

Taller Arquitectura de la Información

Versión: 2020.11.09 12:11

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.

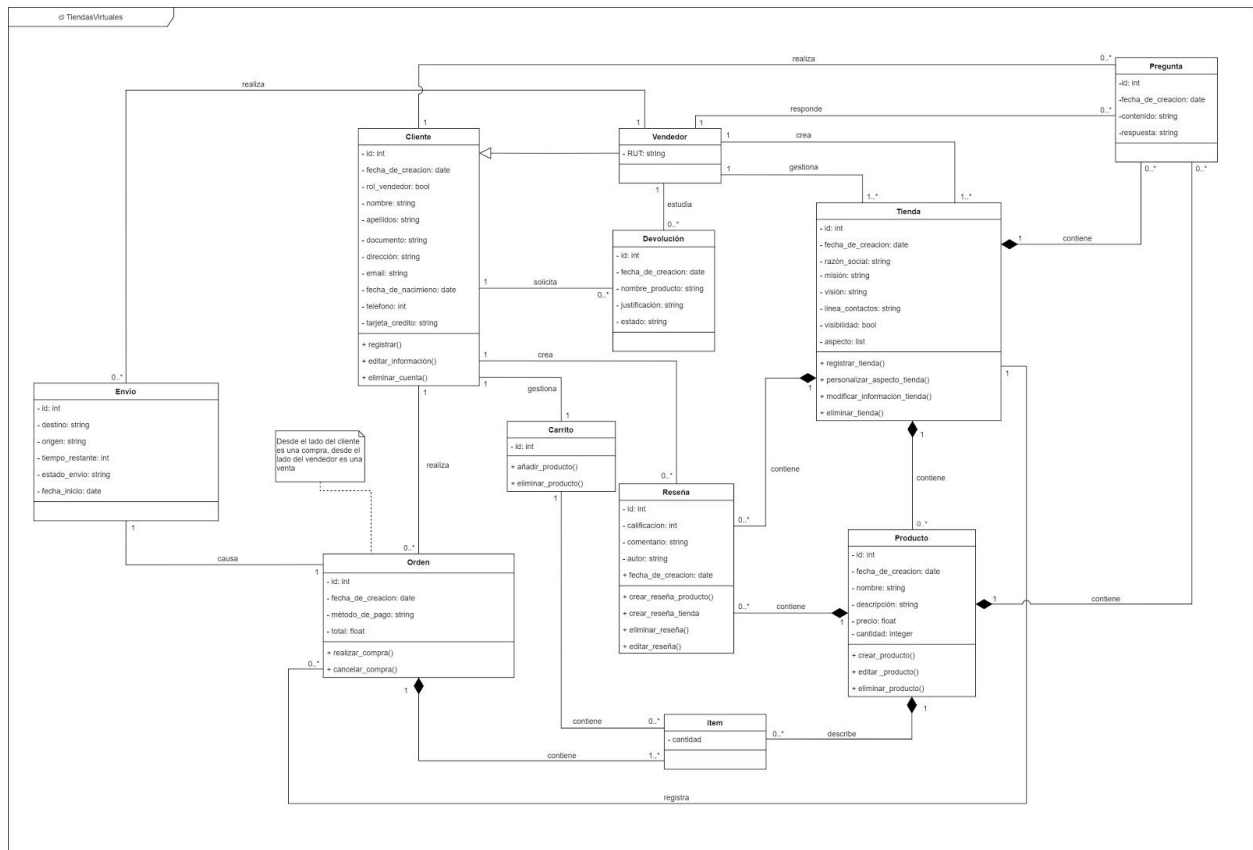
RESPONSABLES

Nombre Completo
David Escobar Ruiz
Hinara Pastora Sanchez Mata
Juan Camilo Molina Roncancio
Mateo Espinal Londono
Pablo Restrepo Osorio
Santiago Eusse Gil
Yamid Andres Campo Gallego

Nota: La versión más reciente de este trabajo y sus correspondientes diagramas se encuentran en el repositorio ubicado en: <https://github.com/Mateoe/Fundamentos-de-Arquitectura-de-Software.git>

1 MODELO DEL DOMINIO

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



Para una mejor visualización el diagrama se encuentra disponible en el siguiente enlace:

<https://github.com/Mateoe/Fundamentos-de-Arquitectura-de-Software/blob/master/Trabajo%20Arquitectura%20de%20la%20Informaci%C3%B3n/Diagrama%20de%20clases.jpg>

2 MODELO ENTIDAD-RELACIÓN

2.1 SENTENCIA O CONSULTA DE CREACIÓN DEL

Observación: Escriba el código en el Sistema de Gestión de Bases de Datos Relacionales de su elección para crear la tabla que corresponda con su conjunto de datos específico.

```
CREATE TABLE "tbCliente" (  
    "Id"      INTEGER NOT NULL UNIQUE,  
    "fecha_Creacion"  DATETIME NOT NULL,  
    "rol_Vendedor"   BOOL NOT NULL,  
    "nombre"        VARCHAR(255) NOT NULL,  
    "apellidos"     VARCHAR(255) NOT NULL,  
    "documento_identidad"  VARCHAR(255) NOT NULL UNIQUE,  
    "direccion"     VARCHAR(255) NOT NULL,  
    "email"         VARCHAR(255) NOT NULL UNIQUE,  
    "fecha_nacimiento"  DATETIME NOT NULL,  
    "telefono"      DECIMAL(10, 0) NOT NULL,  
    "tarjeta_credito"  VARCHAR(255),  
    PRIMARY KEY("Id" AUTOINCREMENT)  
);
```

2.2 SENTENCIAS PARA INSERTAR DATOS

Observación: Escriba el código para insertar al menos 5 elementos en la tabla(s) creadas en el punto 2.1.

```
INSERT INTO "tbCliente" VALUES (1, '05/10/2020', 'false', 'Camila', 'Gomez',  
'4526886952', 'Carrera 14 # 5 - 99', 'camilagomezzz@hotmail.com', '20/05/2000',  
3105228695, NULL);
```

```
INSERT INTO "tbCliente" VALUES (2, '15/08/2020', 'true', 'Fernando', 'Cortes  
Montero', '1001459817', 'Carrera 3 # 18 - 45', 'cortesf@hotmail.com',  
'31/10/1985', 3105429966, NULL);
```

```
INSERT INTO "tbCliente" VALUES (3, '03/11/2020', 'true', 'Maria', 'Fernandez  
Lutero', '1201918926', 'Calle 10 No. 9 - 78 Centro', 'mariafl@gmail.com',  
'23/02/1996', 3044822557, '1361931883719376');
```

```
INSERT INTO "tbCliente" VALUES (4, '04/11/2020', 'false', 'Petra', 'Rodriguez  
Serna', '119368568', 'Carrera 50 # 48 - 65 Edificio Montreal',  
'prodriguezs@outlook.com', '10/04/98', 3145172557, '19466391837319376');
```

```
INSERT INTO "tbCliente" VALUES (5, '13/05/2020', 'false', 'Carlos Mario',  
'Vargas Llosa', '4523984125', 'Calle 95 # 75 - 85 interior 204',  
'camavallo@yahoo.com ', '01/01/1973', 3205269874, NULL);
```

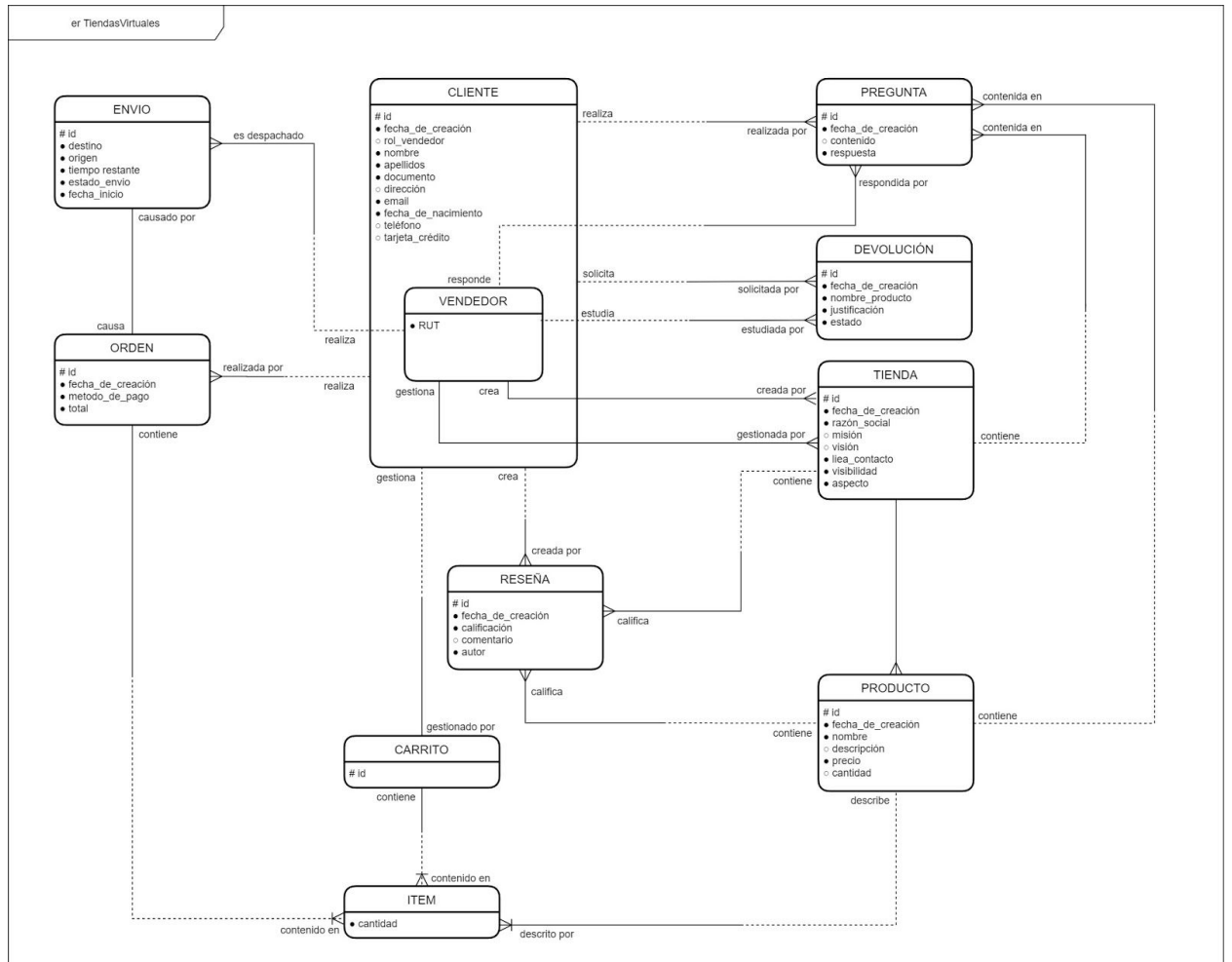
2.3 SENTENCIA DE CONSULTA

Observación: Escriba el código para consultar solo tres atributos específicos de los insertados en la sección 2.2.

```
SELECT nombre, direccion, telefono FROM tbCliente;
```

2.4 TOMA DE PANTALLA DEL MODELO E-R

Observación: lo que se pide.



Para una mejor visualización el diagrama se encuentra disponible en el siguiente enlace:

<https://github.com/Mateoe/Fundamentos-de-Arquitectura-de-Software/blob/master/Trabajo%20Arquitectura%20de%20la%20Informaci%C3%B3n/Diagrama%20entidad%20-%20relacion.jpg>

3 MongoDB

3.1 SENTENCIA O CONSULTA DE CREACIÓN DEL DOCUMENTO(S)

Observación: Escriba el código en MongoDB para crear al menos cuatro documentos que correspondan a su conjunto de datos específico.S

```
db.Cliente.insertMany([

    {_id: 1,

    fecha_de_creacion: new Date(2020, 10, 15),

    rol_vendedor: true,

    nombre: "Pedro",

    apellidos: "Gonzalez Giraldo",

    documento: "1234",

    direccion: "Calle 1 #2 - 3",

    email: "pgonzalez@gmail.com",

    fecha_de_nacimiento: new Date(1997, 7, 6),

    telefono: 1234567,

    tarjeta_de_credito: "2222"},

    {_id: 2,

    fecha_de_creacion: new Date(2020, 10, 16),

    rol_vendedor: false,

    nombre: "Ana",

    apellidos: "Ramirez Rojas",

    documento: "2345",

    direccion: "Calle 2 #3 - 4",

    email: "aramirez@gmail.com",

    fecha_de_nacimiento: new Date(2002, 1, 1),

    telefono: 2345678,

    tarjeta_de_credito: "4444"}],
```

```

    {_id: 3,
    fecha_de_creacion: new Date(2020, 10, 19),
    rol_vendedor: false,
    nombre: "Andres",
    apellidos: "Perez Velez",
    documento: "3456",
    direccion: "Calle 3 #4 - 5",
    email: "aperez@gmail.com",
    fecha_de_nacimiento: new Date(2001, 7, 20),
    telefono: 3456789,
    tarjeta_de_credito: "5555"
    },

    {_id: 4,
    fecha_de_creacion: new Date(2020, 11, 1),
    rol_vendedor: true,
    nombre: "María",
    apellidos: "Rodriguez Ospina",
    documento: "4567",
    direccion: "Calle 4 #5 - 6",
    email: "mrodriguez@gmail.com",
    fecha_de_nacimiento: new Date(1989, 5, 31),
    telefono: 4567890,
    tarjeta_de_credito: "6666"}
  ]);

```

3.2 SENTENCIA DE CONSULTA

Observación: Escriba el código en mongo para consultar 2 documentos (use limit) y muestre dos atributos específicos que usted considere, de los creados en el punto 1.1.

```
db.Cliente.find({}, {"nombre":1,"apellidos":1,"documento":1}).limit(2);
```

[illegible]

```
INSERT INTO cliente VALUES ("3", "fecha_creacion", "2020/10/20"),  
    ("3", "rol_vendedor", "false"),  
    ("3", "nombre", "Tomas"),  
    ("3", "apellidos", "Morales Duque"),  
    ("3", "documento_identidad", "1236564345"),  
    ("3", "direccion", "Carrera 19 #22-23"),  
    ("3", "email", "tomimorales@gmail.com"),  
    ("3", "fecha_nacimiento", "2001/09/02"),  
    ("3", "telefono", "3989563465"),  
    ("3", "tarjeta_credito", "34567");
```

```
INSERT INTO cliente VALUES ("4", "fecha_creacion", "2020/11/01"),  
    ("4", "rol_vendedor", "false"),  
    ("4", "nombre", "Evaluna"),  
    ("4", "apellidos", "Pastor Marulanda"),  
    ("4", "documento_identidad", "1023456857"),  
    ("4", "direccion", "Carrera 33 #22-21"),  
    ("4", "email", "pastormarula@hotmail.com"),  
    ("4", "fecha_nacimiento", "1995/02/11"),  
    ("4", "telefono", "3114569807"),
```

```
        ("4", "tarjeta_credito", "45678");

INSERT INTO cliente VALUES ("5", "fecha_creacion", "2020/11/04"),
        ("5", "rol_vendedor", "true"),
        ("5", "nombre", "Jorge"),
        ("5", "apellidos", "Lopez Velez"),
        ("5", "documento_identidad", "1923467543"),
        ("5", "direccion", "Carrera 77 #12-14"),
        ("5", "email", "jorgelove@yahoo.com"),
        ("5", "fecha_nacimiento", "1998/10/31"),
        ("5", "telefono", "3102346543"),
        ("5", "tarjeta_credito", "56789");
```

4.3 SENTENCIA DE CONSULTA

Observación: Escriba el código en *hypertable* consultar solo tres atributos específicos de los insertados en la sección 2.2.

```
SELECT nombre, direccion, email FROM cliente;
```

5 Neo4J

5.1 SENTENCIA O CONSULTA DE CREACIÓN DEL NODO(S) Y RELACIONES

Observación: Escriba el código en Neo4J para crear el (los) nodo(s) y relaciones que corresponda(n) con su conjunto de datos específico. Debe contener al menos 4 relaciones y 3 nodos.

Creación de nodos:

```
CREATE (vend:Vendedor{id:1,
                        fecha_de_creacion:"04-11-2020",
                        rol_vendedor:True,
                        nombre:'Jaime',
                        apellidos: 'Cano Mendoza',
                        documento: '1152686009',
                        direccion: 'Cll 20 # 23-21',
                        email: 'jaimestores@correo.co',
                        fecha_de_nacimiento: '22-04-1989',
                        telefono: 3023232222,
                        tarjeta_credito:'3232323231',
                        RUT: '890000323'
                        });

CREATE (tiend:Tienda{id:1,
                     fecha_de_creacion:"05-11-2020",
                     rol_vendedor:True,
                     razon_social:'El platanito',
                     linea_de_contacto: '5882211',
                     visibilidad: True,
                     aspecto: "[default]"
                     });
```

```

CREATE (tiend:Tienda{id:2,
                    fecha_de_creacion:"11-11-2020",
                    rol_vendedor:True,
                    razon_social:'El manguito',
                    linea_de_contacto: '5882211',
                    visibilidad: True,
                    aspecto: "[default]"
                });

```

Creación de relaciones:

```

MATCH (a:Vendedor),(b:Tienda)
WHERE a.nombre="Jaime" and b.razon_social="El platanito"
CREATE (a)-[:crea]->(b)
CREATE (a)-[:gestionaria]->(b)

```

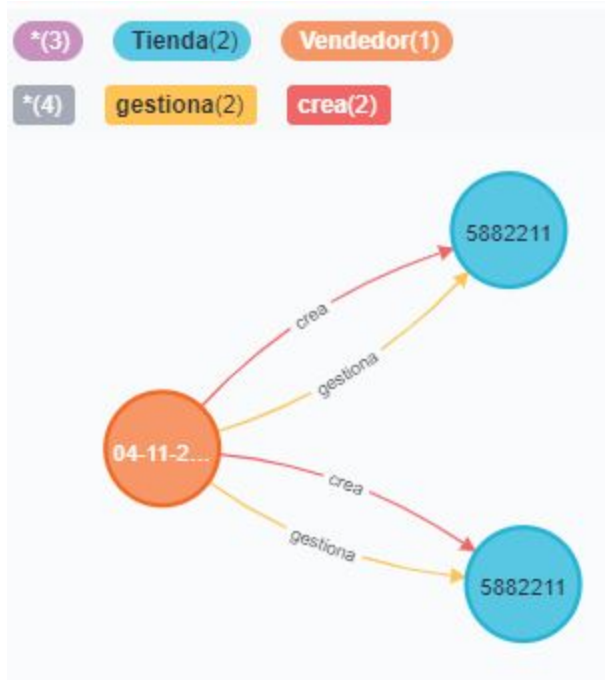
```

MATCH (a:Vendedor),(b:Tienda)
WHERE a.nombre="Jaime" and b.razon_social="El manguito"
CREATE (a)-[:crea]->(b)
CREATE (a)-[:gestionaria]->(b)

```

5.2 GRAFO GENERADO

Observación: inserte la toma de pantalla del grafo generado.



6 ANÁLISIS DE LECTURA

Observación: Considerando el artículo: “A Framework for Migrating Relational Datasets to NoSQL” de Leonardo Rocha et al. (2015). Analice y responda concretamente:

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

La lectura afirma que las organizaciones deciden migrar de una base de datos relacional a una NoSQL debido a la necesidad de almacenar y administrar una masiva cantidad de datos para ser utilizados por aplicaciones de software actuales, para las cuales el rendimiento en una RDBMS no es el adecuado. Entre las limitaciones presentes cuando se piensa migrar la base de datos se encuentran las dependencias de datos de cada aplicación con la misma por lo que se debe adaptar cada aplicación dependiente del modelo original para que sea compatible con el nuevo modelo, cambiando la manera en cómo realizan cada request a la sintaxis del nuevo DBMS. Además, se debe mantener consistencia semántica entre los dos modelos, para evitar redundancia entre datos y la distorsión de los mismos. La recuperación y migración de metadatos hace parte de la preservación de la consistencia semántica entre modelos; metadatos a considerar incluyen tipos de datos y restricciones de integridad.

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

Además de lo mencionado previamente, se debe de considerar la totalidad de la arquitectura del sistema, teniendo en cuenta el flujo de datos que pasa por cada componente y las dependencias y las interfaces utilizadas para recibir y transferir datos entre estos. Esto es debido a la repartición de los requisitos del sistema en cada componente, por lo que puede existir un nivel de priorización mayor con respecto a los datos en algunas partes del sistema que en otras.

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

Se deben tener en cuenta el tipo de dataset con el que se está tratando, para evaluar la necesidad de migrar el conjunto a un modelo NoSQL. El volumen y el tipo de datos a almacenar y administrar determinará la decisión de migrar de un modelo relacional a un modelo NoSQL. Además, se debe de tener en cuenta las dependencias de las aplicaciones del sistema con la base de datos, y el flujo de datos que pasa a través de ellas. Puede darse el caso de que las aplicaciones no utilizan una cantidad de datos significativa como para decidir cambiar todo el modelo. Por último, la necesidad de optimizar el desempeño del sistema también puede determinar la decisión de migrar el conjunto a NoSQL.