

*Licenciatura en Sistemas - Bases de Datos II*

*Titular: Ing. Federico Ribeiro **fribeirounla@gmail.com***

CONTENIDO

Trabajo práctico integrador .....	2
Objetivos.....	2
Objetivos en perspectiva.....	2
Entrega del trabajo.....	2
Evaluación / aprobación.....	2
- Primera parte .....	2
Descripción del problema: .....	3
Relevamiento.....	3
Reportes requeridos.....	3
Tareas a realizar:.....	4
Entregas del trabajo.....	5
Entrega 01.....	5
Entrega 02.....	5

*Licenciatura en Sistemas - Bases de Datos II*

*Titular: Ing. Federico Ribeiro **fribeirounla@gmail.com***

## TRABAJO PRÁCTICO INTEGRADOR

### OBJETIVOS

Se desea que el alumno pueda analizar un problema de bases de datos desde un enfoque NoSQL, debiendo partir primeramente desde un punto de vista relacional, para luego generar los documentos JSON que serán almacenados en una base de datos MongoDB.

### OBJETIVOS EN PERSPECTIVA

De manera ordenada, los objetivos planteados en el trabajo serían los siguientes, para los cuales se irán requiriendo las diferentes entregas:

- I. Análisis del problema desde un enfoque relacional (ERD)
- II. Armado de clases Java que grafiquen las relaciones encontradas.
- III. Construcción de mecanismos de serialización de la información (para el armado de los documentos Json)
- IV. Almacenamiento de los documentos Json en Mongo DB
- V. Construcción de consultas a la base de datos Mongo DB.

### ENTREGA DEL TRABAJO

El trabajo será entregado de manera parcial donde se establecerán fechas para cada una de ellas, que deberán ser respetadas para lograr llegar al final de la cursada con el trabajo terminado.

### EVALUACIÓN / APROBACIÓN

Cada entrega será evaluada y en caso de que sea necesario, se requerirá su corrección. Al final del trabajo se establecerá la nota final teniendo en cuenta todas las entregas anteriores<sup>1</sup>.

### - PRIMERA PARTE

La primera parte se centra en el análisis del problema y en la generación del diseño de entidades, relaciones y sus clases asociadas (POJOs). A partir del siguiente enunciado deberán efectuar las tareas de relevamiento del problema.

<sup>1</sup> Para la cursada del 2020, dada la situación de cuarentena dispuesta por el gobierno argentino en relación a la pandemia COVID-19, la evaluación de la aprobación de la cursada será llevada adelante mayormente utilizando este trabajo.

## DESCRIPCIÓN DEL PROBLEMA:

### RELEVAMIENTO

Una cadena de farmacias nos pide informatizar la operatoria de su negocio. Como resultado de su relevamiento, obtenemos la siguiente información:

La farmacia vende medicamentos y productos de perfumería.

Algunos clientes tienen obra social y otros son privados.

De todos los clientes se registran apellido, nombre, DNI, domicilio (calle, número, localidad, provincia), obra social si la tiene (nombre de la obra social y número de afiliado)

De los empleados se requiere conocer el apellido, nombre, DNI, CUIL, domicilio (calle, número, localidad, provincia) y obra social a la que aporta (nombre de la obra social y número de afiliado).

De cada producto se requiere conocer si es un medicamento o producto de perfumería, una descripción, el laboratorio que lo produce, un código numérico y el precio.

De cada venta se registra: fecha, número de ticket, total de la venta, forma de pago (efectivo, tarjeta o débito), productos vendidos con sus cantidades, precio unitario y total; empleado que atendió al cliente y el empleado que realizó el cobro en caja.

De cada sucursal se conoce que empleados pertenecen a la misma, el domicilio (calle, número, localidad, provincia) y cuál de los empleados es encargado de la misma. El número de ticket se emite con un punto de venta distinto para cada sucursal (el punto de venta se identifica con los primeros cuatro números del ticket fiscal, ej: 0001-00001234 es punto de venta 1)

### REPORTES REQUERIDOS

El cliente desea que el sistema le provea la siguiente información mediante consultas realizadas a la base de datos MongoDB:

1. *Detalle y totales de ventas para la cadena completa y por sucursal, entre fechas.*
2. *Detalle y totales de ventas para la cadena completa y por sucursal, por obra social o privados entre fechas.*
3. *Detalle y totales de cobranza para la cadena completa y por sucursal, por medio de pago y entre fechas.*
4. *Detalle y totales de ventas de productos, total de la cadena y por sucursal, entre fechas, diferenciados entre farmacia y perfumería.*
5. *Ranking de ventas de productos, total de la cadena y por sucursal, entre fechas, por monto.*
6. *Ranking de ventas de productos, total de la cadena y por sucursal, entre fechas, por cantidad vendida.*
7. *Ranking de clientes por compras, total de la cadena y por sucursal, entre fechas, por monto.*
8. *Ranking de clientes por compras, total de la cadena y por sucursal, entre fechas, por cantidad vendida.*

---

## TAREAS A REALIZAR:

---

### ANÁLISIS DEL DOMINIO DEL PROBLEMA.

Diseño de una solución que satisfaga los requisitos del cliente.

Implementación de la solución utilizando:

- Lenguaje de programación java
- Motor de base de datos MongoDB

---

### ANÁLISIS DEL PROBLEMA DESDE UN ENFOQUE RELACIONAL (ERD)

El diagrama ERD pueden construirlo con la herramienta que deseen, incluso la entrega puede ser mediante una fotografía a un documento impreso / escrito manualmente.

---

### ARMADO DE CLASES JAVA QUE GRAFIQUEN LAS RELACIONES ENCONTRADAS (POJOS).

La entrega deberá ser efectuada mediante un archivo ZIP que contenga el proyecto a ejecutar en el IDE que deseen (ECLIPSE, INTELLIJ, etc.).

Se recomienda seguir una metodología de desarrollo top-down, comenzando con el diseño orientado a objetos basado en POJOs, pasando por la construcción de los documentos JSON que serán ingresados en la base de datos MongoDB. Deben tener en cuenta que la construcción de los documentos JSON tienden a ser autocontenidos. Esto significa que por ejemplo si tengo un documento con la información de una farmacia, dentro de ese documento existirá un elemento que es una lista de productos, donde cada uno de sus elementos tiene otros atributos. Este tipo de diseño orientado a documentos es un enfoque DES-NORMALIZADO: Si bien parto de un modelo normalizado graficado en el ERD, al construir las instancias de los objetos y serializando los documentos termino con una salida desnormalizada de la información, dado que por ejemplo podría tener un elemento “producto” repetido en varias instancias de “ventas”.

---

### CONSTRUCCIÓN DE MECANISMOS DE SERIALIZACIÓN DE LA INFORMACIÓN (PARA EL ARMADO DE LOS DOCUMENTOS JSON)

En el mismo proyecto de la entrega anterior deberán incluir métodos de creación de instancias de las clases anteriormente confeccionadas y la serialización de la información utilizando GSON o Jackson.

---

### ALMACENAMIENTO DE LOS DOCUMENTOS JSON EN MONGO DB

Deberán “subir” los diferentes documentos contruidos en el paso anterior a la base MongoDB previamente instalada

---

### CONSTRUCCIÓN DE CONSULTAS A LA BASE DE DATOS MONGO DB.

En el proyecto deberán construir los métodos que resuelven las consultas JSON en la base de datos MongoDB.

A fin de poder realizar las consultas, de deberá generar un set de datos conteniendo como mínimo 3 sucursales, 10 clientes, 3 vendedores por sucursal, uno de ellos con categoría encargado; 10 productos (7 medicamentos y 3 perfumería), y un promedio de 30 ventas por sucursal. Es deseable que haya variación en las cantidades de venta por sucursal (+/- 20%). Las ventas deben tener un promedio mínimo de 1,5 productos.

## ENTREGAS DEL TRABAJO

A continuación, iremos editando la sección agregando las sucesivas entregas requeridas del trabajo.

### ENTREGA 01

Se requiere la entrega del modelo de entidad/relación (ERD) que surge del análisis del enunciado. En lista:

1. Análisis del problema
2. Construcción del ERD

### ENTREGA 02

Se requiere la entrega de las clases *POJOs* en java, el proyecto de construcción de la estructura de datos utilizando estas clases, la instanciación con los datos ejemplo y su posterior serialización en un único documento JSON, esto último se entrega en un documento *.json*. En lista:

1. En base al ERD de la entrega anterior construir los POJOs necesarios
2. Instanciar la estructura de datos con los datos de prueba
3. Serializar la estructura completa y construir el documento JSON a entregar con el proyecto de clases anterior.

La idea de esta entrega es entender que el documento termina siendo una única estructura JSON donde luego efectuaremos las consultas que serán requeridas posteriormente. Este documento termina siendo una versión “desnormalizada” del ERD inicial, o sea que partimos de un problema a analizar, efectuamos su estudio y lo normalizamos, luego armamos las estructuras de datos y al final de este paso terminamos en un documento desnormalizado.

Esta desnormalización la van a visualizar por ejemplo al detectar que un mismo producto va a aparecer anidado en diferentes listas de diferentes compras, donde los datos del producto podrían verse repetidos. Esta es la esencia de este tipo de bases de datos y necesitamos visualizar este punto.

*Pista: No queremos tener diferentes listas JSON con las diferentes colecciones de elementos modelados en los POCOs, porque seria mantener el esquema normalizado del que estamos “escapando”.*

### ENTREGA 03

Para aprobar el trabajo se requiere efectuar la carga de las ventas en la base de datos MongoDB (solo las ventas) y sobre ellas efectuar las consultas 1 y 4 de la lista presentada en el enunciado. En caso de realizar el resto de las consultas, será tenido en cuenta para la nota de la aprobación de la cursada.

Debe tenerse en cuenta que solo se requiere la carga de Ventas, y las consultas deben realizarse sobre ese diccionario únicamente.

La carga de datos se debe realizar con un proceso en Java, y esa clase debe ser incluida en el proyecto entregado.

No obstante, las consultas pueden ser entregadas en un archivo aparte si es que las efectúan directamente en el cliente de MongoDB, o pueden estar embebidas en la clase java si es que desean resolverlas desde ahí.