>Primera entrega</i>)	6
3.2. ¿Cuales son las consideraciones legales o eticas del uso de la información? (Max 100 Palabras) - (<i>Primera entrega</i>)	6
3.3. ¿Cuales son los retos de la información y los datos que utilizara en la base de datos en terminos de la calidad y la consolidación? (Max 100 Palabras) - (<i>Primera entrega</i>)	6

3.4. ¿Que espera de la utilización de un sistema de Bases de Datos para su proyecto? (Max 100 Palabras) - (<i>Primera entrega</i>)	7
4. Diseño del Modelo de Datos del SMBD (Sistema Manejador de Bases de Datos)(<i>Primera entrega</i>)	8
4.1. Características del SMBD (Sistema Manejador de Bases de Datos) para el proyecto (<i>Primera entrega</i>)	8
4.2. Diagrama modelo de datos (<i>Primera entrega</i>)	8
4.2.1. Modelo Lógico:	8
4.2.2. Modelo Relacional:	9
4.3. Imágenes de la Base de Datos (<i>Primera entrega</i>)	10
4.4. Código SQL - lenguaje de definición de datos (DDL) (<i>Primera entrega</i>)	14
4.5. Código SQL - Manipulación de datos (DML) (<i>Primera entrega</i>)	17
4.6. Código SQL + Resultados: Vistas (<i>Primera entrega</i>)	22
4.7. Código SQL + Resultados: Triggers (<i>Primera entrega</i>)	22
4.8. Código SQL + Resultados: Funciones (<i>Primera entrega</i>)	23
4.9. Código SQL + Resultados: procedimientos almacenados (<i>Primera entrega</i>)	24
5. Bases de Datos No-SQL (<i>Segunda entrega</i>)	26
5.1. Diagrama Bases de Datos No-SQL (<i>Segunda entrega</i>)	26
5.2. SMBD utilizado para la Base de Datos No-SQL (<i>Segunda entrega</i>)	26
6. Aplicación de ETL (Extract, Transform, Load) y Bodega de Datos (<i>Tercera entrega</i>)	27
6.1. Ejemplo de aplicación de ETL y Bodega de Datos (<i>Tercera entrega</i>)	27
6.2. Automatización de Datos (<i>Tercera entrega</i>)	27
6.3. Integración de Datos (<i>Tercera entrega</i>)	27
7. Proximos pasos (<i>Tercera entrega</i>)	28
8. Lecciones aprendidas (<i>Tercera entrega</i>)	29
9. Bibliografía	30

1. Introducción (Max 250 Palabras) - (*Primera entrega*)

Este proyecto tiene como objetivo principal desarrollar una base de datos especializada que categorice las transacciones de fraude con tarjetas de crédito y débito, permitiendo el uso de esta información para realizar mediciones precisas y generar recomendaciones estratégicas para reducir la incidencia del fraude en Colombia.

La base de datos estará diseñada para capturar y almacenar de manera eficiente una amplia gama de datos sobre transacciones, incluyendo detalles sobre las transacciones sospechosas y confirmadas como fraudulentas. Este registro detallado es esencial para identificar patrones y tendencias que son vitales para entender y combatir el comportamiento del fraude.

Además, el sistema permitirá un monitoreo continuo y en near real time de las transacciones, facilitando la detección temprana de posibles fraudes y permitiendo a las instituciones financieras adaptar rápidamente sus estrategias de mitigación frente a tácticas fraudulentas que emergen constantemente.

El proyecto también incluirá herramientas para generar reportes visuales y recomendaciones personalizadas. Estos informes no solo resaltarán los comportamientos más vulnerables y los tipos de fraude más prevalentes, sino que también proporcionarán sugerencias concretas para ajustar políticas de seguridad y procedimientos operativos, ayudando a las instituciones financieras a implementar estrategias más efectivas para proteger a sus clientes y reducir las pérdidas económicas debido al fraude.

En conclusión, este proyecto busca transformar la manera en que las entidades financieras en Colombia manejan el riesgo de fraude, convirtiendo los datos en una herramienta poderosa para la prevención y el control estratégico del fraude en tarjetas de crédito y débito.

2. Características del proyecto de investigación que hace uso de Bases de Datos (Max 500 Palabras) - (*Primera entrega*)

Este proyecto presenta varias características clave que son fundamentales para su éxito las cuales relacionamos a continuación:

Colección de Datos Amplia y Diversa: Las bases de datos en estos proyectos deben incluir una gran cantidad de transacciones financieras, cubriendo un rango amplio de situaciones, desde transacciones legítimas hasta diversas formas de fraude. Esto permite un análisis más profundo y la capacidad de detectar patrones sutiles y complejos de actividad fraudulenta.

Integridad y Seguridad de los Datos: Dada la sensibilidad de la información financiera, es crucial que la base de datos garantice la integridad y la seguridad de los datos. Esto implica implementar medidas de seguridad robustas para proteger la información contra accesos no autorizados y garantizar que los datos no sean alterados o corrompidos.

Flexibilidad y Escalabilidad: La base de datos debe ser flexible y escalable para adaptarse al crecimiento del volumen de datos y a la evolución de las técnicas de fraude. Debe permitir la fácil integración de nuevas fuentes de datos y la implementación de nuevas reglas de detección y algoritmos de análisis sin degradar el rendimiento.

Herramientas de Visualización y Reporte: Es importante que el proyecto incluya herramientas que permitan visualizar los datos y generar reportes. Estas herramientas ayudan a los analistas a entender mejor los patrones de fraude, a comunicar hallazgos a las partes interesadas y a tomar decisiones informadas sobre estrategias de mitigación.

2.1. Título del proyecto de investigación (Max 100 Palabras) - (*Primera entrega*)

Sistema de Detección y Prevención de Fraude en Tarjetas de Crédito y Débito: Desarrollo de una Base de Datos Segura y Dinámica para Análisis en Near Real Time y Respuesta Proactiva.

2.2. Objetivo general (Max 100 Palabras) - (*Primera entrega*)

Desarrollar y establecer un sistema avanzado de detección y prevención de fraude en transacciones con tarjetas de crédito y débito, utilizando una base de datos segura y dinámica para el análisis en near real time. Este sistema no sólo identificará y medirá patrones de fraude, sino que también proporcionará datos para generar recomendaciones estratégicas basadas en datos para minimizar el riesgo y proteger a las entidades financieras y sus clientes.

2.2.1. Objetivos específicos (Max 100 Palabras) - (*Primera entrega*)

- Desarrollar una base de datos especializada que permita la captura, almacenamiento y visualización eficiente de datos relacionados con el fraude financiero en tarjetas. Esta base de datos deberá ser capaz de soportar análisis en tiempo real y facilitar la identificación de patrones sospechosos mediante una interfaz clara y accesible.
- Configurar y establecer los diferentes procesos técnicos dentro de la base de datos, incluyendo vistas, triggers, funciones y procedimientos almacenados. Estos procesos están diseñados para optimizar la operación y respuesta del sistema frente a transacciones potencialmente fraudulentas, asegurando un manejo de datos ágil y seguro.
- Integrar la base de datos diseñada para la detección de fraudes con otros sistemas y fuentes de datos existentes dentro de la organización. Este objetivo busca garantizar la coherencia y consistencia de los datos a través de las diferentes plataformas, facilitando una visión unificada del riesgo financiero y mejorando la capacidad de respuesta ante el fraude.
- Definir y establecer métricas claras y criterios de medición del fraude, incluyendo indicadores clave de rendimiento (KPIs). Estos indicadores permitirán cuantificar y evaluar la magnitud del fraude en las transacciones, proporcionando una base sólida para la toma de decisiones y la formulación de estrategias de mitigación efectivas.

2.3. Alcance (Max 200 Palabras) - (*Primera entrega*)

El alcance de este proyecto se centra en desarrollar una base de datos avanzada diseñada específicamente para la gestión de fraudes en transacciones con tarjetas de crédito y débito. Esta base de datos incluirá la implementación de estructuras de almacenamiento eficientes, junto con procesos automatizados como vistas, triggers, y procedimientos almacenados que permitirán análisis en tiempo real y una detección temprana de actividades sospechosas. Además, se llevará a cabo la integración de esta base de datos con otros sistemas de información y bases de datos ya existentes en la organización, asegurando la coherencia y consistencia de los datos a través de las diferentes plataformas y departamentos.

Para evaluar la efectividad de las estrategias de prevención de fraude, se establecerán métricas específicas y KPIs, junto con un sistema de reportes que permitirá un monitoreo constante y una evaluación del impacto del fraude en la organización. También se incluirá en el alcance del proyecto la capacitación de usuarios y partes interesadas en el manejo eficiente del sistema y la interpretación de los datos y reportes generados. Finalmente, el proyecto garantizará que todas las operaciones y manipulaciones de datos cumplan con las

regulaciones de privacidad y seguridad de datos aplicables, proporcionando una solución integral y segura para la detección y prevención de fraude con tarjetas.

2.4. Pregunta de investigación (Max 100 Palabras) - (*Primera entrega*)

¿Cómo puede una base de datos especializada mejorar la detección y prevención de fraude en transacciones con tarjetas de crédito y débito, y cuál es su impacto en la reducción de pérdidas financieras para las instituciones financieras en Colombia?

2.5. Hipotesis (Max 100 Palabras) - (*Primera entrega*)

La implementación de una base de datos especializada y altamente integrada mejora significativamente la detección y prevención del fraude en transacciones con tarjetas de crédito y débito, resultando en una reducción notable de las pérdidas financieras para las instituciones financieras en Colombia.

3. Reflexiones sobre el origen de datos e información (Max 400 Palabras) **- (Primera entrega)**

Desde su introducción como un método de pago en comercios en 1950, las tarjetas de crédito han presentado una oportunidad para la generación de fraude, una preocupación que las entidades bancarias y las franquicias abordan mediante informes mensuales. Sin embargo, estos informes suelen ser planos y se presentan en formatos de Excel, lo que limita su utilidad debido al volumen de datos históricos involucrados. Esta falta de eficacia en el análisis puede obstaculizar la detección y prevención efectiva del fraude. La necesidad de soluciones más avanzadas se hace evidente para mejorar la capacidad de las instituciones financieras para identificar patrones de comportamiento sospechoso y tomar medidas proactivas. La implementación de sistemas de gestión de datos más sofisticados, como bases de datos relacionales o herramientas de análisis de big data, podría permitir un análisis más profundo y en tiempo real de los datos de fraude. Esto facilitaría la identificación temprana de actividades fraudulentas, proporcionando a las instituciones financieras una ventaja crucial en la lucha contra el fraude y mejorando la seguridad de las transacciones con tarjetas de crédito para los consumidores.

3.1. ¿Cual es el origen de los datos e información ? (Max 100 Palabras) - (Primera entrega)

El origen de los datos e información proviene de un reporte detallado sobre el fraude asociado con las franquicias Visa y Mastercard. Este reporte se enfoca específicamente en el canal de adquisición y está dirigido al contexto colombiano. Dicho canal implica a las entidades que procesan pagos con tarjetas de crédito o débito, lo que indica que la información recabada es crucial para entender las dinámicas y el volumen de actividades fraudulentas dentro del país. Este informe constituye una fuente primaria valiosa para analizar y mitigar los riesgos relacionados con el fraude en transacciones realizadas mediante estas franquicias en Colombia.

3.2. ¿Cuales son las consideraciones legales o eticas del uso de la información? (Max 100 Palabras) - (Primera entrega)

Las consideraciones legales y éticas del uso de la información del reporte de fraude de Visa y Mastercard en Colombia incluyen garantizar la privacidad y protección de datos personales, cumpliendo con normativas como la Ley de Protección de Datos. Éticamente, es fundamental usar la información de manera responsable, evitando estigmatizaciones y asegurando que las medidas de prevención y corrección no vulneren derechos individuales. Legalmente, se debe respetar el marco jurídico aplicable a fraudes financieros y comercio electrónico, promoviendo transparencia, justicia y equidad en el manejo y análisis de los datos.

3.3. ¿Cuales son los retos de la información y los datos que utilizara en la base de datos en terminos de la calidad y la consolidación? (Max 100 Palabras) - (Primera entrega)

En el proyecto de base de datos para la detección y prevención de fraude con tarjetas de crédito y débito, enfrentamos varios retos relacionados con la calidad y consolidación de los datos. Uno de los principales desafíos es asegurar la calidad de los datos ingresados al sistema, lo que implica garantizar su precisión, completitud y relevancia para permitir un análisis efectivo. Esto es fundamental para evitar problemas como datos incompletos, inexactos o desactualizados, que podrían afectar negativamente la capacidad de detectar y prevenir fraudes eficientemente.

Además, la consolidación de datos provenientes de diversas fuentes presenta un desafío significativo.

Dado que las transacciones con tarjetas se pueden capturar a través de varios sistemas y entidades, es crucial unificar estos datos en un formato coherente y utilizable para realizar análisis integrales. Esto también implica mantener la integridad de los datos a lo largo de todo el proceso, desde la captura hasta el almacenamiento y el análisis, para evitar errores y manipulaciones que puedan comprometer las decisiones tomadas a partir de esta información.

Otro aspecto crítico es la latencia en la actualización y accesibilidad de los datos, que puede limitar la capacidad de responder a las amenazas de fraude en tiempo real. Por tanto, la base de datos debe ser optimizada para procesar y analizar datos rápidamente y de manera eficiente. Además, garantizar la seguridad de estos datos sensibles contra accesos no autorizados y ataques cibernéticos es esencial, lo que requiere medidas de seguridad robustas tanto para la información en reposo como en tránsito.

Finalmente, el proyecto debe cumplir con todas las regulaciones relevantes en materia de privacidad y protección de datos, lo que puede requerir ajustes y actualizaciones continuas para alinearse con las nuevas normativas y estándares de la industria. Asumir estos desafíos de manera estratégica es crucial para maximizar la efectividad del sistema en la lucha contra el fraude financiero.

3.4. ¿Que espera de la utilización de un sistema de Bases de Datos para su proyecto? (Max 100 Palabras) - (Primera entrega)

Desde la perspectiva del proyecto enfocado en la detección y prevención de fraude con tarjetas de crédito y débito, esperamos que la utilización de un sistema de bases de datos especializado aporta varias ventajas cruciales. Primero, esperamos que el sistema proporcione una plataforma robusta y segura para la captura y almacenamiento de grandes volúmenes de datos de transacciones financieras, facilitando el acceso rápido y eficiente a la información necesaria para el análisis en tiempo real. Esto es fundamental para identificar patrones y comportamientos sospechosos lo más rápido posible, lo cual es esencial para mitigar el impacto del fraude.

Además, la base de datos diseñada específicamente para este propósito deberá permitir la integración de diversos datos y sistemas, lo que mejorará la coherencia y consistencia de la información a través de diferentes plataformas y departamentos. Esta integración es clave para proporcionar una visión holística del comportamiento del usuario y las transacciones, permitiendo una detección de fraude más precisa y fundamentada.

También esperamos que el sistema de bases de datos facilite la implementación de herramientas de visualización y reporte que ayuden a comunicar los hallazgos a las partes interesadas de manera efectiva. Estos reportes no solo deben resaltar las transacciones sospechosas, sino también ofrecer insights que puedan ser utilizados para mejorar continuamente las estrategias de prevención de fraude.

Finalmente, confiamos en que la base de datos nos permita cumplir con los estándares de seguridad y privacidad requeridos, protegiendo los datos sensibles de los clientes y asegurando el cumplimiento de las regulaciones relevantes. Esto no solo protege a la institución de posibles sanciones legales, sino que también fortalece la confianza del cliente en nuestra capacidad para proteger su información financiera. En resumen, la base de datos es vista como una herramienta integral y estratégica que apoya directamente los objetivos de nuestro proyecto en la lucha contra el fraude.

4. Diseño del Modelo de Datos del SMBD (Sistema Manejador de Bases de Datos) *(Primera entrega)*

4.1. Características del SMBD (Sistema Manejador de Bases de Datos) para el proyecto *(Primera entrega)*

La selección de Oracle Cloud, Oracle Database Server y Oracle Data Modeler como nuestra plataforma de Sistema de Gestión de Bases de Datos (SGBD) fue por una serie de razones clave que se alinean con las metas de nuestro proyecto. Oracle Database Server se destaca por su capacidad para escalar eficientemente y manejar grandes volúmenes de información, una cualidad esencial para adaptarse al crecimiento esperado de nuestro proyecto. Este servidor garantiza una protección superior de los datos, mediante tecnologías de cifrado avanzadas y controles de acceso exhaustivos, elementos fundamentales para la seguridad de datos financieros y otros tipos de información sensible.

Oracle Data Modeler ofrece una solución eficaz para el diseño y la gestión de estructuras de bases de datos a través de una interfaz gráfica fácil de usar. Integrando esta herramienta con Oracle Cloud, se facilita una mayor flexibilidad en la implementación y una administración de recursos más ágil y eficiente. La plataforma está completamente alineada con los estándares de la industria y respaldada por un soporte técnico constante y herramientas analíticas de avanzada, lo que hace de Oracle una solución robusta y confiable para enfrentar las complejidades de nuestro entorno de datos, asegurando así un sistema eficaz para el manejo de la base de datos de finanzas.

4.2. Diagrama modelo de datos *(Primera entrega)*

Un modelo de datos es esencialmente un esquema conceptual que representa de manera abstracta la información gestionada en una base de datos utilizada en un sistema de información. Este modelo es clave para explicar cómo se estructura, almacena y enlaza la información dentro del sistema. Hay varios tipos de modelos de datos, y cada uno se especializa en mostrar diferentes facetas de la información y cómo se interrelacionan los datos entre sí. Estos modelos son herramientas cruciales para comprender y manejar la organización de datos en cualquier entorno informático.

4.2.1. Modelo Lógico:

Modelo Lógico: El modelo lógico de una base de datos es una representación más detallada de los datos que el modelo conceptual y está diseñado para ser implementado en un sistema de gestión de bases de datos específico. Se centra en describir los datos, sus relaciones y restricciones con mayor precisión, sin abordar los detalles a nivel de hardware.

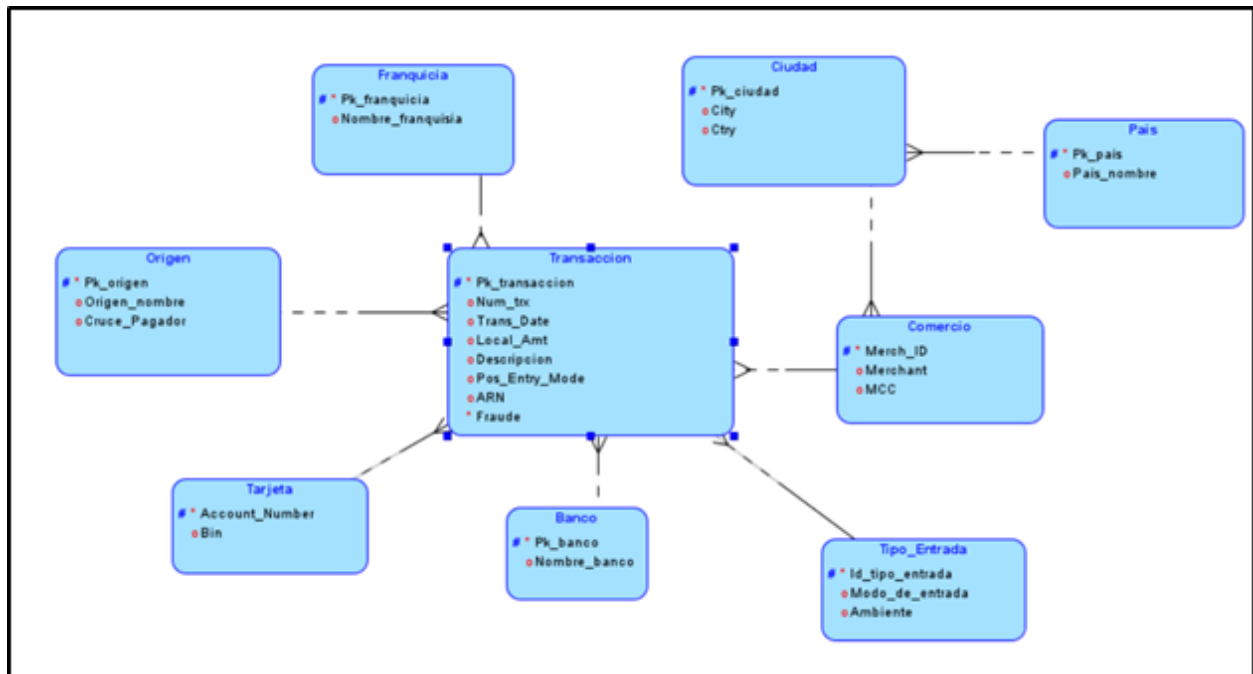
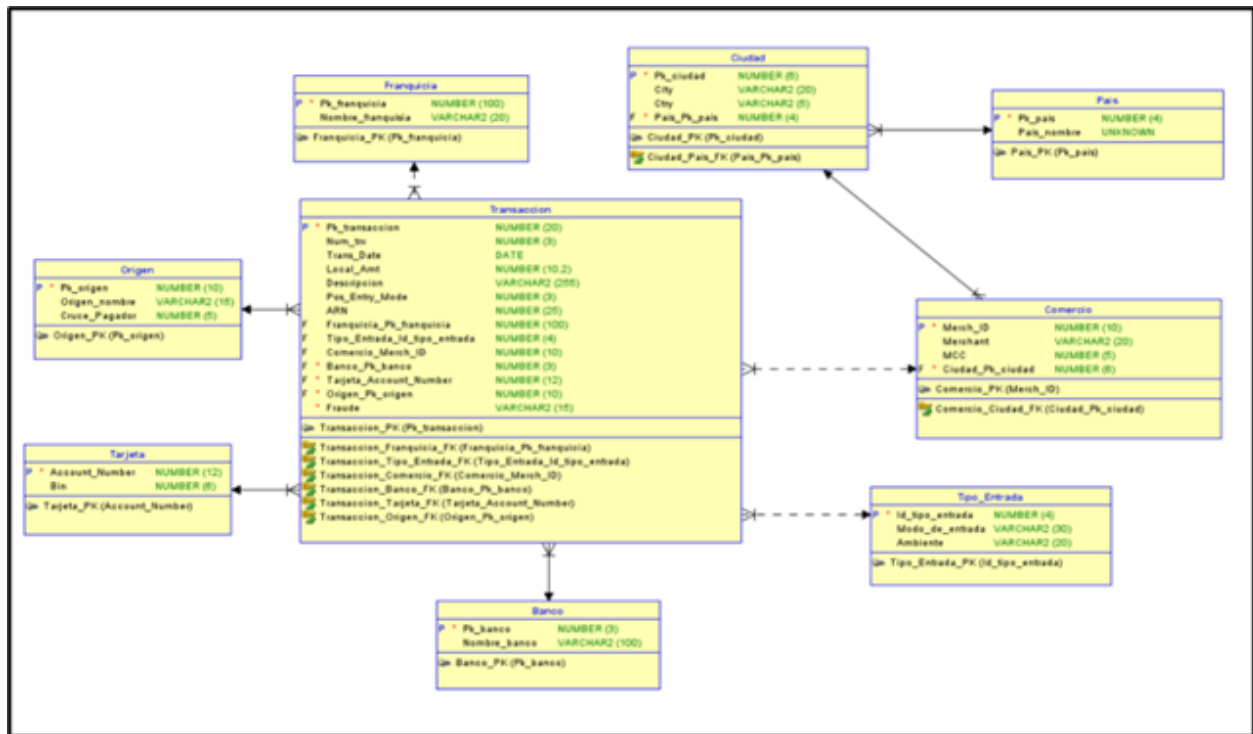


Figura 1: Modelo Logico de base de datos

4.2.2. Modelo Relacional:

Modelo Relacional: El modelo relacional en bases de datos es una estructura que organiza los datos en tablas, incluyendo tipos de datos, longitudes de campos y relaciones entre tablas. Cada tabla tiene una llave primaria que identifica de manera única cada fila en la tabla, y puede tener una o varias llaves foráneas que establecen relaciones con otras tablas. Este modelo permite evidenciar de manera más eficiente el uso de operaciones futuras basadas en el álgebra relacional, como selección, proyección y join.



4.3. Imágenes de la Base de Datos (Primera entrega)

■ Tabla Banco:

```
SELECT * FROM banco;
```

PK_BANCO	NOMBRE_BANCO
1	1 BANCO ITAU CORPBANCA COLOMBIA S.A.
2	2 CITIBANK N.A.
3	3 BARCLAYS BANK DELAWARE
4	4 BANCO DE GALICIA Y BUENOS AIRES, S.A.
5	5 PNC BANK, NATIONAL ASSOCIATION
6	6 CAPITAL ONE BANK (USA) NATIONAL ASSOCIATION
7	7 WELLS FARGO BANK, N.A.
8	8 THE BANK OF NEW YORK MELLON
9	9 BANCO SANTANDER (BRASIL) S.A.
10	10 HSBC BANK USA, N.A.

■ Tabla ciudad:

SELECT * FROM ciudad;

PK_CIUAD	CITY	CTRY	PAIS_PK_PAIS
1	1 BOGOTA	(null)	1
2	2 CARTAGENA	(null)	1
3	3 BARRANQUILLA	(null)	1
4	4 CALI	(null)	1
5	5 MEDELLIN	(null)	1
6	6 BUENOS AIRES	(null)	3
7	7 SANTIAGO	(null)	4
8	8 LIMA	(null)	6
9	9 QUITO	(null)	7
10	10 MONTEVIDEO	(null)	10

- Tabla comercio:

SELECT * FROM comercio;

MERCH_ID	MERCHANT	MCC	CIUDAD_PK_CIUAD
1	6 LA GLORIETA CALI	(null)	4
2	1 MERCADOS NOGALES BOGOTA	(null)	1
3	2 MADISSON INN HOTEL CARTAGENA	(null)	2
4	3 ARANA GOURMET BARRANQUILLA	(null)	3
5	4 DROGAS LA ECONOMIA N 43 BARRANQUILLA	(null)	3
6	5 HOTEL BH BARRANQUILLA	(null)	3
7	7 OPER REST Y EVENTOS CARTAGENA	(null)	2
8	8 SUPERMERCADO EL PORTAL MEDELLIN	(null)	5
9	9 TIENDAS MAX BUENOS AIRES	(null)	6
10	10 CENTRO COMERCIAL SANTIAGO	(null)	7

- Tabla franquicia:

SELECT * FROM franquicia;

Salida de Script x Resultado de la Consulta x

Todas las Filas Recuperadas: 10 en 0.011 segundos

PK_FRANQUICIA	NOMBRE_FRANQUICIA
1	1 MASTERCARD
2	2 VISA
3	3 AMERICAN EXPRESS
4	4 DINERS CLUB
5	5 DISCOVER
6	6 JCB
7	7 UNIONPAY
8	8 MAESTRO
9	9 ELO
10	10 MIR

- Tabla origen:

SELECT * FROM origen;

Salida de Script x Resultado de la Consulta x

Todas las Filas Recuperadas: 2 en 0.012 segundos

PK_ORIGEN	ORIGEN_NOMBRE	CRUCE_PAGADOR
1	1 Nacional	(null)
2	2 Internacional	(null)

- Tabla país:

`SELECT * FROM pais;`

	PK_PAIS	PAIS_NOMBRE
1	1	COLOMBIA
2	2	ESTADOS UNIDOS
3	3	ARGENTINA
4	4	CHILE
5	5	MEXICO
6	6	PERU
7	7	ECUADOR
8	8	BRASIL
9	9	VENEZUELA
10	10	URUGUAY

- Tabla tarjeta:

`SELECT * FROM tarjeta;`

	ACCOUNT_NUMBER	BIN
1	546687*****5851	546687
2	515735*****0608	515735
3	439707*****9985	439707
4	454640*****6756	454640
5	431196*****1095	431196
6	415417*****3464	415417
7	456317*****8800	456317
8	474476*****1234	474476
9	492940*****5678	492940
10	453978*****9101	453978

- Tabla tipo entrada:

SELECT * FROM tipo_entrada;

Salida de Script x Resultado de la Consulta x

Todas las Filas Recuperadas: 10 en 0.011 segundos

ID_TIPO_ENTRADA	MODULO_DE_ENTRADA	AMBIENTE
1	1 CHIP	PRESENTE
2	2 LECTURA COMPLETA DE BANDA	PRESENTE
3	3 COMPRA POR INTERNET	NO PRESENTE
4	4 CONTACTLESS	PRESENTE
5	5 MANUAL	PRESENTE
6	6 NFC	PRESENTE
7	7 QR CODE	NO PRESENTE
8	8 LECTURA FALLIDA	PRESENTE
9	9 TECLADO	PRESENTE
10	10 RECONOCIMIENTO FACIAL	NO PRESENTE

■ Tabla transacción:

SELECT * FROM transaccion;

Salida de Script x Resultado de la Consulta x

Todas las Filas Recuperadas: 10 en 0.011 segundos

ID_TRANSACCION	ID_USUARIO	ID_TIPO_ENTRADA	MODULO_DE_ENTRADA	AMBIENTE	FECHA	HORA	VALOR	MONEDA	ESTADO	COMENTARIO
1	1	1	1 CHIP	PRESENTE	2023-10-10	10:00:00	1000	USD	1	
2	2	2	2 LECTURA COMPLETA DE BANDA	PRESENTE	2023-10-10	10:01:00	2000	USD	2	
3	3	3	3 COMPRA POR INTERNET	NO PRESENTE	2023-10-10	10:02:00	3000	USD	3	
4	4	4	4 CONTACTLESS	PRESENTE	2023-10-10	10:03:00	4000	USD	4	
5	5	5	5 MANUAL	PRESENTE	2023-10-10	10:04:00	5000	USD	5	
6	6	6	6 NFC	PRESENTE	2023-10-10	10:05:00	6000	USD	6	
7	7	7	7 QR CODE	NO PRESENTE	2023-10-10	10:06:00	7000	USD	7	
8	8	8	8 LECTURA FALLIDA	PRESENTE	2023-10-10	10:07:00	8000	USD	8	
9	9	9	9 TECLADO	PRESENTE	2023-10-10	10:08:00	9000	USD	9	
10	10	10	10 RECONOCIMIENTO FACIAL	NO PRESENTE	2023-10-10	10:09:00	10000	USD	10	

4.4. Código SQL - lenguaje de definición de datos (DDL) (Primera entrega)

Creacion de tabla Banco

```
CREATE TABLE banco (
    pk_banco NUMBER(3) NOT NULL,
    nombre_banco VARCHAR2(100)
);
```

```
ALTER TABLE banco ADD CONSTRAINT banco_pk PRIMARY KEY (pk_banco);
```

Creacion de tabla Ciudad

```
CREATE TABLE ciudad (
    pk_ciudad NUMBER(6) NOT NULL,
    city VARCHAR2(20),
    ctry VARCHAR2(5),
    pais_pk_pais NUMBER(4) NOT NULL
);
```

```
ALTER TABLE ciudad ADD CONSTRAINT ciudad_pk PRIMARY KEY (pk_ciudad);
```

Creacion de tabla Comercio

```
CREATE TABLE comercio (  
    merch_id NUMBER(10) NOT NULL,  
    merchant VARCHAR2(20),  
    mcc NUMBER(5),  
    ciudad_pk_ciudad NUMBER(6) NOT NULL  
);
```

```
ALTER TABLE comercio ADD CONSTRAINT comercio_pk PRIMARY KEY (merch_id);
```

Creacion de tabla Franquicia

```
CREATE TABLE franquicia (  
    pk_franquicia NUMBER(10) NOT NULL,  
    nombre_franquicia VARCHAR2(20)  
);
```

```
ALTER TABLE franquicia ADD CONSTRAINT franquicia_pk  
PRIMARY KEY (pk_franquicia);
```

Creacion de tabla Origen

```
CREATE TABLE origen (  
    pk_origen NUMBER(10) NOT NULL,  
    origen_nombre VARCHAR2(15),  
    cruce_pagador NUMBER(5)  
);
```

```
ALTER TABLE origen ADD CONSTRAINT origen_pk PRIMARY KEY (pk_origen);
```

Creacion de tabla Pais

```
CREATE TABLE pais (  
    pk_pais NUMBER(4) NOT NULL,  
    pais_nombre VARCHAR2(20)  
);
```

```
ALTER TABLE pais ADD CONSTRAINT pais_pk PRIMARY KEY (pk_pais);
```

Creacion de tabla Tarjeta

```
CREATE TABLE tarjeta (  
    account_number NUMBER(12) NOT NULL,  
    bin NUMBER(6)  
);
```

```
ALTER TABLE tarjeta ADD CONSTRAINT tarjeta_pk  
PRIMARY KEY (account_number);
```

Creacion de tabla Tipo de entrada

```
CREATE TABLE tipo_entrada (  
    id_tipo_entrada NUMBER(4) NOT NULL,  
    modo_de_entrada VARCHAR2(30),  
    ambiente VARCHAR2(20)  
);
```

```
ALTER TABLE tipo_entrada ADD CONSTRAINT tipo_entrada_pk  
PRIMARY KEY (id_tipo_entrada);
```

Creacion de tabla Transaccion

```
CREATE TABLE transaccion (  
    pk_transaccion NUMBER(4) NOT NULL,  
    num_trx NUMBER(3),  
    trans_date DATE,  
    local_amt NUMBER(10, 2),  
    descripcion VARCHAR2(255),  
    pos_entry_mode NUMBER(3),  
    arn NUMBER(25),  
    franquicia_pk_franquicia NUMBER(28),  
    tipo_entrada_id_tipo_entrada NUMBER(4),  
    comercio_merch_id NUMBER(10),  
    banco_pk_banco NUMBER(3) NOT NULL,  
    tarjeta_account_number NUMBER(12) NOT NULL,  
    origen_pk_origen NUMBER(10) NOT NULL,  
    fraude VARCHAR2(15) NOT NULL  
);
```

```
ALTER TABLE transaccion ADD CONSTRAINT transaccion_pk  
PRIMARY KEY (pk_transaccion);
```

Creacion de llaves foraneas

```
ALTER TABLE ciudad  
ADD CONSTRAINT ciudad_pais_fk FOREIGN KEY (ciudad_pk_pais)  
REFERENCES pais (pk_pais);
```

```
ALTER TABLE comercio  
ADD CONSTRAINT comercio_ciudad_fk FOREIGN KEY (ciudad_pk_ciudad)  
REFERENCES ciudad (pk_ciudad);
```

```
ALTER TABLE transaccion  
ADD CONSTRAINT transaccion_banco_fk FOREIGN KEY (banco_pk_banco)
```



```

REFERENCES banco (pk_banco);

ALTER TABLE transaccion
  ADD CONSTRAINT transaccion_comercio_fk
  FOREIGN KEY (comercio_merch_id) REFERENCES comercio (merch_id);

ALTER TABLE transaccion
  ADD CONSTRAINT transaccion_franquicia_fk
  FOREIGN KEY (franquicia_pk_franquicia)
  REFERENCES franquicia (pk_franquicia);

ALTER TABLE transaccion
  ADD CONSTRAINT transaccion_origen_fk
  FOREIGN KEY (origen_pk_origen) REFERENCES origen (pk_origen);

ALTER TABLE transaccion
  ADD CONSTRAINT transaccion_tarjeta_fk
  FOREIGN KEY (tarjeta_account_number) REFERENCES tarjeta (account_number);

ALTER TABLE transaccion
  ADD CONSTRAINT transaccion_tipo_entrada_fk
  FOREIGN KEY (tipo_entrada_id_tipo_entrada)
  REFERENCES tipo_entrada (id_tipo_entrada);

```

4.5. Código SQL - Manipulación de datos (DML) *(Primera entrega)*

Insercion datos tabla Banco

```

INSERT INTO Banco (pk_banco, nombre_banco)
  VALUES (1, 'BANCO-ITAU-CORPBANCA-COLOMBIA-S.A. ');
INSERT INTO Banco (pk_banco, nombre_banco)
  VALUES (2, 'CITIBANK-N.A. ');
INSERT INTO Banco (pk_banco, nombre_banco)
  VALUES (3, 'BARCLAYS-BANK-DELAWARE');
INSERT INTO Banco (pk_banco, nombre_banco)
  VALUES (4, 'BANCO-DE-GALICIA-Y-BUENOS-AIRES-S.A. ');
INSERT INTO Banco (pk_banco, nombre_banco)
  VALUES (5, 'PNC-BANK, -NATIONAL-ASSOCIATION');
INSERT INTO Banco (pk_banco, nombre_banco)
  VALUES (6, 'CAPITAL-ONE-BANK-(USA)-NATIONAL-ASSOCIATION');
INSERT INTO Banco (pk_banco, nombre_banco)
  VALUES (7, 'WELLS-FARGO-BANK, -N.A. ');
INSERT INTO Banco (pk_banco, nombre_banco)
  VALUES (8, 'THE-BANK-OF-NEW-YORK-MELLON');
INSERT INTO Banco (pk_banco, nombre_banco)
  VALUES (9, 'BANCO-SANTANDER-(BRASIL)-S.A. ');
INSERT INTO Banco (pk_banco, nombre_banco)
  VALUES (10, 'HSBC-BANK-USA, -N.A. ');

```

Insercion datos tabla Pais

```
INSERT INTO Pais (pk_pais, pais_nombre) VALUES (1, 'COLOMBIA');
INSERT INTO Pais (pk_pais, pais_nombre) VALUES (2, 'ESTADOS-UNIDOS');
INSERT INTO Pais (pk_pais, pais_nombre) VALUES (3, 'ARGENTINA');
INSERT INTO Pais (pk_pais, pais_nombre) VALUES (4, 'CHILE');
INSERT INTO Pais (pk_pais, pais_nombre) VALUES (5, 'MEXICO');
INSERT INTO Pais (pk_pais, pais_nombre) VALUES (6, 'PERU');
INSERT INTO Pais (pk_pais, pais_nombre) VALUES (7, 'ECUADOR');
INSERT INTO Pais (pk_pais, pais_nombre) VALUES (8, 'BRASIL');
INSERT INTO Pais (pk_pais, pais_nombre) VALUES (9, 'VENEZUELA');
INSERT INTO Pais (pk_pais, pais_nombre) VALUES (10, 'URUGUAY');
```

Insercion datos tabla Ciudad

```
INSERT INTO Ciudad (PK_ciudad, City, Pais_PK_Pais)
VALUES (1, 'BOGOTA', 1);
INSERT INTO Ciudad (PK_ciudad, City, Pais_PK_Pais)
VALUES (2, 'CARTAGENA', 1);
INSERT INTO Ciudad (PK_ciudad, City, Pais_PK_Pais)
VALUES (3, 'BARRANQUILLA', 1);
INSERT INTO Ciudad (PK_ciudad, City, Pais_PK_Pais)
VALUES (4, 'CALI', 1);
INSERT INTO Ciudad (PK_ciudad, City, Pais_PK_Pais)
VALUES (5, 'MEDELLIN', 1);
INSERT INTO Ciudad (PK_ciudad, City, Pais_PK_Pais)
VALUES (6, 'BUENOS-AIRES', 3);
INSERT INTO Ciudad (PK_ciudad, City, Pais_PK_Pais)
VALUES (7, 'SANTIAGO', 4);
INSERT INTO Ciudad (PK_ciudad, City, Pais_PK_Pais)
VALUES (8, 'LIMA', 6);
INSERT INTO Ciudad (PK_ciudad, City, Pais_PK_Pais)
VALUES (9, 'QUITO', 7);
INSERT INTO Ciudad (PK_ciudad, City, Pais_PK_Pais)
VALUES (10, 'MONTEVIDEO', 10);
```

Insercion datos tabla Comercio

```
INSERT INTO Comercio (Merch_ID, Merchant, Ciudad_PK_Ciudad)
VALUES (1, 'MERCADOS-INGLES-BOGOTA', 1);
INSERT INTO Comercio (Merch_ID, Merchant, Ciudad_PK_Ciudad)
VALUES (2, 'NORDSTROM-HOTEL-CARTAGENA', 2);
INSERT INTO Comercio (Merch_ID, Merchant, Ciudad_PK_Ciudad)
VALUES (3, 'DROGAS-LA-ECONOMIA-BARRANQUILLA', 3);
INSERT INTO Comercio (Merch_ID, Merchant, Ciudad_PK_Ciudad)
VALUES (4, 'RAPPI-GOURMET-BARRANQUILLA', 3);
INSERT INTO Comercio (Merch_ID, Merchant, Ciudad_PK_Ciudad)
VALUES (5, 'HOTEL-DEL-BARRANQUILLA', 3);
INSERT INTO Comercio (Merch_ID, Merchant, Ciudad_PK_Ciudad)
```

```

VALUES (6, 'LA-GORRERIA-CALI', 4);
INSERT INTO Comercio (Merch_ID, Merchant, Ciudad_PK_Ciudad)
VALUES (7, 'OPEN-MARKET-EVENTOS-CARTAGENA', 2);
INSERT INTO Comercio (Merch_ID, Merchant, Ciudad_PK_Ciudad)
VALUES (8, 'SUPERMERCADO-EL-PORTAL-MEDELLIN', 5);
INSERT INTO Comercio (Merch_ID, Merchant, Ciudad_PK_Ciudad)
VALUES (9, 'TIENDAS-MAX-BUENOS-AIRES', 6);
INSERT INTO Comercio (Merch_ID, Merchant, Ciudad_PK_Ciudad)
VALUES (10, 'CENTRO-COMERCIAL-SANTIAGO', 7);

```

Insercion datos tabla Tipo de entrada

```

INSERT INTO Tipo_Entrada (id_tipo_entrada, modo_de_entrada, ambiente)
VALUES (1, 'CHIP', 'PRESENTE');
INSERT INTO Tipo_Entrada (id_tipo_entrada, modo_de_entrada, ambiente)
VALUES (2, 'LECTURA-COMPLETA-DE-BANDA', 'PRESENTE');
INSERT INTO Tipo_Entrada (id_tipo_entrada, modo_de_entrada, ambiente)
VALUES (3, 'COMPRA-POR-INTERNET', 'NO-PRESENTE');
INSERT INTO Tipo_Entrada (id_tipo_entrada, modo_de_entrada, ambiente)
VALUES (4, 'CONTACTLESS', 'PRESENTE');
INSERT INTO Tipo_Entrada (id_tipo_entrada, modo_de_entrada, ambiente)
VALUES (5, 'MANUAL', 'PRESENTE');
INSERT INTO Tipo_Entrada (id_tipo_entrada, modo_de_entrada, ambiente)
VALUES (6, 'NFC', 'PRESENTE');
INSERT INTO Tipo_Entrada (id_tipo_entrada, modo_de_entrada, ambiente)
VALUES (7, 'QR-CODE', 'NO-PRESENTE');
INSERT INTO Tipo_Entrada (id_tipo_entrada, modo_de_entrada, ambiente)
VALUES (8, 'LECTURA-FALLIDA', 'PRESENTE');
INSERT INTO Tipo_Entrada (id_tipo_entrada, modo_de_entrada, ambiente)
VALUES (9, 'TECLADO', 'PRESENTE');
INSERT INTO Tipo_Entrada (id_tipo_entrada, modo_de_entrada, ambiente)
VALUES (10, 'RECONOCIMIENTO-FACIAL', 'NO-PRESENTE');

```

Insercion datos tabla Tarjeta

```

INSERT INTO Tarjeta (Account_Number, Bin) VALUES (54667000000000551, 546670);
INSERT INTO Tarjeta (Account_Number, Bin) VALUES (51753500000000608, 517535);
INSERT INTO Tarjeta (Account_Number, Bin) VALUES (43970000000009955, 439700);
INSERT INTO Tarjeta (Account_Number, Bin) VALUES (45640000000006756, 456400);
INSERT INTO Tarjeta (Account_Number, Bin) VALUES (43119600000001095, 431196);
INSERT INTO Tarjeta (Account_Number, Bin) VALUES (45117000000003464, 451170);
INSERT INTO Tarjeta (Account_Number, Bin) VALUES (45631700000000800, 456317);
INSERT INTO Tarjeta (Account_Number, Bin) VALUES (47447600000001234, 474476);
INSERT INTO Tarjeta (Account_Number, Bin) VALUES (49294000000005678, 492940);
INSERT INTO Tarjeta (Account_Number, Bin) VALUES (45397800000009101, 453978);

```

Insercion datos tabla Origen

```

INSERT INTO Origen (PK_origen, Origen_nombre) VALUES (1, 'Nacional');

```

```
INSERT INTO Origen (PK_origen, Origen_nombre) VALUES (2, 'Internacional');
```

Insercion datos tabla Franquicia

```
INSERT INTO Franquicia (PK_franquicia, Nombre_franquicia)
VALUES (1, 'MASTERCARD');
INSERT INTO Franquicia (PK_franquicia, Nombre_franquicia)
VALUES (2, 'VISA');
INSERT INTO Franquicia (PK_franquicia, Nombre_franquicia)
VALUES (3, 'AMERICAN-EXPRESS');
INSERT INTO Franquicia (PK_franquicia, Nombre_franquicia)
VALUES (4, 'DINERS-CLUB');
INSERT INTO Franquicia (PK_franquicia, Nombre_franquicia)
VALUES (5, 'DISCOVERY');
INSERT INTO Franquicia (PK_franquicia, Nombre_franquicia)
VALUES (6, 'JCB');
INSERT INTO Franquicia (PK_franquicia, Nombre_franquicia)
VALUES (7, 'UNIONPAY');
INSERT INTO Franquicia (PK_franquicia, Nombre_franquicia)
VALUES (8, 'MAESTRO');
INSERT INTO Franquicia (PK_franquicia, Nombre_franquicia)
VALUES (9, 'ELO');
INSERT INTO Franquicia (PK_franquicia, Nombre_franquicia)
VALUES (10, 'MIR');
```

Insercion datos tabla Transaccion

```
INSERT INTO Transaccion (pk-transaccion, num_trx, trans_date, local_amt,
descripcion, pos_entry_mode, arn, franquicia_pk_franquicia,
tipo_entrada_id.tipo_entrada, comercio_merch_id, banco_pk_banco,
tarjeta_account_number, origen_pk_origen, fraude)
VALUES (1, 1, TO_DATE('2018-01-02', 'YYYY-MM-DD'), 59200.00,
'MERCADOS-NOGALES-BOGOTA', 5, 85426468002153806263522,
1, 1, 1, 1, 546687, 1, 'No');
```

```
INSERT INTO Transaccion (pk-transaccion, num_trx, trans_date, local_amt,
descripcion, pos_entry_mode, arn, franquicia_pk_franquicia,
tipo_entrada_id.tipo_entrada, comercio_merch_id, banco_pk_banco,
tarjeta_account_number, origen_pk_origen, fraude)
VALUES (2, 2, TO_DATE('2018-01-02', 'YYYY-MM-DD'), 56250.00,
'MERCADOS-NOGALES-BOGOTA', 5, 85426468002153420245525,
1, 1, 1, 1, 546687, 1, 'No');
```

```
INSERT INTO Transaccion (pk-transaccion, num_trx, trans_date, local_amt,
descripcion, pos_entry_mode, arn, franquicia_pk_franquicia,
tipo_entrada_id.tipo_entrada, comercio_merch_id, banco_pk_banco,
tarjeta_account_number, origen_pk_origen, fraude)
VALUES (3, 1, TO_DATE('2018-01-02', 'YYYY-MM-DD'), 2856702.56,
'MADISSON-INN-HOTEL-CARTAGENA', 90, 85426468002221005156704,
```

```

1, 1, 2, 2, 515735, 2, 'No');

INSERT INTO Transaccion (pk_transaccion,num_trx,trans_date,local_amt,
    descripcion,pos_entry_mode,arn,franquicia_pk_franquicia,
    tipo_entrada_id_tipo_entrada,comercio_merch_id,banco_pk_banco,
    tarjeta_account_number,origen_pk_origen,fraude)
VALUES (4, 1, TO_DATE('2018-01-02', 'YYYY-MM-DD'), 70000.00,
    'ARANA-GOURMETH-BARRANQUILLA', 90, 74541008003003130508141,
    2, 1, 3, 3, 439707, 2, 'No');

INSERT INTO Transaccion (pk_transaccion,num_trx,trans_date,local_amt,
    descripcion,pos_entry_mode,arn,franquicia_pk_franquicia,
    tipo_entrada_id_tipo_entrada,comercio_merch_id,banco_pk_banco,
    tarjeta_account_number,origen_pk_origen,fraude)
VALUES (5, 2, TO_DATE('2018-01-02', 'YYYY-MM-DD'), 141130.00,
    'DROGAS-LA-ECONOMIA-N-43-BARRANQUILLA', 90, 74541008004004130507678,
    2, 1, 3, 3, 439707, 2, 'No');

INSERT INTO Transaccion (pk_transaccion,num_trx,trans_date,local_amt,
    descripcion,pos_entry_mode,arn,franquicia_pk_franquicia,
    tipo_entrada_id_tipo_entrada,comercio_merch_id,banco_pk_banco,
    tarjeta_account_number,origen_pk_origen,fraude)
VALUES (6, 1, TO_DATE('2018-01-02', 'YYYY-MM-DD'), 140000.00,
    'HOTEL-BH-BARRANQUILLA', 1, 74541008003003130519312,
    2, 1, 3, 4, 454640, 2, 'No');

INSERT INTO Transaccion (pk_transaccion,num_trx,trans_date,local_amt,
    descripcion,pos_entry_mode,arn,franquicia_pk_franquicia,
    tipo_entrada_id_tipo_entrada,comercio_merch_id,banco_pk_banco,
    tarjeta_account_number,origen_pk_origen,fraude)
VALUES (7, 2, TO_DATE('2018-01-02', 'YYYY-MM-DD'), 193200.00,
    'HOTEL-BH-BARRANQUILLA', 1, 74541008004004130536149,
    2, 1, 3, 4, 454640, 2, 'No');

INSERT INTO Transaccion (pk_transaccion,num_trx,trans_date,local_amt,
    descripcion,pos_entry_mode,arn,franquicia_pk_franquicia,
    tipo_entrada_id_tipo_entrada,comercio_merch_id,banco_pk_banco,
    tarjeta_account_number,origen_pk_origen,fraude)
VALUES (8, 1, TO_DATE('2018-01-02', 'YYYY-MM-DD'), 60000.00,
    'LA-GLORIETA-CALI', 5, 74541008003003130506558,
    2, 1, 4, 5, 431196, 1, 'No');

INSERT INTO Transaccion (pk_transaccion,num_trx,trans_date,local_amt,
    descripcion,pos_entry_mode,arn,franquicia_pk_franquicia,
    tipo_entrada_id_tipo_entrada,comercio_merch_id,banco_pk_banco,
    tarjeta_account_number,origen_pk_origen,fraude)
VALUES (9, 1, TO_DATE('2018-01-02', 'YYYY-MM-DD'), 1663265.00,
    'OPER-REST-Y-EVENTOS-CARTAGENA', 5, 74541008003003130504504,
    2, 1, 2, 6, 415417, 2, 'No');

INSERT INTO Transaccion (pk_transaccion,num_trx,trans_date,local_amt,

```

```

descripcion , pos_entry_mode , arn , franquicia_pk_franquicia ,
tipo_entrada_id_tipo_entrada , comercio_merch_id , banco_pk_banco ,
tarjeta_account_number , origen_pk_origen , fraude)
VALUES (10, 2, TO_DATE('2018-01-02', 'YYYY-MM-DD'), 184658.00,
'OPER-REST-Y-EVENTOS-CARTAGENA', 5, 74541008003003130504496,
2, 1, 2, 6, 415417, 2, 'No');

```

4.6. Código SQL + Resultados: Vistas (Primera entrega)

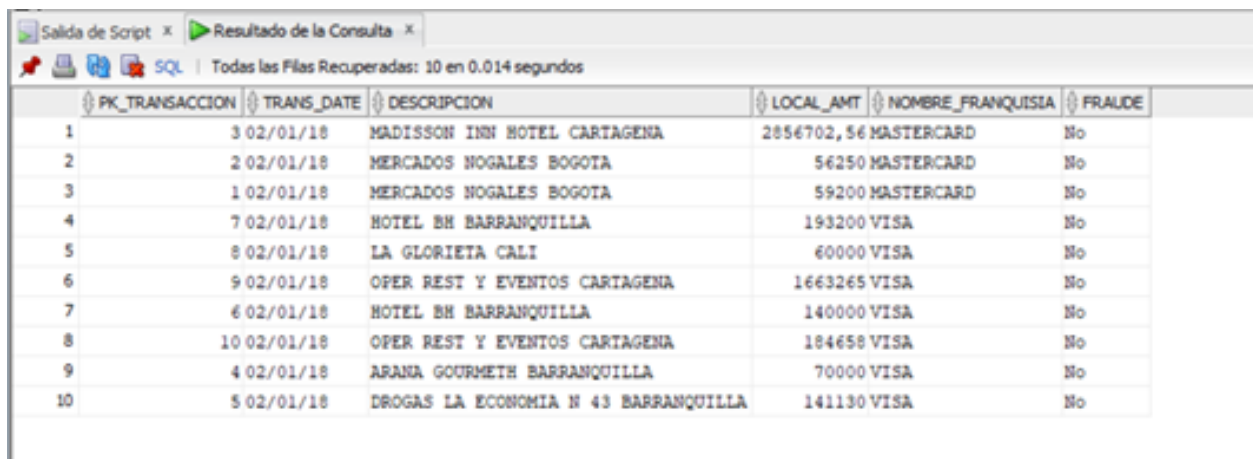
Una vista para listar todas las transacciones con detalles sobre la tarjeta, la franquicia y si fue reportada como fraude o no.

```

CREATE OR REPLACE FORCE EDITIONABLE VIEW VistaTransaccionesDetalle AS
SELECT
    t.pk_transaccion ,
    t.trans_date ,
    t.descripcion ,
    t.local_amt ,
    f.nombre_franquicia ,
    t.fraude
FROM
    transaccion t
JOIN
    franquicia f ON t.franquicia_pk_franquicia = f.PK_franquicia;

SELECT * FROM VistaTransaccionesDetalle;

```



Salida de Script x Resultado de la Consulta x

Todas las Filas Recuperadas: 10 en 0.014 segundos

	PK_TRANSACCION	TRANS_DATE	DESCRIPCION	LOCAL_AMT	NOMBRE_FRANQUISIA	FRAUDE
1	3	02/01/18	MADISSON INN HOTEL CARTAGENA	2856702,56	MASTERCARD	No
2	2	02/01/18	MERCADOS NOGALES BOGOTA	56250	MASTERCARD	No
3	1	02/01/18	MERCADOS NOGALES BOGOTA	59200	MASTERCARD	No
4	7	02/01/18	HOTEL BH BARRANQUILLA	193200	VISA	No
5	8	02/01/18	LA GLORIETA CALI	60000	VISA	No
6	9	02/01/18	OPER REST Y EVENTOS CARTAGENA	1663265	VISA	No
7	6	02/01/18	HOTEL BH BARRANQUILLA	140000	VISA	No
8	10	02/01/18	OPER REST Y EVENTOS CARTAGENA	184658	VISA	No
9	4	02/01/18	ARANA GOURMETH BARRANQUILLA	70000	VISA	No
10	5	02/01/18	DROGAS LA ECONOMIA N 43 BARRANQUILLA	141130	VISA	No

4.7. Código SQL + Resultados: Triggers (Primera entrega)

Un trigger que actualice un campo en la tabla transaccion para marcar la transacción como revisada por fraude cada vez que se inserte una nueva transacción.

```

CREATE OR REPLACE TRIGGER TriggerRevisarFraude
BEFORE INSERT ON transaccion

```

```

FOR EACH ROW
BEGIN
    :NEW.fraude := 'En-revision';
END;

INSERT INTO Transaccion (pk-transaccion,num-trx,trans_date,local_amt,
    descripcion,pos_entry_mode,arn,franquicia_pk-franquicia,
    tipo-entrada_id-tipo-entrada,comercio_merch_id,banco_pk-banco,
    tarjeta_account_number,origen_pk-origen,fraude)
VALUES (11, 1, TO_DATE('2018-01-02', 'YYYY-MM-DD'), 59200.00,
    'MERCADOS-NOGALES-BOGOTA', 5, 85426468002153806263522,
    1, 1, 1, 1, 546687, 1, 'No');

SELECT * FROM transaccion where pk-transaccion = 11;

```

TRANS_DATE	LOCAL_AMT	DESCRIPCION	POS_ENTRY_MODE	ARN	FRANQUICIA_PK_FRANQUICIA	TIPO_ENTRADA_ID TIPO_ENTRADA	COMERCIO_MERCH_ID	BANCO_PK_BANCO	TARJETA_ACCOUNT_NUMBER	ORIGEN_PK_ORIGEN	FRAUDE
1/02/18	59200	MERCADOS NOGALES BOGOTA	5	85426468002153806263522	1	1	1	1	546687	1	En revision

4.8. Código SQL + Resultados: Funciones (*Primera entrega*)

Una función que calcule el monto total de las transacciones para un comercio específico en un día determinado.

```

create or replace FUNCTION
    TotalVentasDia(comercio_id IN NUMBER, fecha IN DATE)
RETURN NUMBER IS
    total_ventas NUMBER(10,2);
BEGIN
    SELECT NVL(SUM(local_amt), 0) INTO total_ventas
    FROM transaccion
    WHERE comercio_merch_id = comercio_id AND TRUNC(trans_date) = TRUNC(fecha);

    RETURN total_ventas;
END;
/

-- Ejecutar la funcion
DECLARE
    resultado NUMBER;
BEGIN
    resultado := TotalVentasDia(1, '02-01-2018');
    dbms_output.put_line('Totla-ventas-' || resultado);
END;

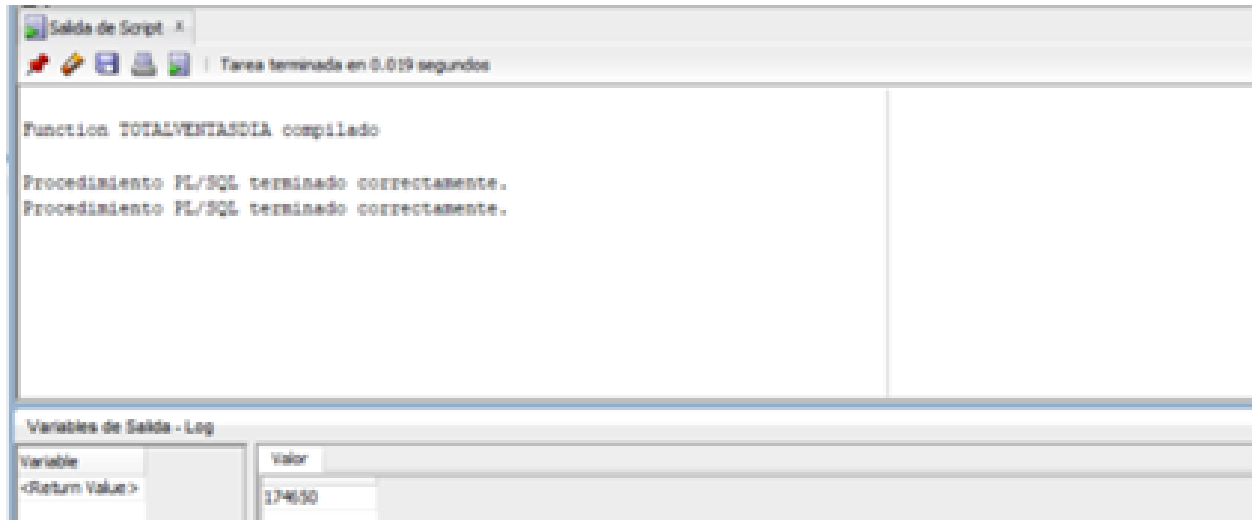
--Ejecucion
DECLARE

```

```

resultado NUMBER;
BEGIN
    resultado := TotalVentasDia(1, '02-01-2018');
    dbms_output.put_line('Total-ventas-' || resultado);
END;

```



4.9. Código SQL + Resultados: procedimientos almacenados (*Primera entrega*)

Este procedimiento aceptará el ID del comercio y la fecha como entradas, ejecutará la función previamente definida y luego imprimirá el resultado para una mejor vista al usuario

```

create or replace PROCEDURE MostrarTotalVentasDia
    (comercio_id IN NUMBER, fecha IN VARCHAR2) AS
    total NUMBER;
    fecha_convertida DATE;
BEGIN
    — Convertir la entrada de texto a fecha
    fecha_convertida := TO_DATE(fecha, 'YYYY-MM-DD');
    — Obtener el total utilizando la funcion
    total := TotalVentasDia(comercio_id, fecha_convertida);
    — Imprimir el resultado
    DBMS_OUTPUT.PUT_LINE(
        'Total-de-ventas-para-el-comercio-' || comercio_id || '-en-' ||
        TO_CHAR(fecha_convertida, 'DD-MON-YYYY') || '-es:-' || total);
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Error:-' || SQLERRM);
END;
/

EXECUTE MostrarTotalVentasDia(1, '2018-01-02')

```




Un procedimiento para insertar una nueva transacción, asegurándose de que todos los datos necesarios estén presentes y sean válidos.

```

create or replace PROCEDURE InsertarTransaccion (
    p_pk_transaccion IN transaccion.pk_transaccion%TYPE,
    p_num_trx IN transaccion.num_trx%TYPE,
    p_trans_date IN transaccion.trans_date%TYPE,
    p_local_amt IN transaccion.local_amt%TYPE,
    p_descripcion IN transaccion.descripcion%TYPE,
    p_pos_entry_mode IN transaccion.pos_entry_mode%TYPE,
    p_arn IN transaccion.arn%TYPE,
    p_franquicia_pk IN transaccion.franquicia_pk_franquicia%TYPE,
    p_tipo_entrada_id IN transaccion.tipo_entrada_id_tipo_entrada%TYPE,
    p_comercio_merch_id IN transaccion.comercio_merch_id%TYPE,
    p_banco_pk IN transaccion.banco_pk_banco%TYPE,
    p_tarjeta_num IN transaccion.tarjeta_account_number%TYPE,
    p_origen_pk IN transaccion.origen_pk_origen%TYPE,
    p_fraude IN transaccion.fraude%TYPE
) AS
BEGIN
    INSERT INTO transaccion (
        pk_transaccion , num_trx , trans_date , local_amt , descripcion , pos_entry_mode ,
        arn , franquicia_pk_franquicia , tipo_entrada_id_tipo_entrada ,
        comercio_merch_id , banco_pk_banco , tarjeta_account_number ,
        origen_pk_origen , fraude
    ) VALUES (
        p_pk_transaccion , p_num_trx , p_trans_date , p_local_amt , p_descripcion ,
        p_pos_entry_mode , p_arn , p_franquicia_pk , p_tipo_entrada_id ,
        p_comercio_merch_id , p_banco_pk , p_tarjeta_num , p_origen_pk , p_fraude
    );

    COMMIT;
EXCEPTION
    WHEN OTHERS THEN
        ROLLBACK;
        RAISE;
END;

```

5. Bases de Datos No-SQL (*Segunda entrega*)

5.1. Diagrama Bases de Datos No-SQL (*Segunda entrega*)

5.2. SMBD utilizado para la Base de Datos No-SQL (*Segunda entrega*)

6. Aplicación de ETL (Extract, Transform, Load) y Bodega de Datos (*Tercera entrega*)

6.1. Ejemplo de aplicación de ETL y Bodega de Datos (*Tercera entrega*)

6.2. Automatización de Datos (*Tercera entrega*)

6.3. Integración de Datos (*Tercera entrega*)

7. Proximos pasos (*Tercera entrega*)

8. Lecciones aprendidas (*Tercera entrega*)

9. Bibliografía