**Analítica de Grandes Datos**

**Departamento de Ciencias de la Computación y la Decisión**

**Facultad de Minas**

**Universidad Nacional de Colombia**

Trabajo Nociones de Arquitectura de la Información

Versión: 2021.05.17. 16:47

*Observación: Cada vez que agregue nuevos elementos al documento, o que modifique algún componente del informe, revise la coherencia y consistencia con los otros elementos que hacen parte del mismo.*

**Responsables**

|  |
| --- |
| Nombre Completo – Documento de Identificación |
| 1. Julian Rodriguez de la Hoz - 1.128.282.329 |
| 2. Johnatan Andrés Salazar Giraldo – 1.040.041.335 |
| 3. Natalia Aguinaga Restrepo - 1.017.124.727 |
| 4. Jenifer Rios Sierra- |
| 5.Jhon Fredy Isaza Trujillo- |
| **REPO EN GITHUB: https://github.com/joasalazargi/TrabajoAGD202101** |

*Realiza este trabajo considerando los datos que generan los sistemas transaccionales e información no estructurada de tu dominio (si trabajas por ejemplo para TCC tu dominio es la mensajería; también puedes explorar en la página* [*https://www.kaggle.com/datasets*](https://www.kaggle.com/datasets) *o* [*https://arxiv.org/*](https://arxiv.org/)*). Considera tener acceso a esta información, de al menos 10 MB (puede ser uno o varios archivos de texto), y* ***tener al menos cuatro clases conceptuales. Este documento también debe almacenarse en el REPO. Plazo Máximo de Entrega 23 de Mayo, NO SE recibirá por correo electrónico, envío por*** [***https://forms.gle/h7ty3yZykaUq5m7y6***](https://forms.gle/h7ty3yZykaUq5m7y6)

1. **Comprensión del negocio**
   1. **Descripción del contexto del negocio.**

Describa en máximo 250 palabras el contexto en el cual se generan los datos y cuál es el proceso que los genera.

Olist es la tienda por departamentos más grande del mercado brasilero. Esta tienda electrónica actúa como un tercero que conecta a las pequeñas empresas con los canales, sin la relación de un contrato. Los vendedores pueden ofrecer sus productos y enviarlos directamente a los clientes interesados utilizando los socios logísticos de Olist.

A partir de las transacciones de compra/venta realizadas en el sitio de Olist, se cuenta con información asociada a clientes, vendedores, compras, productos, medios de pago, tiempos de entrega y evaluación de órdenes. Esta información es almacenada en bases de datos y constituye un insumo de gran importancia para la evaluación de las tendencias de compra, medición en la satisfacción de los clientes, estadísticas descriptivas sobre el comportamiento de las ventas, entre otros.

(Adjuntar link)

* 1. **Identificación del problema:**

Delimite en máximo 150 palabras la problemática, así como identificar los requisitos, supuestos, restricciones y beneficios de la solución de este.

Los distintos departamentos de la tienda Olist se han propuesto mejorar la experiencia de compra del usuario, las cuales impactan en las dinámicas de ventas de los proveedores y de manera directa con la rentabilidad esperada para la compañía.

Por esta razón, ellos requieren realizar un análisis de sus transacciones, con el fin de implementar estrategias comerciales que permitan a sus aliados comerciales mejorar el rendimiento en términos de niveles de ventas, atracción y fidelización de clientes.

Las órdenes de compra siempre son entregadas y distribuidas por un solo proveedor de logística.

* 1. **Determinación de objetivos:**

Describa en máximo 150 palabras las metas a lograr al proponer una solución basada en un modelo de datos o de analítica (cómo y qué tipo de ventaja competitiva se ganará).

Identificar si hay demoras en la entrega de los productos, y en caso de existir retrasos, determinar si éstos poseen correlación con la localización del cliente. Esto permitirá enfocar la estrategia de Olist, de aumentar la cobertura en la ciudad, ya sea con una nueva sede o/y un cambio de proveedor logístico (distribución o almacenaje).

Analizar las principales causas por las cuales una orden no fue finalizada con éxito en la tienda de Olist. Así se podrá identificar la posibilidad de establecer convenios con otros bancos o la necesidad de implementar mejoras en el sistemas de control de inventarios.

Necesidad de ampliar el mercado, por ello quieren fortalecer el actual, por medio de la fidelización de clientes.

* 1. **Evaluación de la situación actual:**

Describa en máximo 150 palabras el estado actual antes de implementar la solución de analítica, a fin de tener un punto de comparación que permita medir el grado de éxito de la solución.

Actualmente la tienda virtual de comercio Olist cuenta con X departamentos distribuidos en Brasil, los cuales ofrecen productos tales como x, y y z.

Las ventas son realizadas por distintos vendedores, quienes asesoran a los clientes en sus necesidades, y los acompañan en el proceso de compra.

Metricas de circulación y permanencia en la página

1. **Compression de los datos**
   1. **recolección de datos**

Describa en máximo 150 palabras los datos a utilizar identificando las fuentes, las técnicas empleadas en su recolección, los problemas encontrados en su obtención y la forma como se resolvieron los mismos. Además, adjunte los datos (archivos de texto, etc.) agréguelos en el github **(REPO EN GITHUB)** en un solo archivo, por favor comprímalo(s). Llame el archivo T1.2.1.Datos.zip

https://www.kaggle.com/olistbr/brazilian-ecommerce

* 1. **Descripción de datos (diccionario) (Julian):**

Diligencia la siguiente tabla, puede agregar otra columna si lo considera necesario.

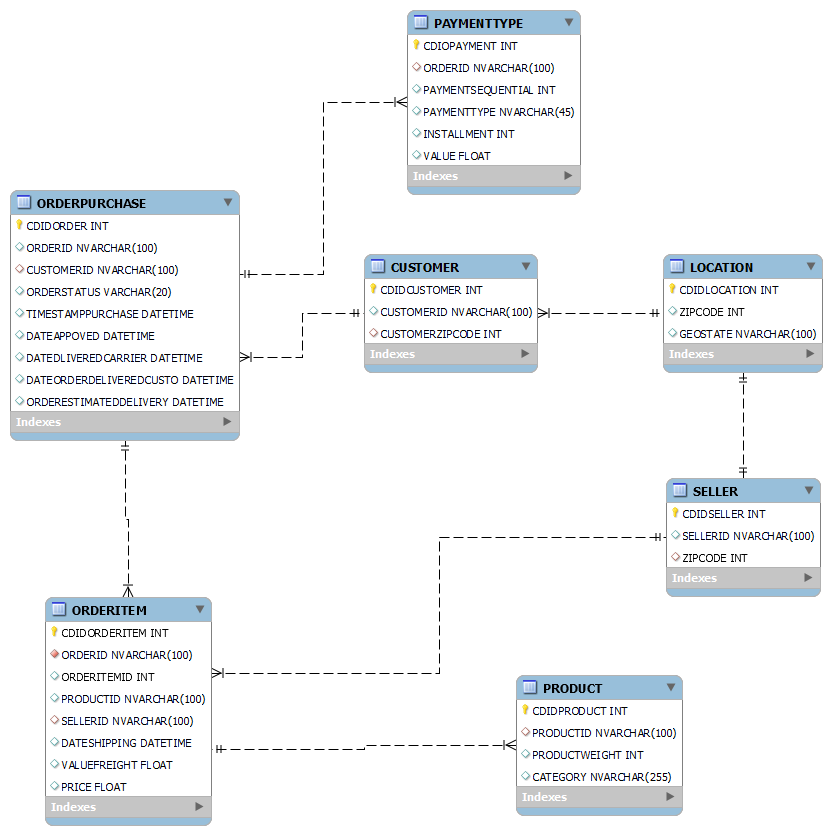
|  |  |  |  |
| --- | --- | --- | --- |
| Nombre del atributo / variable | Tabla | Formato o Tipo de Dato | Descripción |
| CDID ORDER | ORDER | INT | Código numérico del número de Orden de Pedido |
| CUSTOMERID | ORDER | NVARCHAR(100) | Identificador del Cliente |
| ORDERSTATUS | ORDER | VARCHAR(20) | Estado de la Orden |
| TIMESTAMPPURCHASE | ORDER | VARCHAR(45) | Fecha y hora de compra |
| DATE APROVED | ORDER | VARCHAR(45) | Fecha aprobación de la orden |
| CDIPAYMENT | PAYMENT TYPE | INT | ID del medio de pago |
| ORDERID | PAYMENT TYPE | NVARCHAR(100) | Código Alfa Numérico de la Orden |
| PAYMENTSEQUENTIAL | PAYMENT TYPE | INT | Consecutivo entero del medio de pago utilizado en una órden |
| PAYMENTTYPE | PAYMENT TYPE | NVARCHAR(45) | Tipo Carácter representado por medio de pago |
| INSTALLMENT | PAYMENT TYPE | INT | Número entero de cuotas aplicado a la orden |
| VALUE | PAYMENT TYPE | FLOAT | Valor decimal de la compra |
| CDICUSTOMER | CUSTOMER | INT | Identificador entero único de base de datos de un cliente. |
| CUSTOMERID | CUSTOMER | NVARCHAR(100) | Número entero de Identifición del comprador |
| CUSTOMERZIPCODE | CUSTOMER | INT | Código postal del comprador |
| CDILOCATION | LOCATION | INT | Identificador único de la ubicación |
| ZIPCODE | LOCATION | INT | Número entero Código postal de la localización |
| GEOSTATE | LOCATION | NVARCHAR(100) | Carácter alfanumérico Código ISO del estado |
| CDI ORDERITEM | ORDERITEM | INT | Identificador único numérico entero de un ítem de una orden en base de datos |
| ORDEID | ORDERITEM | NVARCHAR(100) | Código Alfa Numérico de la Orden |
| ORDERITEMID | ORDERITEM | INT | Identificador único numérico entero de un ítem de una orden |
| PRODUCTD | ORDERITEM | NVARCHAR(100) | Identificador único del producto |
| SELLERID | ORDERITEM | NVARCHAR(100) | Identificador único del vendedor |
| DATESHIPPING | ORDERITEM | VARCHAR(45) | Fecha de envío del producto |
| VALUEFREIGHT | ORDERITEM | FLOAT | Costo de envío |
| CDIPRODUCT | PRODUCT | INT | Número entero de Identificación de productos |
| PRODUCTID | PRODUCT | NVARCHAR(100) | Identificador único alfanumérico del producto. |
| PRODUCTWEIGHT | PRODUCT | INT | Identificador entero Peso del producto. |
| CATEGORY | PRODUCT | NVARCHAR(250) | Carácter Tipo en Categorías del producto |
| CODIDSELLER | SELLER | INT | Identificador entero a nivel de base de datos que identifica a un vendedor. |
| SELLERID | SELLER | NVARCHAR(100) | Identificador único alfanumérico que identifica a un vendedor. |
| ZIPCODE | SELLER | INT | Código postal entero del vendedor |

* 1. **Modelo del dominó (Johnatan)**

***Observación****: Incluya el gráfico del modelo del dominio que representa la estructura de datos de su problema.*

1. **Modelo Entidad-Relación**
   1. **Toma de pantalla del modelo E-R**

***Observación****: lo que se pide, puede usar https://draw.io o Microsoft Visio® y modele usando la notación de Barker.*



* 1. **Sentencia o consulta de creación del tabla(s)**

***Observación****: Escriba el código en el Sistema de Gestión de Bases de Datos Relacionales de su elección (se recomienda SQLite por simplicidad, mediante https://sqlitebrowser.org/) para crear las tablas que corresponda con su conjunto de datos específico. Almacene en el repositorio* **(REPO EN GITHUB)** *el script con el nombre de T1.3.2.Creacion\_Tablas.sql*

***Se recomienda repasar SQL en https://www.w3schools.com/sql/default.asp***

* 1. **Sentencias para Insertar datos**

***Observación****: Escriba el código para insertar los datos en cada una de las tablas creadas. Almacene en el repositorio* **(REPO EN GITHUB)** *el script con el nombre de T1.3.3.Insertar\_Datos.sql*

* 1. **Sentencia de consulta**

***Observación****: realice la exploración básica de los datos, conteos totales y por categorías, máximos, promedio y mínimos.* Es decir, aplique estadística descriptiva con el fin de conocer las propiedades de los datos y entenderlos lo mejor posible. Use solamente sentencias SQL. Anexe las tomas de pantalla donde evidencie la sentencia SQL y su correspondiente ejecución. Además, *Almacene en el repositorio* **(REPO EN GITHUB)** *el script con el nombre de T1.3.4.Consultar\_Datos.sql*

1. **MongoDB**
   1. **Sentencia o consulta de creación del documento(s)**

***Observación****: Escriba el código en MongoDB para crear al menos 20 documentos que correspondan a su conjunto de datos específico. Almacene en el repositorio* **(REPO EN GITHUB)** *el script con el nombre de T1.4.1.Creacion\_Documentos.sql*

* 1. **Sentencia de consulta**

***Observación****: Realice la exploración básica de los datos, conteos totales y por categorías, máximos, promedio y mínimos.* Es decir, aplique estadística descriptiva con el fin de conocer las propiedades de los datos y entenderlos lo mejor posible. Use solamente sentencias SQL. Anexe las tomas de pantalla donde evidencie la sentencia SQL y su correspondiente ejecución. Además, *Almacene en el repositorio* **(REPO EN GITHUB)** *el script con el nombre de T1.4.2.Consultar\_Datos.sql*

1. **Análisis de lectura**

***Observación****: Considerando el artículo: “*The Definitive Guide to Graph Databases for the RDBMS Developer*” de Neo4J. Compartido en las carpeta de lecturas recomendadas. Analice y responda cada pregunta en máximo 150 palabras:*

1. ¿Cuáles son las limitaciones, que se pueden inferir de la lectura, para migrar los conjuntos de datos relacionales a NoSQL?

* Una gran cantidad de JOINs Cuando utiliza consultas que JOIN en muchas tablas diferentes, hay una explosión de complejidad y consumo de recursos informáticos. Esto da como resultado un aumento correspondiente en los tiempos de respuesta a las consultas.
* Numerosos Self-JOINs (o JOINs recursivos) Las declaraciones Self-JOIN son comunes para las representaciones jerárquicas y de árbol de datos, pero atravesar relaciones uniendo tablas repetidamente a sí mismas es ineficaz.
* Cambios frecuentes en el esquema En un momento en el que la agilidad empresarial es primordial, los administradores de bases de datos suelen postergar las solicitudes de cambios porque el esquema de Las bases de datos relacionales no está diseñadas para cambios y modificaciones frecuentes. Los cambios de esquema comunes indican que los datos o requisitos están evolucionando rápidamente, lo que exige un modelo de datos más flexible.
* Consultas de ejecución lenta (a pesar de un gran ajuste) Su administrador de bases de datos puede utilizar todos los trucos del libro para acelerar los tiempos de consulta, pero muchas consultas aún no son lo suficientemente rápidas para respaldar sus las necesidades de la aplicación. Además, la desnormalización de los modelos de datos para el rendimiento puede afectar negativamente la calidad de los datos y el comportamiento de las actualizaciones.
* Calcular previamente sus resultados Debido a que las consultas se ejecutan tan lentamente, muchas aplicaciones calculan previamente sus resultados utilizando datos anteriores. Sin embargo, esto está utilizando de manera efectiva el dato para consultas que deben manejarse en tiempo real hoy. Además, su sistema generalmente debe pre-calcular el 100% de su datos, incluso si sólo se accede al 1-2% de ellos en un momento dado.

1. ¿Cuáles limitaciones adicionales que se deben considerar, a parte de las mencionadas en el artículo?

* No están diseñadas para la velocidad de la agilidad empresarial.
* No están diseñadas o/y optimizadas para manejar datos conectados
* No todas la bases de datos contemplan la atomicidad de las instrucciones y la integridad de los datos.
* son simplemente inapropiadas siempre que las relaciones de datos sean clave para el éxito en tiempo real ya que dificulta la agilidad en otros sistemas de software.
* las bases de datos relacionales no son suficientes para manejar el volumen, la velocidad y variedad de los datos actuales
* Las bases de datos relacionales, con sus esquemas rígidos y características de modelado complejas, no son una herramienta especialmente buena para respaldar un cambio rápido

1. ¿Cuáles son las razones (criterios) que se deben considerar para migrar un conjunto de datos relacionados a NoSQL?
2. Su crecimiento y Evolución según la demanda de la aplicación del Usuario
3. Tener una representación precisa del modelo del dominio
4. El proceso de construir un modelo de base de datos relacional no es adecuado para preguntar o/y responder.