



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: 2021.05.19 10:22

1. RESPONSABLES

Nombre Completo – Documento de Identificación
1. María Jenifer Montoya Grajales 42'694.359
2. Efraín Galvis Amaya 1'152.193.093
3. Daniel Felipe Pérez Grajales 1'152.435.277
4. Sandy Pahola Hernandez Guerrero 1'022.161.373
5. Tomás Simón Gómez 1'214.720.739
REPO EN GITHUB: https://github.com/Dfperezgdatascientist/Nociones-de-Arquitectura-de-la-Informacion.git

2. 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> o <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, envía por <https://forms.gle/h7ty3yZykaUq5m7y6>**
-

1 COMPRENSIÓN DEL NEGOCIO

1.1 DESCRIPCIÓN DEL CONTEXTO DEL NEGOCIO.

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



Los datos pertenecen al sector servicios específicamente contact center, donde se recolectan a partir de llamadas que los usuarios realizan.

Las marcas contratan con el contact center el manejo de la información en la etapa postventa de sus productos, allí se contratan asesores que se encargan de responder inquietudes, quejas o reclamos que los usuarios tienen con respecto a un producto.

La generación de los datos se hace a partir de las llamadas que los clientes realizan y la encuesta de satisfacción que realizan algunos usuarios después de cada llamada. Cada llamada queda registrada con un ID, cédula, hora de llamada, fecha, motivo de llamada, tiempo de duración, tiempo de espera, asesor que respondió. La encuesta de satisfacción contiene las calificaciones a las preguntas de “Qué tan satisfecho quedaste con la atención recibida” calificación del 1 a 5 y “Recomendarías la marca a un familiar” calificación de 0-10, el id de la llamada, asesor que lo atendió y tiempo de duración.

1.2 IDENTIFICACIÓN DEL PROBLEMA:

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

Dentro de las métricas internas de negocio se encuentran la duración y la satisfacción del usuario. La duración de las llamadas determina la cantidad de personas que se requieren para atenderlas, afecta directamente los ingresos de la cuenta. Mientras que la satisfacción y el NPS de los usuarios determinan la relación directa con el cliente y empatía con la marca, entre mejor sea este indicador, quiere decir que los usuarios se sienten bien atendidos y satisfechos.

Se requiere consultar las métricas de duración de llamadas y duración por motivo de contacto, para identificar aquellas que deben mejorar en tiempos y hacer seguimientos continuos. En cuanto a encuesta de satisfacción es necesario saber el promedio de satisfacción con el servicio y la satisfacción con la marca. Una vez conocidos estos indicadores se pueden implementar planes de acción para mejorar rentabilidad.

1.3 DETERMINACIÓN DE OBJETIVOS:

Describe 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).

Desarrollar el seguimiento de métricas como: duración de llamadas, duración por motivo de llamada y satisfacción para mejorar la rentabilidad y el servicio de las líneas del contact center.



Conociendo cuales son los motivos de llamada que mayor duración tiene, el equipo de formación puede plantear herramientas para mejorar la ruta de entrenamiento en asesores nuevos y así atender llamadas de una manera más eficiente y el equipo de procesos busque la optimización en aplicativos para mejorar los tiempos. Con este tipo de conocimiento se mejoraría la rentabilidad de la línea al atender llamadas con menos tiempos de atención

Identificar los motivos de llamadas que generan mayor insatisfacción en los usuarios que se comunican a la línea de atención, con este tipo de conocimiento se despliegan planes de acción entre el área de calidad y el cliente para mejorar los tipos de soluciones que se le brindan al usuario y así mejorar el indicador.

1.4 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 se hace seguimiento del indicador de duración de llamadas de manera mensual, pero no se hace seguimiento a la duración por tipo de llamadas, debido a que los libros de Excel no soportan el volumen de datos, adicional no existen maneras de relacionar las tablas de satisfacción con las de llamadas, por lo tanto, el contact center no tiene conocimiento de estas cifras que pueden atacar desde la raíz.

Con la solución la operación podrá conocer cuáles son los motivos de llamadas que deben enfocar sus esfuerzos por las duraciones largas o por altas insatisfacciones por parte del usuario y así mejorar los indicadores de manera significativa

2 COMPRENSIÓN DE LOS DATOS

2.1 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 compáralo(s). Llame el archivo T1.2.1.Datos.zip

Se tienen dos tipos de fuentes la primera corresponde a las bases de datos de las llamadas de una línea que ingresan al contact center y la segunda corresponde a las encuestas de satisfacción que los usuarios responden, la recolección de la información se realiza de manera diaria y 7*24 horas, sin embargo, se



encuentran en las tablas de manera diaria en las tablas antes de las 6 am para comenzar a realizar todas las mediciones y tableros de control que se deben entregar al cliente. El problema encontrado radica en la volumetría de los datos para trabajarlos. Para solucionar esto se tomó una muestra y se dividió en sub-archivos para subirlos a Mongo y SQL.

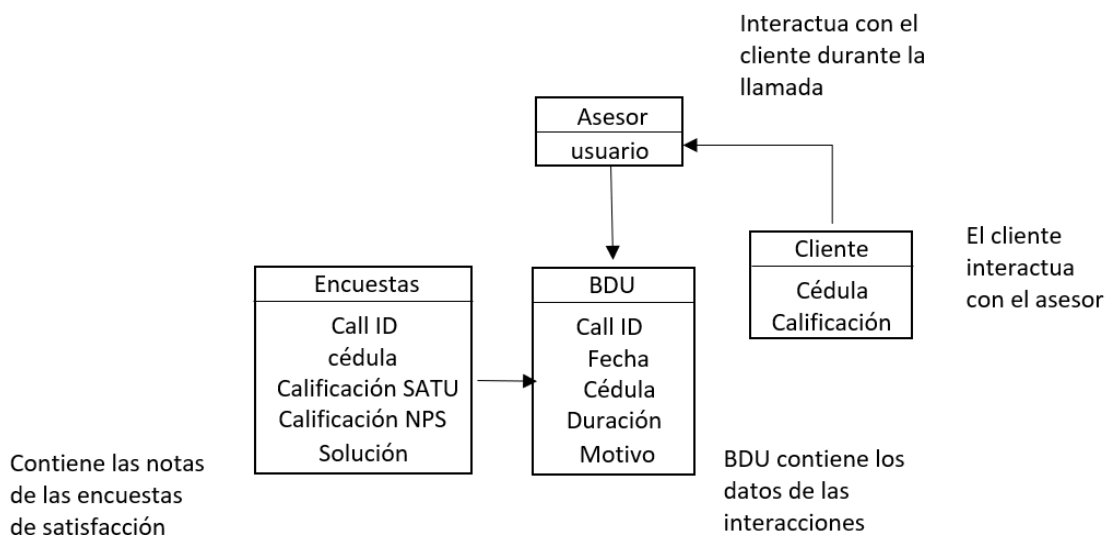
2.2 DESCRIPCIÓN DE DATOS (DICCIONARIO):

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

Nombre del atributo / variable	Formato o Tipo de Dato	Descripción
Callid	Numérico/entero	Identificador único de cada llamada
Cc	Texto	Cédula del usuario que realizó la llamada
Duración	Numérico/decimal	Tiempo de duración de cada llamada
Motivo	Texto	Motivo por el cual el usuario se contactó
Fecha/hora	Fecha	Fecha y hora cuando se realizó la llamada
NPS	Numérico/entero	Calificación de 1-10 recomendación de la marca
SATU	Numérico/entero	Calificación 1-5 con la satisfacción con el servicio

2.3 MODELO DEL DOMINIO

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

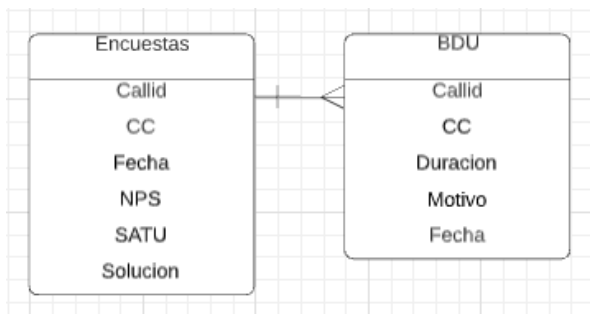




3 MODELO ENTIDAD-RELACIÓN

3.1 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.



3.2 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>

```
import sqlite3
conn = sqlite3.connect('mydb')
#Create a cursor
cur = conn.cursor()
#Create a table
cur.execute(""" CREATE TABLE BDU (
Callid integer,
CC text,
Duracion REAL,
Motivo text,
Fecha blob
```



```
)  
""")  
##Commit our command  
conn.commit()  
#Close our Connection  
conn.close()  
cur.execute(""" CREATE TABLE Encuestas (  
S integer,  
Callid integer,  
CC text,  
Fecha blob,  
NPS integer,  
Satu integer,  
Solucion integer  
)  
""")  
##Commit our command  
conn.commit()  
#Close our Connection  
conn.close()
```

3.3 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*

```
#### BDU TABLE DATA DEFINITION  
BDU_data = [(1,667045,80.1687678307369,"Motivo 1","2021-01-26"),  
(2,1047252,15.0818250553401,"Motivo 6","2021-02-14"),  
(3,1447707,21.0223313801845,"Motivo 5","2021-02-16"),
```



UNIVERSIDAD
NACIONAL
DE COLOMBIA

Facultad de Minas

Departamento de Ciencias de la Computación y de la Decisión

Medellín - Colombia

```
(4,1063849,5.96981487337549,"Motivo 1","2021-03-16"),
(5,602560,4.57964067345543,"Motivo 10","2021-02-07"),
(6,179189,97.2372250139739,"Motivo 7","2021-03-16"),
(7,260250,45.006232583378,"Motivo 3","2021-03-07"),
(8,317913,24.9202013654295,"Motivo 9","2021-03-28"),
(9,162902,12.4202135308601,"Motivo 3","2021-03-22"),
(10,1301957,64.6590549414908,"Motivo 3","2021-01-03"),
(11,1343547,53.2969008848885,"Motivo 8","2021-01-29"),
(12,150089,14.7663090161538,"Motivo 4","2021-01-15"),
(13,665807,38.7945306326397,"Motivo 10","2021-02-05"),
(14,617421,25.3158006043877,"Motivo 2","2021-04-05"),
(15,628623,68.227389577953,"Motivo 10","2021-03-03"),
(16,320464,42.5750403198457,"Motivo 2","2021-02-19"),
(17,1047556,33.0461752899077,"Motivo 4","2021-04-11"),
(18,430611,83.2193743492679,"Motivo 2","2021-04-03"),
(19,175952,37.2940649932286,"Motivo 1","2021-01-04"),
(20,363078,116.016717968589,"Motivo 10","2021-01-14")]

#### BDU TABLE DATA LOAD

cur.executemany("INSERT INTO BDU VALUES (?, ?, ?, ?, ?)",BDU_data)

#### ENCUESTAS TABLE DATA DEFINITION
Encuestas_data = [("",1,667045,"2021-01-26",1,1,1),
("2",2,1047252,"2021-02-14",4,5,1),
("3",3,1447707,"2021-02-16",6,3,1),
("4",4,1063849,"2021-03-16",6,3,0),
("5",5,602560,"2021-02-07",4,5,1),
```



```
("6",6,179189,"2021-03-16",9,5,1),
("7",7,260250,"2021-03-07",4,2,1),
("8",8,317913,"2021-03-28",8,1,1),
("9",9,162902,"2021-03-22",5,1,1),
("10",10,1301957,"2021-01-03",3,4,1),
("11",11,1343547,"2021-01-29",1,4,1),
("12",12,150089,"2021-01-15",2,4,1),
("13",13,665807,"2021-02-05",8,3,1),
("14",14,617421,"2021-04-05",6,1,1),
("15",15,628623,"2021-03-03",8,2,1),
("16",16,320464,"2021-02-19",6,2,1),
("17",17,1047556,"2021-04-11",2,1,1),
("18",18,430611,"2021-04-03",1,4,1),
("19",19,175952,"2021-01-04",8,3,0),
("20",20,363078,"2021-01-14",4,3,1)]
```

3.4 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*

- Carga de tablas BDU y Encuestas

```
#### ENCUESTAS TABLE DATA LOAD

cur.executemany("INSERT INTO Encuestas VALUES
(?,?,?,?,?,?,?)",Encuestas_data)

##VERIFYING BDU DATA LOAD WITH PRINT STATEMENT

cur.execute("SELECT * FROM BDU")
```




```
items = cur.fetchall()

for item in items:

    print(item)

##VERIFYING ENCUESTAS DATA LOAD WITH PRINT STATEMENT

cur.execute("SELECT * FROM Encuestas")

items = cur.fetchall()

for item in items:

    print(item)
```

```
(1, '667045', 80.1687678307369, 'Motivo 1', '2021-01-26')
(2, '1047252', 15.0818250553401, 'Motivo 6', '2021-02-14')
(3, '1447707', 21.0223313801845, 'Motivo 5', '2021-02-16')
(4, '1063849', 5.96981487337549, 'Motivo 1', '2021-03-16')
(5, '602560', 4.57964067345543, 'Motivo 10', '2021-02-07')
(6, '179189', 97.2372250139739, 'Motivo 7', '2021-03-16')
(7, '260250', 45.006232583378, 'Motivo 3', '2021-03-07')
(8, '317913', 24.9202013654295, 'Motivo 9', '2021-03-28')
(9, '162902', 12.4202135308601, 'Motivo 3', '2021-03-22')
(10, '1301957', 64.6590549414908, 'Motivo 3', '2021-01-03')
(11, '1343547', 53.2969008848885, 'Motivo 8', '2021-01-29')
(12, '150089', 14.7663090161538, 'Motivo 4', '2021-01-15')
(13, '665807', 38.7945306326397, 'Motivo 10', '2021-02-05')
(14, '617421', 25.3158006043877, 'Motivo 2', '2021-04-05')
(15, '628623', 68.227389577953, 'Motivo 10', '2021-03-03')
(16, '320464', 42.5750403198457, 'Motivo 2', '2021-02-19')
(17, '1047556', 33.0461752899077, 'Motivo 4', '2021-04-11')
(18, '430611', 83.2193743492679, 'Motivo 2', '2021-04-03')
(19, '175952', 37.2940649932286, 'Motivo 1', '2021-01-04')
```



(20, '363078', 116.016717968589, 'Motivo 10', '2021-01-14')

(' ', 1, '667045', '2021-01-26', 1, 1, 1)
(2, 2, '1047252', '2021-02-14', 4, 5, 1)
(3, 3, '1447707', '2021-02-16', 6, 3, 1)
(4, 4, '1063849', '2021-03-16', 6, 3, 0)
(5, 5, '602560', '2021-02-07', 4, 5, 1)
(6, 6, '179189', '2021-03-16', 9, 5, 1)
(7, 7, '260250', '2021-03-07', 4, 2, 1)
(8, 8, '317913', '2021-03-28', 8, 1, 1)
(9, 9, '162902', '2021-03-22', 5, 1, 1)
(10, 10, '1301957', '2021-01-03', 3, 4, 1)
(11, 11, '1343547', '2021-01-29', 1, 4, 1)
(12, 12, '150089', '2021-01-15', 2, 4, 1)
(13, 13, '665807', '2021-02-05', 8, 3, 1)
(14, 14, '617421', '2021-04-05', 6, 1, 1)
(15, 15, '628623', '2021-03-03', 8, 2, 1)
(16, 16, '320464', '2021-02-19', 6, 2, 1)
(17, 17, '1047556', '2021-04-11', 2, 1, 1)
(18, 18, '430611', '2021-04-03', 1, 4, 1)
(19, 19, '175952', '2021-01-04', 8, 3, 0)
(20, 20, '363078', '2021-01-14', 4, 3, 1)

- Se requiere identificar cual es el motivo de contacto que tiene una duración máxima de llamada

```
###BDU LONGEST CALL DURATION RECORD

cur.execute("SELECT * FROM BDU ORDER BY Duracion DESC")

col_name_list = [tuple[0] for tuple in cur.description]

print("Longest call duration record","\n")

print(col_name_list)

print(cur.fetchone(), "\n", "\n")|
```



Longest call duration record

```
['Callid', 'CC', 'Duracion', 'Motivo', 'Fecha']  
(20, '363078', 116.016717968589, 'Motivo 10', '2021-01-14')
```

El el registro de llamada con motivo 10 presenta una duración máxima de 116 segundos, lo que indica que se debe analizar con el equipo de formación que se debe optimizar para evitar este tipo de datos

- Se requiere identificar cual es el motivo de contacto que tiene una duración mínima de llamada

```
###BDU SHORTEST CALL DURATION RECORD  
  
cur.execute("SELECT * FROM BDU ORDER BY Duracion ASC")  
col_name_list = [tuple[0] for tuple in cur.description]  
print("Shortest call duration record","\n")  
print(col_name_list)  
print(cur.fetchone(), "\n", "\n")
```

Shortest call duration record

```
['Callid', 'CC', 'Duracion', 'Motivo', 'Fecha']  
(5, '602560', 4.57964067345543, 'Motivo 10', '2021-02-07')
```

El registro de llamada con motivo 10 presenta una duración mínima de 4 segundos, lo que indica que debe monitorearse para identificar si se presentó problemas de cuelgue de llamada.

- Identificar la duración promedio por tipo de llamada



```
###AVERAGE DURATION BY CALL REASON

cur.execute("SELECT Motivo,AVG(duracion)*60 FROM BDU GROUP BY Motivo
ORDER BY AVG(Duracion)")

print("Average duration by call reason","\n")

print("Motivo","\t\t",Duracion)

items = cur.fetchall()

for item in items:

    print(item[0],",",item[1])
```

Average duration by call reason

Motivo	Duracion
Motivo 6 ,	904.9095033204061
Motivo 5 ,	1261.33988281107
Motivo 4 ,	1434.374529181845
Motivo 9 ,	1495.21208192577
Motivo 3 ,	2441.710021114578
Motivo 1 ,	2468.6529539468197
Motivo 2 ,	3022.2043054700257
Motivo 8 ,	3197.81405309331
Motivo 10 ,	3414.274182789557
Motivo 7 ,	5834.233500838434

En promedio los tipos de llamada por motivo 6 presenta una duración inferior al resto de tipo de llamadas con 904 segundos, con esta información el equipo de formación evalúa que se esté dando la información de manera correcta. Mientras que el motivo 7 tiene un promedio de duración de llamadas de 5834, de ahí el equipo de procesos debe escuchar este tipo de llamadas para identificar si se puede optimizar con ayudas tecnológicas o directamente formando a los asesores en temas específicos que estén dificultando la agilidad, todo depende del problema encontrado.

- Identificar el promedio de la calificación de recomendación de la marca NPS



```
###AVERAGE NPS IN POLL

cur.execute("SELECT avg(NPS) FROM Encuestas")

print("Average NPS:",cur.fetchall()[0][0],"\n\n")
```

Average NPS: 4.8

En promedio los usuarios están calificando la recomendación de la marca NPS en 4.8 puntos, lo que indica que se deben emprender planes de acción con el cliente para identificar los procesos que más molestan a los usuarios para comenzar a subir la calificación.

- Identificar los motivos que mayor y menor satisfacción presentan los usuarios que afectan la calificación general del NPS

```
###AVG NPS BY CALL REASON - BDU and ENCUESTAS TABLE JOIN

cur.execute("""SELECT Motivo,AVG(NPS)
FROM BDU INNER JOIN Encuestas On Encuestas.Callid=BDU.callid
GROUP BY Motivo ORDER BY AVG(NPS) DESC""")

print("Average NPS by call reason: \n\n")

print("Motivo","\t","NPS","\n")

items = cur.fetchall()

for item in items:

    print(item[0],",",item[1])
```

Average NPS by call reason:

Motivo NPS

Motivo 7 , 9.0

Motivo 9 , 8.0

Motivo 10 , 6.0

Motivo 5 , 6.0

Motivo 1 , 5.0

Motivo 2 , 4.333333333333333

Motivo 3 , 4.0

Motivo 6 , 4.0



Motivo 4 , 2.0

Motivo 8 , 1.0

El motivo 8 y motivo 4 en promedio tiene una calificación de 1 y 2 puntos respectivamente, por lo tanto el equipo de calidad debe identificar las posibles causas de insatisfacción de los clientes con este proceso, si son largos trámites o la no solución a través de la línea por lo que induce al cliente a calificar mal.

El motivo 7 presenta una mayor satisfacción con la marca, lo que se debe indagar qué de positivo se hace en este proceso para replicarlo a los demás motivos de llamadas.

4 MONGODB

4.1 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*

Se hace la creación de dos bases de datos cada una con 20 documentos correspondientes a la información de llamadas y las encuestas de satisfacción

- Cargue datos BDU

```
mongo> use atlas; use db; insertMany([{"callid":1,"cc":"333962","Duracion":100,"Motivo":"Motivo 1","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":2,"cc":"333963","Duracion":200,"Motivo":"Motivo 1","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":3,"cc":"333964","Duracion":150,"Motivo":"Motivo 2","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":4,"cc":"333965","Duracion":300,"Motivo":"Motivo 3","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":5,"cc":"333966","Duracion":133,"Motivo":"Motivo 4","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":6,"cc":"333967","Duracion":1250,"Motivo":"Motivo 4","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":7,"cc":"333968","Duracion":450,"Motivo":"Motivo 3","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":8,"cc":"333969","Duracion":407,"Motivo":"Motivo 5","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":9,"cc":"333969","Duracion":407,"Motivo":"Motivo 4","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":10,"cc":"333971","Duracion":159,"Motivo":"Motivo 5","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":11,"cc":"333972","Duracion":164,"Motivo":"Motivo 5","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":12,"cc":"333973","Duracion":786,"Motivo":"Motivo 5","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":13,"cc":"333974","Duracion":423,"Motivo":"Motivo 4","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":14,"cc":"333975","Duracion":564,"Motivo":"Motivo 4","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":15,"cc":"333976","Duracion":453,"Motivo":"Motivo 1","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":16,"cc":"333977","Duracion":489,"Motivo":"Motivo 2","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":17,"cc":"333978","Duracion":1357,"Motivo":"Motivo 3","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":18,"cc":"333979","Duracion":1258,"Motivo":"Motivo 4","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":19,"cc":"333980","Duracion":951,"Motivo":"Motivo 2","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}, {"callid":20,"cc":"333981","Duracion":486,"Motivo":"Motivo 2","Fecha":{"$date":"2021-01-05T00:00:00.000Z"}}]);
```

```
{
  "acknowledged": true,
  "insertedIds": [
    ObjectId("6097429366793141e0be0bdc"),
    ObjectId("6097429366793141e0be0bdd"),
    ObjectId("6097429366793141e0be0bde"),
    ObjectId("6097429366793141e0be0bdf"),
    ObjectId("6097429366793141e0be0bee"),
    ObjectId("6097429366793141e0be0bef"),
    ObjectId("6097429366793141e0be0beb"),
    ObjectId("6097429366793141e0be0bec"),
    ObjectId("6097429366793141e0be0bed"),
    ObjectId("6097429366793141e0be0bee"),
    ObjectId("6097429366793141e0be0bef"),
    ObjectId("6097429366793141e0be0b0"),
    ObjectId("6097429366793141e0be0b1"),
    ObjectId("6097429366793141e0be0b2"),
    ObjectId("6097429366793141e0be0b3"),
    ObjectId("6097429366793141e0be0b4"),
    ObjectId("6097429366793141e0be0b5"),
    ObjectId("6097429366793141e0be0b6"),
    ObjectId("6097429366793141e0be0b7"),
    ObjectId("6097429366793141e0be0b8"),
    ObjectId("6097429366793141e0be0b9"),
    ObjectId("6097429366793141e0be0ba"),
    ObjectId("6097429366793141e0be0bb"),
    ObjectId("6097429366793141e0be0bc"),
    ObjectId("6097429366793141e0be0bd"),
    ObjectId("6097429366793141e0be0be"),
    ObjectId("6097429366793141e0be0bf")
  ]
}
```

- Cargue datos Encuestas



```
MongoDB Enterprise atlas-sf3375-shard-0:PRIMARY> db.Encuestas.insertMany([{"callid":"1","cc":"333962","NPS":2,"SATU":1}, {"callid":"2","cc":"333963","NPS":2,"SATU":1}, {"callid":"3","cc":"333964","NPS":3,"SATU":1}, {"callid":"4","cc":"333965","NPS":4,"SATU":1}, {"callid":"5","cc":"333966","NPS":4,"SATU":1}, {"callid":"6","cc":"333967","NPS":3,"SATU":1}, {"callid":"7","cc":"333968","NPS":5,"SATU":1}, {"callid":"8","cc":"333969","NPS":3,"SATU":1}, {"callid":"9","cc":"333960","NPS":3,"SATU":1}, {"callid":"10","cc":"333971","NPS":5,"SATU":1}, {"callid":"11","cc":"333972","NPS":5,"SATU":1}, {"callid":"12","cc":"333973","NPS":4,"SATU":1}, {"callid":"13","cc":"333974","NPS":4,"SATU":1}, {"callid":"14","cc":"333975","NPS":3,"SATU":1}, {"callid":"15","cc":"333976","NPS":3,"SATU":1}, {"callid":"16","cc":"333977","NPS":4,"SATU":1}, {"callid":"17","cc":"333978","NPS":5,"SATU":1}, {"callid":"21","cc":"333979","NPS":3,"SATU":1}, {"callid":"22","cc":"333980","NPS":2,"SATU":1}, {"callid":"23","cc":"333981","NPS":5,"SATU":1}]);
{"acknowledged":true,
"insertedIds":[
  ObjectId("609742d066793141e0be0bf0"),
  ObjectId("609742d066793141e0be0bf1"),
  ObjectId("609742d066793141e0be0bf2"),
  ObjectId("609742d066793141e0be0bf3"),
  ObjectId("609742d066793141e0be0bf4"),
  ObjectId("609742d066793141e0be0bf5"),
  ObjectId("609742d066793141e0be0bf6"),
  ObjectId("609742d066793141e0be0bf7"),
  ObjectId("609742d066793141e0be0bf8"),
  ObjectId("609742d066793141e0be0bf9"),
  ObjectId("609742d066793141e0be0bfa"),
  ObjectId("609742d066793141e0be0bfb"),
  ObjectId("609742d066793141e0be0bfc"),
  ObjectId("609742d066793141e0be0bfd"),
  ObjectId("609742d066793141e0be0bfe"),
  ObjectId("609742d066793141e0be0bff"),
  ObjectId("609742d066793141e0be0c0"),
  ObjectId("609742d066793141e0be0c1"),
  ObjectId("609742d066793141e0be0c2"),
  ObjectId("609742d066793141e0be0c3")
]}
MongoDB Enterprise atlas-sf3375-shard-0:PRIMARY>
```

4.2 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*

- Se realiza la búsqueda del registro de la **duración máxima** de una llamada esto con el fin de identificar cual es el motivo que genera estos valores extremos en la duración y se encontró que el motivo 4 con 987 segundos es el que cumple con esta característica

```
MongoDB Enterprise atlas-sf3375-shard-0:PRIMARY> db.BDU.find().sort({Duracion:-1}).limit(1) // Registro con el valor MAX
{"_id": ObjectId("6097398190e051ae9cc19f6f"), "Callid": "9", "cc": "333960", "Duracion": 987, "Motivo": "Motivo 4", "Fecha": { "$date": "2021-01-05T00:00:00.000Z" } }
```

- Se realiza la búsqueda del registro de la **duración mínima** de una llamada para validar que se estén haciendo bien lo procesos y los asesores no están incurriendo en cuelgues de llamadas y se encuentra que el registro con menor duración es de 100 segundos y corresponde al motivo 1

```
MongoDB Enterprise atlas-sf3375-shard-0:PRIMARY> db.BDU.find().sort({Duracion:+1}).limit(1) // Registro con el valor MIN
{"_id": ObjectId("6097398190e051ae9cc19f6f"), "Callid": "1", "cc": "333962", "Duracion": 100, "Motivo": "Motivo 1", "Fecha": { "$date": "2021-01-05T00:00:00.000Z" } }
```

- Se desea identificar cual es el **promedio de duración de llamadas por motivos de llamadas** y se concluye que el motivo 1 tiene un menor promedio de duración de 251 segundos, mientras que el motivo 2 tiene un mayor promedio de duración de 644 segundos.

```
MongoDB Enterprise atlas-sf3375-shard-0:PRIMARY> db.BDU.aggregate([{$group: {$_id:'$Motivo', Duracion_promedio:{$avg:'$Duracion'}}}]) // duración promedio por motivo de llamada
{"_id": "Motivo 4", "Duracion_promedio": 469.1666666666667 }
{"_id": "Motivo 3", "Duracion_promedio": 369 }
{"_id": "Motivo 2", "Duracion_promedio": 644 }
{"_id": "Motivo 5", "Duracion_promedio": 514 }
{"_id": "Motivo 1", "Duracion_promedio": 251 }
```

con esta información el área de procesos debe analizar el paso a paso del motivo 2 e identificar las mejoras que debe hacer para optimizar este tiempo.



- Se debe identificar cual es el motivo de llamada que genera un **promedio de calificación menor de NPS**.

```
MongoDB Enterprise atlas-sf3375-shard-0:PRIMARY> db.BOU.aggregate([{$lookup: { from: 'Encuestas', localField: 'CallId', foreignField: 'CallId', as: 'combined' } }, {$unwind: '$combined' }],{$group: { _id: '$Motivo', NPS_promedio: { $avg: '$combined.NPS' } } }]); // NPS prom por motivo
{ "_id": "Motivo 5", "NPS_promedio": 4.25 }
{ "_id": "Motivo 1", "NPS_promedio": 2.3333333333333335 }
{ "_id": "Motivo 4", "NPS_promedio": 3.4 }
{ "_id": "Motivo 3", "NPS_promedio": 4.666666666666667 }
{ "_id": "Motivo 2", "NPS_promedio": 3.5 }
```

se encuentra que el área de calidad debe intervenir en el motivo de llamada 1 porque tiene un promedio de recomendación de 2.3, lo que está impactando negativamente a la marca.

5 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?

- El proceso de migrar de un modelo de pensamiento y almacenamiento a otro generan nuevos requerimientos lo cual conlleva a largo procedimientos de migración, modificando esquemas y transformando información para mantener características previas al cambio,
- Existe una variedad considerable de métodos de mapeo y migración de datos de bases relacionales a noSQL por lo que la decisión sobre qué metodología usar puede ser un desafío
- Si bien, muchos gestores de bases de datos permiten descargar queries en diferentes formatos, hay casos en los que los archivos ACC residen dentro de la base de datos y su descarga puede ser un desafío.
- Otra limitante es la velocidad de exportación de la información de una base de datos relacional a NoSQL, pues estas no están optimizadas para la exportación de grandes cantidades de datos en periodos cortos de tiempo.

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

- Interacción humana, los desarrolladores acostumbrados a un método de modelado y almacenamiento de información tenderían a usar la lógica a la que están acostumbrados y pueden ser propensos a errores o no usar bien las ventajas de este nuevo modelo
- Objetos heredados no suelen tener identificadores en bases noSQL las cuales suelen ser necesarias para acceder y comparar información específica



3. Se sacrifica las características ACID de las DDBB relacionales para ganar mayor rendimiento y escalabilidad

4. La data huérfana puede romper las reglas de negocio, las bases SQL a través de las claves foráneas aseguran que la base de datos siempre sea válida.

3. ¿Cuáles son las razones (criterios) que se deben considerar para migrar un conjunto de datos relacionados a NoSQL?

1. Necesidad de flexibilidad para adaptarse a cambios en las relaciones entre entidades

2. Por otro lado, la necesidad de manejar mayores volúmenes de datos (escalabilidad), cuando se desconoce la demanda de los usuarios y cuando se requiere ampliar constantemente y actualizar información

3. Adicionalmente, cuando se requiere un rendimiento consistente independiente del proceso y con bajo tiempo de respuesta para mejorar por ejemplo la experiencia del usuario. Dado que no se requieren hacer múltiples Joins entre tablas y ahorra mucho tiempo de computación