

Explorando el Futuro: Predicción del Comportamiento de Medicamentos Vitales No Disponibles en Colombia

Stephany Enciso¹, Alexander Victoria², Maria Camila Prada³

Universidad Central
Maestría en Analítica de Datos
Curso de Bases de Datos
Bogotá, Colombia

¹*sencisom@ucentral.edu.co*, ²*avictoriag@ucentral.edu.co*, ³*mpradai1@ucentral.edu.co*

November 24, 2023

Contents

1	Introducción	3
2	Características del proyecto de investigación que hace uso de Bases de Datos	3
2.1	Titulo del proyecto de investigación	3
2.2	Objetivo general	3
2.2.1	Objetivos especificos	3
2.3	Alcance	4
2.4	Pregunta de investigación	4
2.5	Hipotesis	4
3	Reflexiones sobre el origen de datos e información	5
3.1	¿Cual es el origen de los datos e información ?	5
3.2	¿Cuales son las consideraciones legales o eticas del uso de la información?	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?	6
3.4	¿Que espera de la utilización de un sistema de Bases de Datos para su proyecto?	6

4	Diseño del Modelo de Datos del SMBD (Sistema Manejador de Bases de Datos)	6
4.1	Características del SMBD (Sistema Manejador de Bases de Datos) para el proyecto	6
4.2	Diagrama modelo de datos	6
4.3	Imágenes de la Base de Datos	8
4.4	Código SQL - lenguaje de definición de datos (DDL)	9
4.5	Código SQL - Manipulación de datos (DML)	11
4.6	Código SQL + Resultados: Vistas	11
4.7	Código SQL + Resultados: Triggers	17
4.8	Código SQL + Resultados: Funciones	19
4.9	Código SQL + Resultados: procedimientos almacenados	22
5	Bases de Datos No-SQL	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	27
6	Aplicación de ETL (Extract, Transform, Load) y Bodega de Datos	30
6.1	Extracción	30
6.2	Transformación	32
6.3	Carga	35
7	Proximos pasos	43
8	Lecciones aprendidas	43
9	Bibliografía	44

1 Introducción

Los medicamentos, tal como lo indica la OMS son todos aquellos fármacos usados para la prevención, tratamiento y alivio de una amplia variedad de problemas de salud, esto incluye enfermedades, trastornos de salud mental y de otro tipo. El acceso a los medicamentos apropiados tiene efectos significativos en la salud de los individuos, algunos de estos garantizan una mejor calidad de vida para los pacientes siendo el único tratamiento disponible o en el caso de cuidados paliativos el único alivio para el dolor.

(OMS) El Instituto Nacional de Vigilancia de Medicamentos y Alimentos (INVIMA) es el ente regulador encargado de la vigilancia sanitaria y la regulación de dispositivos médicos y medicamentos a través del mecanismo de registro sanitario el cual debe ser solicitado por cada fabricante o importador con el fin de autorizar el ingreso, comercialización y uso de los medicamentos en el territorio Colombiano.

INVIMA comparte la información Medicamentos Vitales No Disponibles en Colombia, en la cual reúne información crucial sobre medicamentos que no sólo son indispensables, sino también irremplazables, para preservar la vida y mitigar el sufrimiento de pacientes individuales o grupos de pacientes que enfrentan circunstancias particulares. Estos medicamentos, cuya relevancia es incuestionable, pueden verse limitados en su disponibilidad debido a factores de baja rentabilidad en su comercialización o insuficiencia en las cantidades disponibles en el país.

2 Características del proyecto de investigación que hace uso de Bases de Datos

2.1 Título del proyecto de investigación

Explorando el Futuro: Predicción del Comportamiento de Medicamentos Vitales No Disponibles en Colombia

2.2 Objetivo general

Determinar la existencia de patrones temporales de escasez, identificando intervalos en los que la falta de disponibilidad de los medicamentos vitales es más recurrente y analizar variaciones estacionales a través del uso de series de tiempo.

2.2.1 Objetivos específicos

- Crear una base de datos en Oracle Developer, relacionando información de los medicamentos vitales no disponibles en Colombia.
- Realizar un análisis exploratorio de los datos con el fin de identificar patrones, tendencias y posibles correlaciones entre la disponibilidad de los cinco medicamentos vitales principales por principio activo 1 y otras variables relevantes.

- Identificar patrones temporales específicos de escasez para los cinco medicamentos principales, determinando cuándo y cómo se producen con mayor frecuencia las situaciones de falta de disponibilidad de estos medicamentos.
- Desarrollar modelos ARIMA individualizados para cada uno de los cinco medicamentos principales por principio activo 1, con el propósito de prever la disponibilidad de estos medicamentos en un horizonte de pronóstico de 4 meses. Estos modelos se basarán en datos históricos y considerarán las tendencias y patrones identificados en el análisis de series temporales, con el objetivo de proporcionar pronósticos precisos y útiles para la gestión de inventarios y la toma de decisiones estratégicas.

2.3 Alcance

Este proyecto se enfoca en abordar la escasez de medicamentos vitales en Colombia. Se creará una base de datos en Oracle Developer utilizando datos proporcionados por el INVIMA (Instituto Nacional de Vigilancia de Medicamentos y Alimentos) de Colombia. Esta base de datos desempeñará un papel fundamental en el análisis y la toma de decisiones del estudio.

A continuación, se llevará a cabo un análisis exploratorio de los datos para descubrir patrones y tendencias en la disponibilidad de los cinco medicamentos vitales principales, clasificados por principio activo 1, junto con otras variables relevantes. Este análisis arrojará luz sobre las causas subyacentes de la escasez de medicamentos.

El proyecto se centrará en la identificación de patrones temporales específicos de escasez para estos cinco medicamentos, lo que permitirá determinar cuándo y cómo ocurren con mayor frecuencia las situaciones de falta de disponibilidad.

Por último, se desarrollarán modelos ARIMA personalizados para cada uno de los cinco medicamentos principales por principio activo 1, con el objetivo de prever la disponibilidad de estos medicamentos en un horizonte de pronóstico de 4 meses. Estos modelos se basarán en datos históricos y considerarán las tendencias y patrones identificados en el análisis de series temporales, proporcionando pronósticos precisos para la gestión de inventarios y la toma de decisiones estratégicas en el sector de la salud en Colombia.

2.4 Pregunta de investigación

¿Cuáles son los patrones temporales de escasez de medicamentos vitales en Colombia, específicamente para los cinco medicamentos principales por principio activo 1, y cómo pueden predecirse con precisión para un horizonte de pronóstico de 4 meses utilizando modelos ARIMA?

2.5 Hipotesis

La intermitencia en el suministro de medicamentos esenciales en Colombia se debe a una combinación de factores, incluyendo limitaciones de inventario

ligados a problemas logísticos en la distribución de los mismos, costos de los medicamentos y ausencia de registro INVIMA. Se puede anticipar que un análisis detallado de estos factores proporciona una comprensión más precisa de las causas subyacentes de la escasez, lo que ayudará al planteamiento de estrategias para mejorar la disponibilidad de medicamentos esenciales en Colombia.

3 Reflexiones sobre el origen de datos e información

En este proyecto, los datos e información se obtienen del portal de datos abiertos del gobierno colombiano, una plataforma en línea que ofrece acceso a una amplia variedad de datos de instituciones gubernamentales. El propósito principal de este portal es fomentar la transparencia, la participación ciudadana y la innovación al poner a disposición del público información en formatos accesibles y reutilizables, lo que lo convierte en una fuente confiable y oficial.

Sin embargo, el uso de estos datos conlleva responsabilidades legales y éticas. Al ser parte de las bases de datos abiertas del Instituto Nacional de Vigilancia de Medicamentos y Alimentos (INVIMA), los usuarios deben comprometerse a utilizar la información de manera ética y legal, evitando el uso fraudulento o ilegal de los datos y respetando la privacidad de las personas cuyos datos puedan estar incluidos.

Un desafío clave en este proyecto es garantizar la calidad y consolidación de los datos. Esto incluye tareas como la identificación de datos duplicados, la normalización de variables y la caracterización detallada de cada conjunto de datos. La calidad de los datos es esencial para obtener resultados confiables y precisos en los análisis, mientras que la consolidación permite organizar la información de manera coherente y eficiente.

Se espera que el sistema de Bases de Datos utilizado en el proyecto permita estandarizar, agrupar y ordenar eficientemente la información del INVIMA. Esto facilitará la realización de análisis mediante consultas de información sencillas y prácticas, lo que a su vez optimizará la toma de decisiones informadas y la formulación de políticas basadas en datos sólidos. En resumen, este proyecto se basa en datos gubernamentales confiables y busca abordar desafíos de calidad y consolidación para aprovechar al máximo la información y lograr resultados efectivos.

3.1 ¿Cual es el origen de los datos e información ?

La base de datos fue obtenida del portal de datos abiertos, el cual es una plataforma en línea proporcionada por el gobierno colombiano que ofrece acceso a una amplia gama de datos y conjuntos de datos de diversas instituciones y entidades gubernamentales. Este portal tiene como objetivo fomentar la

transparencia, la participación ciudadana y la innovación al poner a disposición del público información en formatos accesibles y reutilizables.

3.2 ¿Cuales son las consideraciones legales o eticas del uso de la información?

Al ser parte de las bases de datos abiertos del INVIMA, los usuarios deben usarlos de manera ética y legal. Por lo que no se puede utilizar la información de manera fraudulenta, engañosa o para fines ilegales.

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?

Los retos a los que nos enfrentamos es garantizar la calidad y consolidación de los datos, desde la gestión de datos duplicados hasta la normalización de los mismos, pasando por la caracterización de cada una de las variables para posterior clasificación dentro del modelo de análisis planteado.

3.4 ¿Que espera de la utilización de un sistema de Bases de Datos para su proyecto?

Se espera estandarizar, agrupar y ordenar de manera eficiente la información proveniente del INVIMA con el fin de realizar los respectivos análisis con un esquema de consultas de información sencillo y práctico.

4 Diseño del Modelo de Datos del SMBD (Sistema Manejador de Bases de Datos)

4.1 Características del SMBD (Sistema Manejador de Bases de Datos) para el proyecto

En este proyecto, se utilizaron sistemas manejadores de bases de datos (SMBD) basados en Oracle Developer, junto con herramientas de modelado de datos como Oracle Data Modeler. Estas herramientas desempeñaron un papel muy importante al permitir diseñar estructuras de datos y garantizar la integridad de los datos.

4.2 Diagrama modelo de datos

A continuación, se presenta el diagrama de la base de datos que representa la estructura y las relaciones de los datos en este sistema. En este diagrama, se pueden identificar tres tablas principales que desempeñan un papel fundamental en la organización y gestión de la información. Este diagrama proporcionará una visión general clara de cómo se almacenan y se conectan los datos en el contexto de nuestro proyecto.

Por otro lado, se presenta el modelo logico que describe la estructura de la base de datos de la siguiente manera:

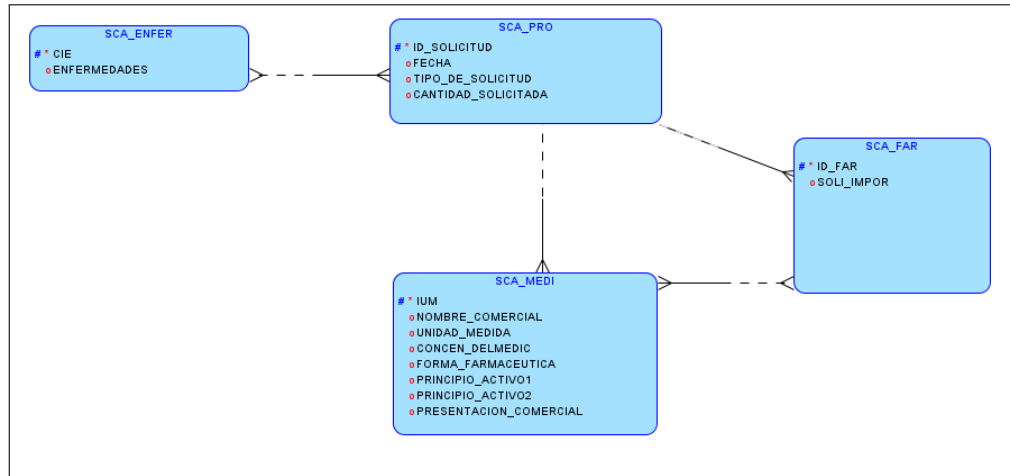


Figure 1: Modelo logico.

De igual forma se expone el modelo relacional, de la siguiente manera:

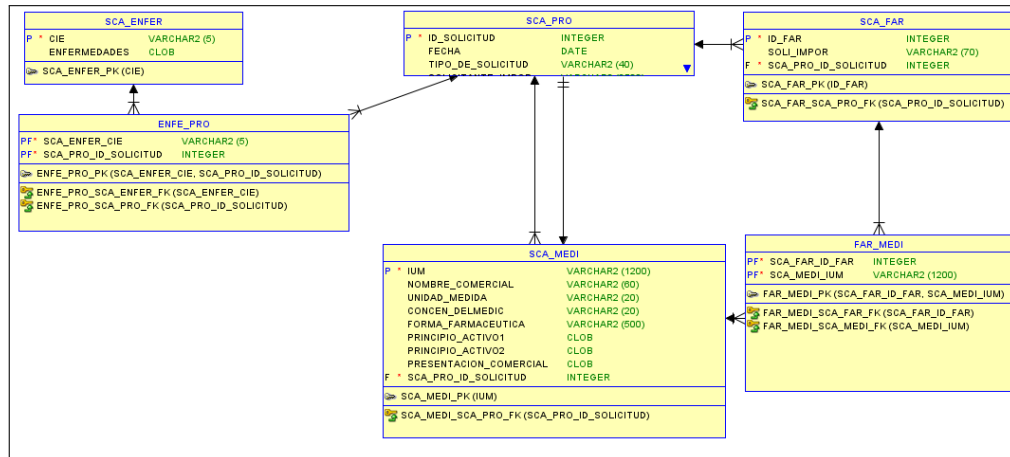


Figure 2: Modelo relacional.

4.3 Imágenes de la Base de Datos

En el contexto de nuestro proyecto, la implementación de una base de datos es el pilar central para la gestión efectiva de la información. Esta base de datos permitirá la organización y almacenamiento estructurado de datos. A través de tablas bien definidas, como se detalla a continuación:

4.3.1 Tabla Enfermedades.

Resultado de la Consulta X

Se han recuperado 50 filas en 1,034 segundos

CIE II ENFERMEDADES

- 1 C463 SARCOMA DE KAPOSI DE LOS GANGLIOS LINFATICOS
- 2 D696 TRONCOSTOMIOPATIA NO ESPECIFICADA
- 3 E759 TRASTORNO DEL METABOLISMO DE LOS AMINOACIDOS, NO ESPECIFICADO
- 4 E878 OTROS TRASTORNOS DEL EQUILIBRIO DE LOS ELECTROLITOS Y DE LOS LIQUIDOS, NO CLASIFICADOS EN OTRA PARTE
- 5 I471 TAQUICARDIA SUPRAVENTRICULAR
- 6 D45X POLICITEMIA VERA
- 7 W419 ENFERMEDAD INFLAMATORIA DE LA PROSTATA, NO ESPECIFICADA
- 8 T561 EFECTO TOXICO DE METALES: MERCURIO Y SUS COMPUESTOS
- 9 B340 INFECCION DEBIDA A ADENOVIRUS, SIN OTRA ESPECIFICACION
- W E248 SINDROME DE CUSHING, NO ESPECIFICADO
- 11 C129 TUMOR MALIGNO DEL COLON, PARTE NO ESPECIFICADA
- 12 C474 TUMOR MALIGNO DE LOS NERVIOS PERIFERICOS DEL ABDOMEN
- 13 Z225 PORTADOR DE HEPATITIS VIRAL
- 14 G10X ENFERMEDAD DE HUNTINGTON
- 15 T780 CHOQUE ANAFILACTICO DEBIDO A REACCION ADVERSA A ALIMENTOS
- 16 I200 ANGINA INESTABLE
- 17 E839 TRASTORNO DEL METABOLISMO DE LOS MINERALES, NO ESPECIFICADO
- 18 B04X VIRUELA DE LOS MONOS
- 19 Q818 OTRAS EPIDERMOLISIS BULLOSAS
- 20 C941 ENFERMEDAD DE SEZAI
- 21 C740 TUMOR MALIGNO DE LA CORTEZA DE LA GLANDULA SUPRARENAL
- 22 D748 METANEMOGLOBINEMIA, NO ESPECIFICADA
- 23 M100 GOTA IDIOPATICA
- 24 W40X HIPERPLASIA DE LA PROSTATA
- 25 E268 OTROS TIPOS DE HIPERALDOSTERONISMO
- 26 L123 EPIDERMOLISIS BULLOSA ADQUIRIDA
- 27 E669 OBESIDAD, NO ESPECIFICADA

Figure 3: Vista preliminar - Tabla Enfermedades.

4.3.2 Tabla Medicamentos.

[illegible]

Figure 4: Vista preliminar - Tabla Medicamentos.

4.3.3 Tabla Solicitud De Productos.

ID SOLICITUD	FECHA	TIPO DE SOLICITUD	SOLICITANTE_IMPOR	CANTIDAD SOLICITADA	SCA_ENFER_CIE	SCA_MEDI_UM
1	678 11/04/22	PACIENTE ESPECIFICO	IMPROVIT MEDICAL SAS	72 Q412	1T1001911004100CAJAS X 10 TABLETAS	
2	678 11/03/22	PACIENTE ESPECIFICO	HAUTIS SAS - VALENTECH PHARMA COLOMB...	24 Q710	1E1029581000101CAJAS X 1 VIAL X 10 mL	
3	678 11/03/22	PACIENTE ESPECIFICO	HAUTIS SAS - VALENTECH PHARMA COLOMB...	48 Q710	1E1029581000102CAJAS X 1 VIAL X 2 mL	
4	678 11/03/22	PACIENTE ESPECIFICO	MEDICAR SUPPLY SAS	8 C509	1T1041301000100CAJA X 1 VIAL	
5	678 11/02/22	PACIENTE ESPECIFICO	AUDIFARMA S.A.	6 Q710	1A1001711000100CAJAS X 30 SOBRES	
6	680 11/02/22	PACIENTE ESPECIFICO	LABORATORIOS BIODAS S.A.	3 B472	1I1038001000101CAJAS X 180 TABLETAS	
7	682 11/02/22	PACIENTE ESPECIFICO	AUDIFARMA S.A.	18 Q710	1A1001711000100CAJAS X 30 SOBRES	
8	682 11/02/22	URGENCIA CLINICA	GESTIFARMA S.A.S.	3 C341	1I1034901000100FRASCO X 30 TABLETAS	
9	683 11/01/22	PACIENTE ESPECIFICO	IMPROVIT MEDICAL SAS	63 Q412	1T1001911004100CAJAS, CADA CAJA CON 10 TABLETAS	
10	684 11/01/22	MÁS DE UN PACIENTE	STERIUS MARKETING S.A.S.	200 C900	1M0174210011000H (1)VIAL CON POLVO LIOFILIZADO + UN (1) VIAL CON DIL	
11	685 11/01/22	MÁS DE UN PACIENTE	STERIUS MARKETING S.A.S.	200 C900	1M0174210011000H (1)VIAL CON POLVO LIOFILIZADO + UN (1) VIAL CON DIL	
12	686 11/01/22	PACIENTE ESPECIFICO	AUDIFARMA S.A.	18 Q710	1A1001711000100CAJAS X 30 SOBRES	
13	687 28/10/22	PACIENTE ESPECIFICO	IMED ORIHAN S.A.S	10 E728	1C1024071000100CAJAS X 30 SOBRES DE 6 g	
14	689 27/10/22	URGENCIA CLINICA	BIO MEDICAL PHARMA S.A.S	4 B048	1C1000151003100CAJAS X 1 VIAL X 5 mL	
15	689 27/10/22	PACIENTE ESPECIFICO	AUDIFARMA S.A.	9 Q774	1P1041901000100CAJA 110 VIALES	
16	690 27/10/22	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	96 Q710	1E1029581000102CAJA X 1 VIAL X 2 mL	
17	691 27/10/22	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	24 Q710	1E1029581000101CAJA X 1 VIAL X 10 mL	
18	692 26/10/22	PACIENTE ESPECIFICO	AUDIFARMA S.A.	12 Q710	1A1001711000100CAJAS X 30 SOBRES	
19	693 26/10/22	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	144 Q710	1C1039371000100CAJAS X 1 VIAL X 2mL	
20	694 26/10/22	PACIENTE ESPECIFICO	AUDIFARMA S.A.	18 Q710	1A1001711000100CAJAS X 30 SOBRES	
21	695 26/10/22	PACIENTE ESPECIFICO	AUDIFARMA S.A.	12 Q710	1A1001711000100CAJAS X 30 SOBRES	
22	696 26/10/22	PACIENTE ESPECIFICO	LABORATORIOS BIODAS S.A.	3 E840	2E1032451000100CAJAS X 84 TABLETAS	
23	697 26/10/22	PACIENTE ESPECIFICO	AUDIFARMA S.A.	12 Q710	1A1001711000100CAJAS X 30 SOBRES	
24	698 26/10/22	PACIENTE ESPECIFICO	GLOBAL SERVICE PHARMACEUTICAL S.A.S.	12 C341	1B1004001000100FRASCO X 100 CAPSULAS	
25	699 26/10/22	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	148 Q710	1G1036141000100CAJAS X UN (1) VIAL DE 2 mL	
26	700 25/10/22	PACIENTE ESPECIFICO	AUDIFARMA S.A.	6 Q710	1A1001701000100CAJAS X 30 SOBRES	
27	701 21/10/22	MÁS DE UN PACIENTE	VALENTECH PHARMA COLOMBIA SAS	100 C740	1M1000341000100CAJAS X 100 TABLETAS	

Figure 5: Vista preliminar - Tabla Solicitud de Productos.

4.4 Código SQL - lenguaje de definición de datos (DDL)

En esta sección, se presentará el código SQL del lenguaje de definición de datos (DDL) que involucra la creación de tres tablas en una base de datos. Estas tablas son: **sca_enfer**, **sca_medi**, **sca_pro**, y **sca_far**. El fragmento de código SQL que se detallará a continuación:

CODIGO

```
CREATE TABLE enf_pro (
    sca_enfer_cie          VARCHAR2(5) NOT NULL,
    sca_pro_id_solicitud   INTEGER NOT NULL
);

ALTER TABLE enf_pro ADD CONSTRAINT enf_pro_pk
PRIMARY KEY ( sca_enfer_cie ,
sca_pro_id_solicitud );

CREATE TABLE far_medi (
    sca_far_id_far         INTEGER NOT NULL,
    sca_medi_ium           VARCHAR2(1200) NOT NULL
);

ALTER TABLE far_medi ADD CONSTRAINT far_medi_pk
PRIMARY KEY ( sca_far_id_far ,
sca_medi_ium );
```

```

CREATE TABLE sca_enfer (
    cie          VARCHAR2(5) NOT NULL,
    enfermedades CLOB
);

ALTER TABLE sca_enfer ADD CONSTRAINT sca_enfer_pk
PRIMARY KEY ( cie );

CREATE TABLE sca_far (
    id_far          INTEGER NOT NULL,
    soli_imple      VARCHAR2(70),
    sca_pro_id_solicitud INTEGER NOT NULL
);
ALTER TABLE sca_far ADD CONSTRAINT sca_far_pk
PRIMARY KEY ( id_far );

CREATE TABLE sca_medi (
    ium          VARCHAR2(1200) NOT NULL,
    nombre_comercial VARCHAR2(60),
    unidad_medida  VARCHAR2(20),
    concen_delmedic VARCHAR2(20),
    forma_farmaceutica VARCHAR2(500),
    principio_activo1 CLOB,
    principio_activo2 CLOB,
    presentacion_comercial CLOB,
    sca_pro_id_solicitud INTEGER NOT NULL
);

```

CODIGO

```

ALTER TABLE sca_pro ADD CONSTRAINT sca_pro_pk
PRIMARY KEY ( id_solicitud );

ALTER TABLE sca_pro
ADD CONSTRAINT sca_pro_sca_enfer_fk FOREIGN KEY
( sca_enfer_cie )
REFERENCES sca_enfer ( cie );

ALTER TABLE sca_pro
ADD CONSTRAINT sca_pro_sca_medi_fk FOREIGN KEY
( sca_medi_ium )
REFERENCES sca_medi ( ium );

SELECT * FROM SCA_ENFER;

```

4.5 Código SQL - Manipulación de datos (DML)

En esta sección, se abordará el lenguaje SQL para la manipulación de datos (DML), tal como se detalla a continuación:

```
CODIGO

—DML DE BASES DE DATOS

—Eliminacion de NULL en cualquier variable

DELETE FROM SCA_PRO
WHERE FECHA IS NULL OR
      TIPO_DE_SOLICITUD IS NULL OR
      SOLICITANTE_IMPOR IS NULL OR
      CANTIDAD_SOLICITADA IS NULL;

—Eliminacion de NULL en cualquier variable

DELETE FROM SCA_ENFER
WHERE CIE IS NULL OR
      ENFERMEDADES IS NULL;
```

4.6 Código SQL + Resultados: Vistas

En el contexto de este informe, se expondrán las diversas vistas que se han desarrollado, estas vistas ofrecen perspectivas específicas y estructuradas de los datos, lo que facilita una comprensión más profunda. A continuación, se describen detalladamente estas vistas.

```
CODIGO

—Vistas SCA_PRODUCTOS.SOLICITADOS
— Cantidades solicitadas por a o y mes

CREATE VIEW SOLICITADAS_POR_AO AS

SELECT
      EXTRACT(YEAR FROM FECHA) AS A O ,
      EXTRACT(MONIH FROM FECHA) AS MES,
      SUM(CANTIDAD_SOLICITADA) AS CANTIDADES_SOLICITADAS
FROM SCA_PRO
GROUP BY EXTRACT(YEAR FROM FECHA) , EXTRACT(MONIH
FROM FECHA);
```

Resultado de la Consulta x		Resultado de la Consulta 1 x	
		Se han recuperado 50 filas en 0,23 segundos	
	AÑO	MES	CANTIDADES_SOLICITADAS
1	2022	8	117665
2	2023	1	72226
3	2021	10	127355
4	2021	1	6204
5	2020	1	412387
6	2020	10	140214
7	2021	4	17294
8	2019	11	634458
9	2019	7	5388
10	2019	6	8178

Figure 6: Vista - Cantidades solicitadas por año y mes.

CODIGO

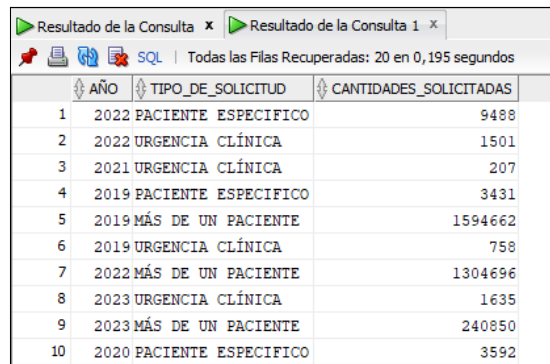
— Cantidades solicitadas por año y tipo_de_solicitud

```
CREATE VIEW SOLICITDAS_AO_TIPO_SOLCITUD AS
SELECT
    EXTRACT(YEAR FROM FECHA) AS AÑO ,
    TIPO_DE_SOLICITUD ,
    SUM(CANTIDAD_SOLICITADA) AS CANTIDADES_SOLICITADAS
FROM SCA_PRO
GROUP BY EXTRACT(YEAR FROM FECHA) , TIPO_DE_SOLICITUD ;
```

CODIGO

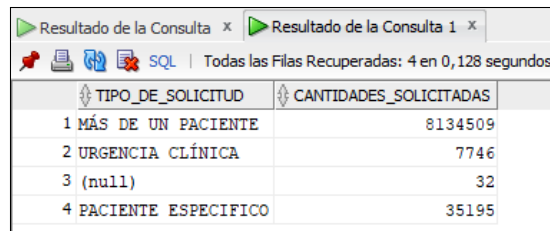
—Discriminaci n de cantidades solicitadas por tipo de solicitud

```
CREATE VIEW SOLICITADAS.POR.TIPO.SOLICITUD AS
SELECT
    TIPO_DE_SOLICITUD,
    SUM(CANTIDAD.SOLICITADA) AS CANTIDADES.SOLICITADAS
FROM SCA_PRO
GROUP BY TIPO_DE_SOLICITUD;
```



	AÑO	TIPO_DE_SOLICITUD	CANTIDADES_SOLICITADAS
1	2022	PACIENTE ESPECIFICO	9488
2	2022	URGENCIA CLÍNICA	1501
3	2021	URGENCIA CLÍNICA	207
4	2019	PACIENTE ESPECIFICO	3431
5	2019	MÁS DE UN PACIENTE	1594662
6	2019	URGENCIA CLÍNICA	758
7	2022	MÁS DE UN PACIENTE	1304696
8	2023	URGENCIA CLÍNICA	1635
9	2023	MÁS DE UN PACIENTE	240850
10	2020	PACIENTE ESPECIFICO	3592

Figure 7: Vista - Cantidades solicitadas por año y tipo de solicitud.



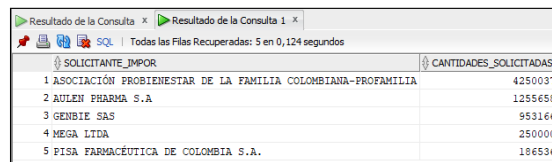
	TIPO_DE_SOLICITUD	CANTIDADES_SOLICITADAS
1	MÁS DE UN PACIENTE	8134509
2	URGENCIA CLÍNICA	7746
3	(null)	32
4	PACIENTE ESPECIFICO	35195

Figure 8: Vista - Discriminación de cantidades solicitadas por tipo de solicitud.

CODIGO

—TOP 5 de los solicitantes por cantidades

```
CREATE VIEW TOP_SOLICITANTES_CANTIDADES AS
SELECT
    SOLICITANTE_IMPORT,
    SUM(CANTIDAD_SOLICITADA) AS CANTIDADES_SOLICITADAS
FROM SCA_PRO
GROUP BY SOLICITANTE_IMPORT
ORDER BY SUM(CANTIDAD_SOLICITADA) DESC
FETCH FIRST 5 ROWS ONLY;
```



The screenshot shows a SQL query result window with two tabs: 'Resultado de la Consulta' and 'Resultado de la Consulta 1'. The active tab displays a table with two columns: 'SOLICITANTE_IMPORT' and 'CANTIDADES_SOLICITADAS'. The table contains five rows of data, representing the top 5 applicants by quantity.

SOLICITANTE_IMPORT	CANTIDADES_SOLICITADAS
1 ASOCIACIÓN PROBIENESTAR DE LA FAMILIA COLOMBIANA-PROFAMILIA	4250037
2 AULEN PHARMA S.A	1255658
3 GENIE SAS	953166
4 MEGA LTDA	250000
5 PISA FARMACEUTICA DE COLOMBIA S.A.	186536

Figure 9: Vista - TOP 5 de los solicitantes por cantidades.

CODIGO

—Vistas SCA_MEDICAMENTOS

—Medicamentos por forma farmaceutica

```
CREATE VIEW MEDICAMENTOS.VS_FARMACEUTICA AS
SELECT
    FORMA_FARMACEUTICA,
    COUNT(IUM) AS MEDICAMENTOS
FROM SCA_MEDI
GROUP BY FORMA_FARMACEUTICA;
```

Resultado de la Consulta x Resultado de la Consulta 1 x

SQL | Todas las Filas Recuperadas: 10 en 0,131 segundos

	NOMBRE_COMERCIAL	MEDICAMENTOS
1	NO REPORTADO	108
2	STRENSIQ	24
3	TRANSLARNA	15
4	FIBRYGA	13
5	MYALEPT	11
6	CRYSVITA	10
7	EXONDYS 51	9
8	ICLUSIG	8
9	ELMIRON	7
10	DUPIXENT	7

Figure 10: Vista - Medicamentos por forma farmaceutica.

CODIGO

—Top 10 de medicamentos por nombre comercial

```
CREATE VIEW MEDICAMENTOS.VS_NOMBRE_COMERCIAL AS
SELECT
NOMBRE_COMERCIAL,
COUNT(IUM) AS MEDICAMENTOS
FROM SCA_MEDI
GROUP BY NOMBRE_COMERCIAL
ORDER BY COUNT(IUM) DESC
FETCH FIRST 10 ROWS ONLY;
```

Resultado de la Consulta x Resultado de la Consulta 1 x

SQL | Todas las Filas Recuperadas: 41 en 0,143 segundos

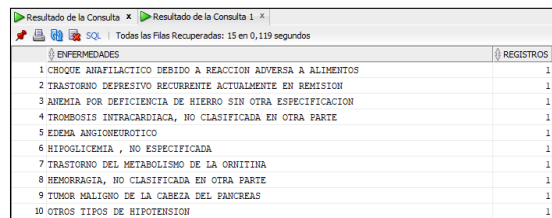
	FORMA_FARMACEUTICA	MEDICAMENTOS
1	TABLETA	184
2	SUSPENSION INYECTABLE	32
3	SUSPENSION	1
4	GEL	5
5	CONCENTRADO PARA SOLUCION PARA PERFUSION	1
6	POLVO PARA RECONSTITUIR A SOLUCION ORAL	1
7	POLVO LIOFILIZADO PARA RECONSTITUIR A SUSPENSION PARA INSTILACION VESICAL	1
8	POLVO LIOFILIZADO PARA RECONSTITUIR A SOLUCION INYECTABLE	164
9	POLVO LIOFILIZADO PARA RECONSTITUIR A SUSPENSION PARA INSTILACION	1
10	POLVO PARA RECONSTITUIR EN SOLUCION ORAL	1

Figure 11: Vista - Top 10 de medicamentos por nombre comercial.

CODIGO

—Vistas SCA_ENFERMEDADES

```
CREATE VIEW REGISTROS_POR_ENFERMEDADES AS
SELECT
ENFERMEDADES,
COUNT(CIE) AS REGISTROS
FROM SCA_ENFER
GROUP BY ENFERMEDADES
ORDER BY COUNT(CIE) DESC
FETCH FIRST 15 ROWS ONLY;
```



Resultado de la Consulta x Resultado de la Consulta 1 x

Todas las Filas Recuperadas: 15 en 0,119 segundos

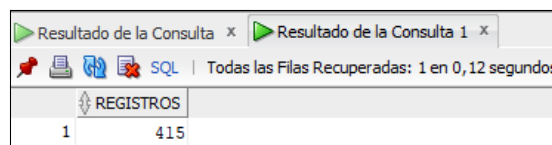
ENFERMEDADES	REGISTROS
1 CHOQUE ANAFILACTICO DEBIDO A REACCION ADVERSA A ALIMENTOS	1
2 TRASTORNO DEPRESIVO RECURRENTE ACTUALMENTE EN REMISION	1
3 ANEMIA POR DEFICIENCIA DE HIERRO SIN OTRA ESPECIFICACION	1
4 TROMBOSIS INTRACARDIACA, NO CLASIFICADA EN OTRA PARTE	1
5 EDEMA ANGIONEUROTICO	1
6 HIPOGLICEMIA , NO ESPECIFICADA	1
7 TRASTORNO DEL METABOLISMO DE LA ORNITINA	1
8 HEMORRAGIA, NO CLASIFICADA EN OTRA PARTE	1
9 TUMOR MALIGNO DE LA CABEZA DEL PANCREAS	1
10 OTROS TIPOS DE HIPOTENSION	1

Figure 12: Vista - Enfermedades.

CODIGO

—Numero registros

```
CREATE VIEW REGISTROS_TOTALES AS
SELECT
COUNT(CIE) AS REGISTROS
FROM SCA_ENFER;
```



Resultado de la Consulta x Resultado de la Consulta 1 x

Todas las Filas Recuperadas: 1 en 0,12 segundos

REGISTROS
1 415

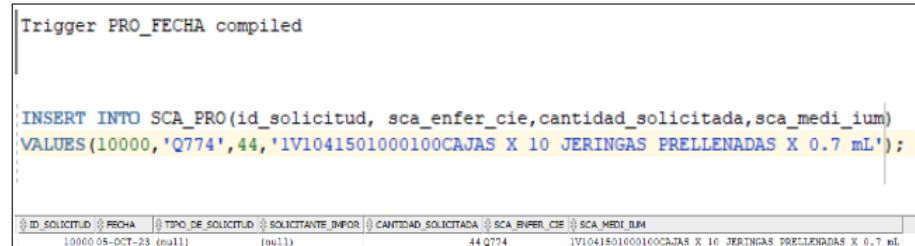
Figure 13: Vista - Numero de registros.

4.7 Código SQL + Resultados: Triggers

En esta sección, se abordarán los triggers implementados en las bases de datos. Los triggers son componentes esenciales que permiten automatizar ciertas acciones o procesos en respuesta a eventos específicos en la base de datos. A continuación, se describirán en detalle estos triggers, de la siguiente manera:

CODIGO

```
CREATE OR REPLACE TRIGGER PRO_FECHA
BEFORE INSERT ON SCA_PRO
FOR EACH ROW
NEW.FECHA:= SYSTIMESTAMP
END;
```



Trigger PRO_FECHA compiled

```
INSERT INTO SCA_PRO(id_solicitud, sca_enfer_cie,cantidad_solicitada,sca_medi_ium)
VALUES(10000,'Q774',44,'1V1041501000100CAJAS X 10 JERINGAS PRELENADAS X 0.7 mL');
```

ID_SOLICITUD	FECHA	TIPO_DE_SOLICITUD	SOLICITANTE_INFOP	CANTIDAD_SOLICITADA	SCA_RIFER_CIE	SCA_MEDI_UM
10000	06-OCT-23 (null)	(null)		44	Q774	1V1041501000100CAJAS X 10 JERINGAS PRELENADAS X 0.7 mL

Figure 14: Trigger PROFECHA.

CODIGO

```
CREATE OR REPLACE TRIGGER PROD_VAL
BEFORE INSERT ON SCA-PRO
FOR EACH ROW
BEGIN
    IF :NEW.CANTIDAD.SOLICITADA < 0 THEN
        RAISE_APPLICATION_ERROR(-20001, 'La cantidad no
        puede ser negativa '
    END IF
END;
```

```

Trigger PROD_VAL compiled

INSERT INTO SCA_PRO(id_solicitud, sca_enfer_cie,cantidad_solicitada,sca_medi_ium)
VALUES(11220,'Q774',-10.5,'1V1041501000100CAJAS X 10 JERINGAS PRELLENADAS X 0.7 mL');

Error at Command Line : 34 Column : 13
Error report -
SQL Error: ORA-20001: La cantidad no puede ser negativa
ORA-06512: at "ADMIN.PROD_VAL", line 3
ORA-04088: error during execution of trigger 'ADMIN.PROD_VAL'

```

Figure 15: Trigger Cantidad.

```

CODIGO

CREATE OR REPLACE TRIGGER PROD_ENT
BEFORE INSERT ON SCA_PRO
FOR EACH ROW
BEGIN
    DBMS_OUTPUT.PUT_LINE ( Hay una nueva solicitud
    de medicamneto vital no disponible );
END;

```

```

Trigger PROD_ENT compiled

INSERT INTO SCA_PRO(id_solicitud,sca_enfer_cie,cantidad_solicitada,sca_medi_ium)
VALUES(11220,'Q774',1035,'1V1041501000100CAJAS X 10 JERINGAS PRELLENADAS X 0.7 mL');

Hay una nueva solicitud de medicamento vital no disponible

1 row inserted.

```

Figure 16: Trigger Solicitud.

4.8 Código SQL + Resultados: Funciones

En esta sección, se explorarán las funciones implementadas dentro del desarrollo del presente informe.

CODIGO

```
CREATE TABLE resultado_registros_por_ano (
    ano NUMBER,
    total_registros NUMBER
);

CREATE OR REPLACE FUNCTION obtener_registros_por_ano
(ano_in IN NUMBER)
RETURN NUMBER
IS
    total_registros NUMBER;
BEGIN
    SELECT COUNT(*) INTO total_registros
    FROM sca_pro
    WHERE EXTRACT(YEAR FROM fecha) = ano_in;

    INSERT INTO resultado_registros_por_ano
    (ano, total_registros)
    VALUES (ano_in, total_registros);

    RETURN total_registros;
END obtener_registros_por_ano;

DECLARE
    total_registros NUMBER;
    ano_a_consultar NUMBER := 2022; — Reemplaza con
    el a o a consultar
BEGIN
    total_registros := obtener_registros_por_ano
    (ano_a_consultar);
    DBMS_OUTPUT.PUT_LINE('El n mero de registros para
    el a o ' || ano_a_consultar || ' es: '
    || total_registros);
END;
/

SELECT * FROM resultado_registros_por_ano;
```

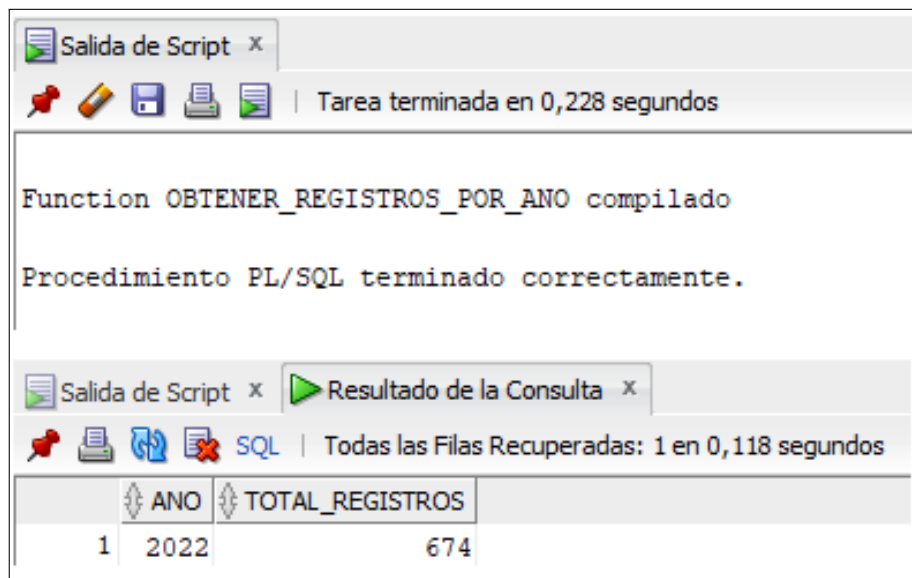


Figure 17: Funcion - obtener registros por año.

CODIGO

```
CREATE TABLE resultado_contar_por_tipo_IMPOR (
    tipo_solicitante VARCHAR2(1200),
    total_registros NUMBER
);

CREATE OR REPLACE FUNCTION contar_registros_por_tipo_IMPOR
(tipo_in IN VARCHAR2)
RETURN NUMBER
IS
    contador NUMBER;
BEGIN
    SELECT COUNT(*) INTO contador
    FROM sca_pro
    WHERE SOLICITANTE_IMPOR = tipo_in;

    INSERT INTO resultado_contar_por_tipo_IMPOR
    (tipo_solicitante , total_registros)
    VALUES (tipo_in , contador);
```

```

CODIGO
    RETURN contador;
END contar_registros_por_tipo_IMPOR;

DECLARE
    contador_resultado NUMBER;
BEGIN
    contador_resultado := contar_registros_por_tipo_IMPOR
    ('IMPROVIT MEDICAL SAS');
    DBMS_OUTPUT.PUT_LINE('El n mero de registros para
    el solicitante "IMPROVIT MEDICAL SAS" es: '
    || contador_resultado);
END;
/

SELECT * FROM resultado_contar_por_tipo_IMPOR;

```

The screenshot shows the SQL Developer interface. The top toolbar includes icons for saving, printing, and running scripts. The main window displays the following messages:

```

Table RESULTADO_CONTAR_POR_TIPO_IMPOR creado.

Function CONTAR_REGISTROS_POR_TIPO_IMPOR compilado

Procedimiento PL/SQL terminado correctamente.

```

Below the messages, the toolbar shows the 'Resultado de la Consulta' (Query Result) tab. The status bar indicates 'Todas las Filas Recuperadas: 1 en 0,105 segundos' (All rows recovered: 1 in 0.105 seconds).

	TIPO_SOLICITANTE	TOTAL_REGISTROS
1	IMPROVIT MEDICAL SAS	43

Figure 18: Funcion - Contar por tipo de Importador.

4.9 Código SQL + Resultados: procedimientos almacenados

En esta sección, se explorarán los procedimientos almacenados en la base de datos. Los procedimientos almacenados son elementos cruciales que permiten encapsular y ejecutar de manera eficiente secuencias de instrucciones en la base de datos.

```
CODIGO

— Inserta enfermedades y codigo CIE a la
tabla Enfermedades

CREATE OR REPLACE PROCEDURE InsertarEnfermedad(
    ENFERMEDADES IN VARCHAR2,
    CIE VARCHAR2)
IS
BEGIN
    INSERT INTO sca_enfer (ENFERMEDADES, CIE)
    VALUES (ENFERMEDADES, CIE);
END InsertarEnfermedad;
/

BEGIN
    InsertarEnfermedad('COLERA DEBIDO A VIBRIO CHOLERA E
01, BIOTIPO EL TOR', 'A001');
END;
/
```

```
Trigger PRO_FECHA compiled

INSERT INTO SCA_PRO(id_solicitud, sca_enfer_cie,cantidad_solicitada,sca_mediu_m)
VALUES(10000,'Q774',44,'IV1041501000100CAJAS X 10 JERINGAS PRELENADAS X 0.7 mL');

ID_SOLICITUD  FECHA      TIPO DE SOLICITUD  SOLICITANTE INFOR  CANTIDAD SOLICITADA  SCA ENFER CIE  SCA MEDI IUM
10000 05-OCT-23 (null)      (null)              44 Q774          IV1041501000100CAJAS X 10 JERINGAS PRELENADAS X 0.7 mL
```

Figure 19: Procedimiento - Inserta enfermedades y codigo CIE a la tabla Enfermedades.

```

CODIGO

— Eliminacion enfermedades y codigo CIE a la
tabla Enfermedades

CREATE OR REPLACE PROCEDURE EliminarEnfermedadPorCIE(
    p_codigo_cie10 IN VARCHAR2
)
IS
BEGIN
    DELETE FROM sca_enfer WHERE CIE = p_codigo_cie10;
    COMMIT;
END EliminarEnfermedadPorCIE;
/

BEGIN
    EliminarEnfermedadPorCIE( 'A001' );
END;
/

SELECT * FROM SCA_ENFER WHERE CIE = 'A001';
/

```

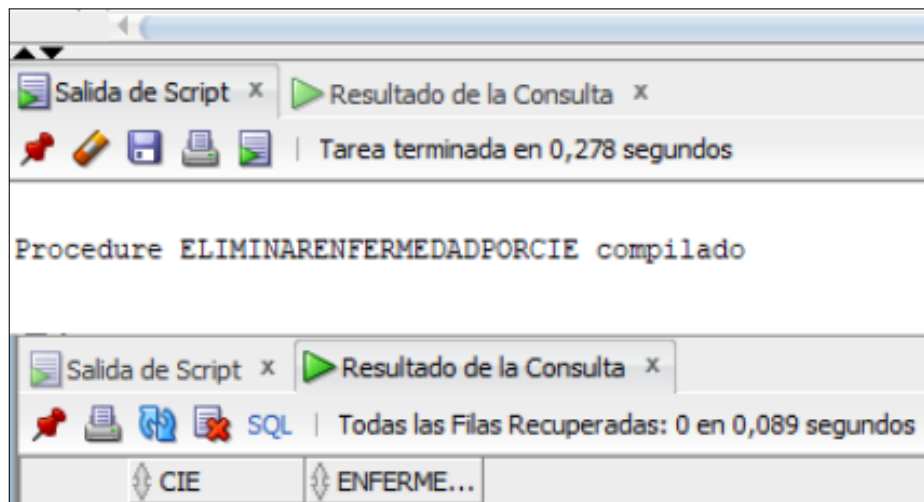


Figure 20: Procedimiento - Eliminacion enfermedades y codigo CIE a la tabla Enfermedades.

```

CODIGO

— Crear un medicamento en la tabla SCA
CREATE OR REPLACE PROCEDURE AgregarMedicamento
(
    p_IUM VARCHAR2,
    p_NOMBRE_COMERCIAL VARCHAR2,
    p_UNIDAD_MEDIDA VARCHAR2,
    p_CONCEN_DELMEDIC VARCHAR2,
    p_FORMA_FARMACEUTICA VARCHAR2,
    p_PRINCIPIO_ACTIV01 VARCHAR2,
    p_PRINCIPIO_ACTIV02 VARCHAR2,
    p_PRESENTACION_COMERCIAL VARCHAR2
)
AS
BEGIN
    — Insertar el nuevo medicamento en la tabla SCA_MEDI
    INSERT INTO SCA_MEDI (IUM, NOMBRE_COMERCIAL,
    UNIDAD_MEDIDA, CONCEN_DELMEDIC, FORMA_FARMACEUTICA,
    PRINCIPIO_ACTIV01, PRINCIPIO_ACTIV02,
    PRESENTACION_COMERCIAL)
    VALUES (p_IUM, p_NOMBRE_COMERCIAL, p_UNIDAD_MEDIDA,
    p_CONCEN_DELMEDIC, p_FORMA_FARMACEUTICA,
    p_PRINCIPIO_ACTIV01, p_PRINCIPIO_ACTIV02,
    p_PRESENTACION_COMERCIAL);
END;
/

```

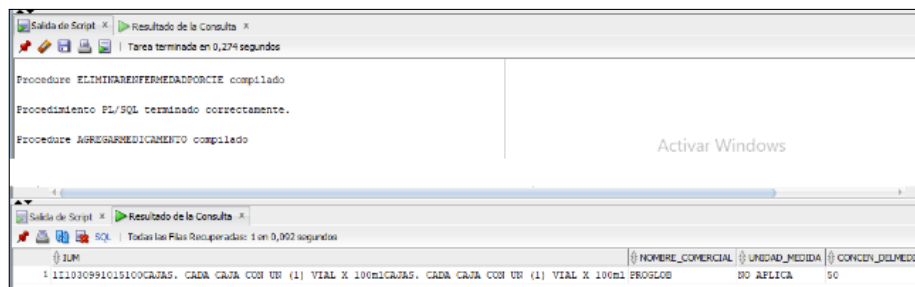


Figure 21: Crear un medicamento en la tabla SCA.

CODIGO

— Crear un procedimiento almacenado para eliminar un medicamento por su IUM

```
CREATE OR REPLACE PROCEDURE EliminarMedicamento(  
    p_IUM VARCHAR2  
)  
AS  
BEGIN  
    DELETE FROM SCA_MEDI WHERE IUM = p_IUM;  
    COMMIT;  
END EliminarMedicamento;  
/
```



Figure 22: Procedimiento - Crear un procedimiento almacenado para eliminar un medicamento por su IUM.

5 Bases de Datos No-SQL

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

Modelo Conceptual

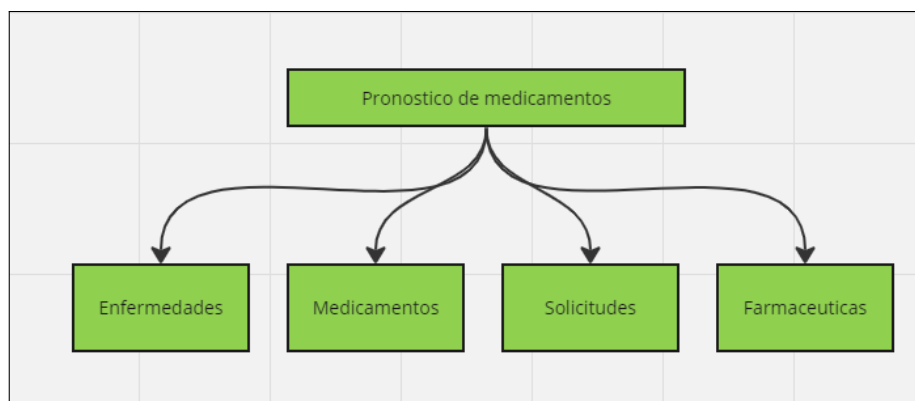


Figure 23: Modelo conceptual No - SQL.

Modelo Lógico

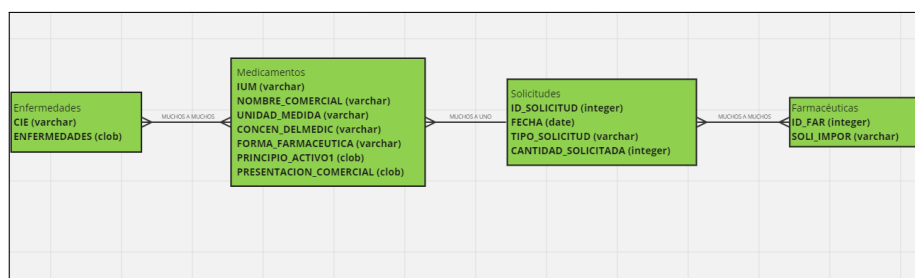


Figure 24: Modelo lógico No - SQL.

Modelo Físico

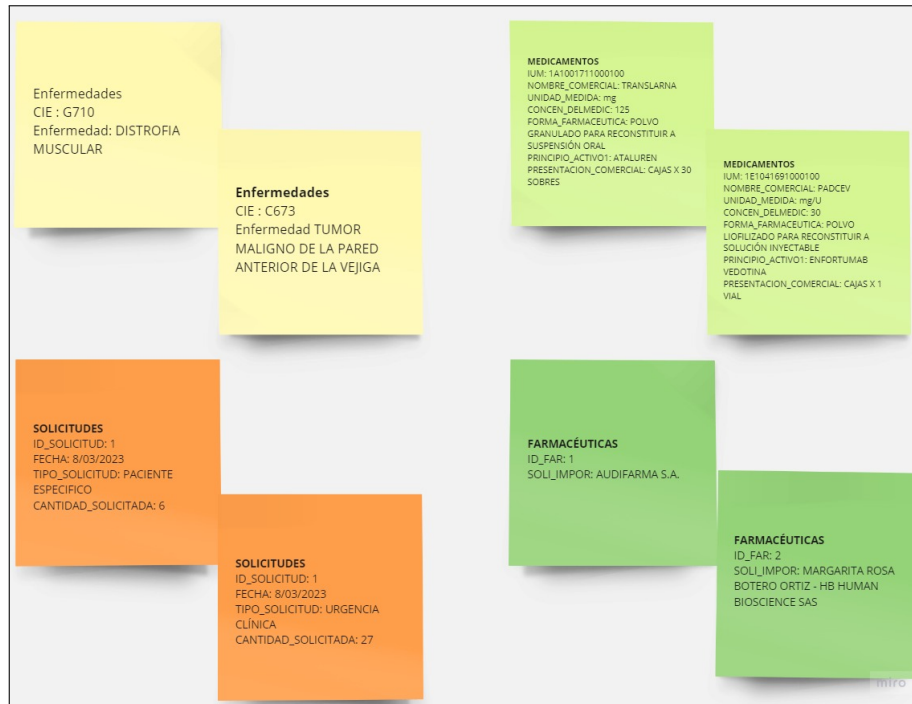


Figure 25: Modelo físico No - SQL.

5.2 SMBD utilizado para la Base de Datos No-SQL

La elección de la base de datos para la implementación No-SQL es la misma base utilizada a lo largo del proyecto. De este modo, se realizó la carga a MongoDB por medio de un cuaderno de Google Colab. A continuación, exponemos el código y la prueba de que la base quedó cargada:

CODIGO

```
pip install pymongo

from pymongo.mongo_client import MongoClient
from pymongo.server_api import ServerApi

uri = "mongodb+srv://avictoriag:1234@cluster0.17
uvqk6.mongodb.net/?retryWrites=true&w=majority"
```

```

# Create a new client and connect to the server
client = MongoClient(uri, server_api=ServerApi('1'))

# Send a ping to confirm a successful connection
try:
    client.admin.command('ping')
    print("Pinged your deployment. You successfully
    connected to MongoDB!")
except Exception as e:
    print(e)

# Seleccionar una base de datos espec ifica
db = client["PROYECTO BASES"]

# Ahora puedes usar 'db' para referirte a tu base de datos
coleccion = db["MEDICAMENTOS"]

# Intentando determinar el delimitador correcto del
archivo CSV
import csv
import pandas as pd

# Ruta al archivo CSV
ruta_archivo_csv = '/content/datos.csv'

# Cargar el archivo CSV con el delimitador detectado
df = pd.read_csv(ruta_archivo_csv, delimiter=';')

# Mostrar las primeras filas del DataFrame
df.head()

# Leer las primeras l neas del archivo para determinar
el delimitador
with open(ruta_archivo_csv, 'r', encoding='utf-8')
as archivo:
    muestra = archivo.read(1024)

delimitador = csv.Sniffer().sniff(muestra).delimiter

# Cargar el archivo CSV con el
delimitador detectado
df = pd.read_csv(ruta_archivo_csv,
delimiter=delimitador)
# Mostrar las primeras filas del DataFrame
df.head()

```

```
# Convertir el DataFrame de pandas a una
lista de diccionarios
datos_para_insertar = df.to_dict(orient='records')

# Ahora puedes usar 'insert_many' con esta
lista de diccionarios
coleccion.insert_many(datos_para_insertar)
```

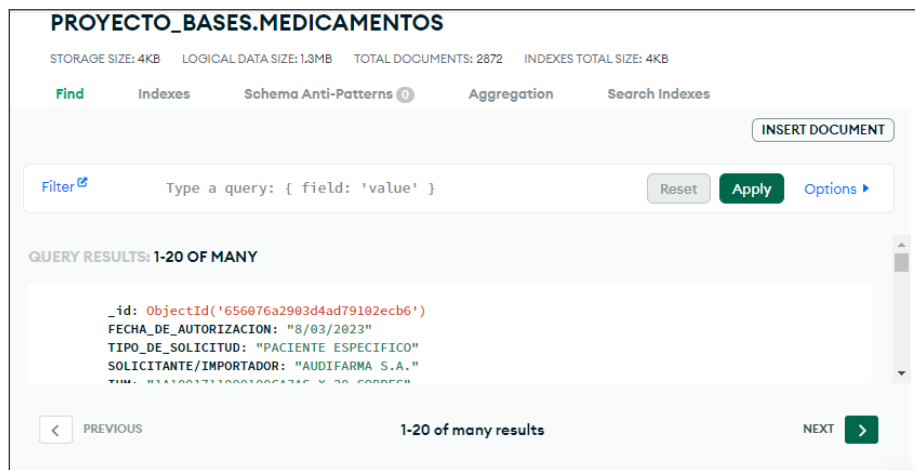


Figure 26: Carga base de datos No - SQL a MONGO.

6 Aplicación de ETL (Extract, Transform, Load) y Bodega de Datos

6.1 Extracción

La extracción de los datos no fue realizada a través de algoritmos y por lo tanto no fue requerido programar no obstante, a continuación se describe el proceso llevado a cabo para obtener los datos.

1. Ingresar a la página de datos abiertos (<https://www.datos.gov.co/>)



Figure 27: Platafoma datos abiertos

2. Buscar "Medicamentos vitales no disponibles"



Figure 28: Búsqueda

3. Seleccionar el resultado de la búsqueda

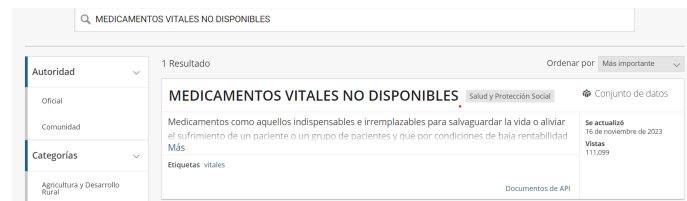


Figure 29: Resultado búsqueda

4. Seleccionar descargar

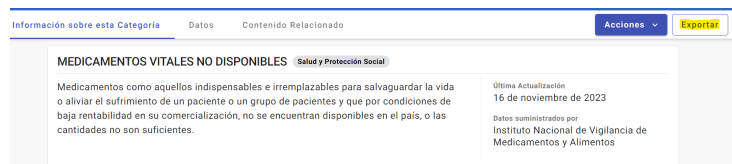


Figure 30: Base de datos en datos abiertos

5. Seleccionar CSV para Excel, Click en descargar



Figure 31: Seleccionar formato de datos

6. Resultado

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
FECHA_DE_AUTORIZACIÓN	TIPO_DE_SOLICITANTE	PRINCIPAL	CONCENT	UNIDAD	F	PRINCIPAL	CONCENT	UNIDAD	F	FORMA	F	NOMBRE	CANTIDAD	PRESENTA	DIAGNOS	CÓDIGO	DIAGNOSTICO_CIE-10
11-09-23 5:00	PACIENTE	VALENTE	10100363	GOLOSINE	50 mg/ml	NO APLIC	NO APLIC	NO APLIC	SOLUCIÓN	VYONDY	120 CAJAS X 1	DISTROFIA	6710				
11-09-23 5:00	PACIENTE	AUDIFAR	1A10037L	ATALUREN	125 mg	NO APLIC	NO APLIC	NO APLIC	POLVO GR	TRANSALF	6 CAJAS X 3	DISTROFIA	6710				
11-09-23 5:00	PACIENTE	AUDIFAR	1A10038A	ATALUREN	250 mg	NO APLIC	NO APLIC	NO APLIC	POLVO GR	TRANSALF	24 CAJAS X 3	DISTROFIA	6710				
11-09-23 5:00	PACIENTE	VALENTE	10100355	LETERJURE	50 mg/ml	NO APLIC	NO APLIC	NO APLIC	SOLUCIÓN	CHONDY	24 CAJAS X 1	DISTROFIA	6710				
11-09-23 5:00	PACIENTE	LABORATI	110103001	IOBENOP	150 mg	NO APLIC	NO APLIC	NO APLIC	TABLETA	RAXONE	6 CAJAS X 3	ATROFIA	1472				
11-08-23 5:00	PACIENTE	AUDIFAR	11034602	INOTERSE	389.3333 mg/ml	NO APLIC	NO APLIC	NO APLIC	SOLUCIÓN	TEGSEDI	6 CAJAS X 4	AMLIODICE	6852				
11-08-23 5:00	MÁS DE UI	BIOSEPIFA	1F10020461	FIBRINOGL	1 g	NO APLIC	NO APLIC	NO APLIC	POLVO UIC	FIBRYGSA	300 DISPOSITO	DEFICIEN	0684				
11-08-23 5:00	MÁS DE UI	BIOSEPIFA	1F10020461	FIBRINOGL	1 g	NO APLIC	NO APLIC	NO APLIC	POLVO UIC	FIBRYGSA	300 FILTRO DE	DEFICIEN	0684				
11-08-23 5:00	MÁS DE UI	BIOSEPIFA	1F10020461	FIBRINOGL	1 g	NO APLIC	NO APLIC	NO APLIC	POLVO UIC	FIBRYGSA	300 VIALES DE	DEFICIEN	0684				
11-08-23 5:00	MÁS DE UI	BIOSEPIFA	1F10020461	FIBRINOGL	1 g	NO APLIC	NO APLIC	NO APLIC	POLVO UIC	FIBRYGSA	300 CAJAS	DEFICIEN	0684				
11-08-23 5:00	MÁS DE UI	LABORATI	1H103034	HEPATITIS	20 mg/ml (1	NO APLIC	NO APLIC	NO APLIC	SUSPENS	SILCOV	1,500 CAJAS	CA PORTADO	2225				
11-07-23 5:00	URGENCIA	VALENTE	2E1012451	ELEXACAF	100/50/75 mg	IVACAFITC	150 mg	TABLETA	TRIKAFTA	1 CAJAS X 3	B FIBROSIS	1E849					
11-07-23 5:00	URGENCIA	VALENTE	2E1012451	ELEXACAF	100/50/75 mg	IVACAFITC	150 mg	TABLETA	TRIKAFTA	1 CAJAS X 3	B FIBROSIS	1E849					
11-07-23 5:00	PACIENTE	SALUTIS	5,2E1040753	ELEXACAF	50/25/72,7 mg	IVACAFITC	75 mg	TABLETA	TRIKAFTA	6 CAJAS X 3	B FIBROSIS	1E849					
11-07-23 5:00	URGENCIA	UNO	HEAL	1M100644	CISTEAMINE	75 mg	NO APLIC	NO APLIC	NO APLIC	CAPISULA	PROCTYBRI	8 FRASCOS	INSUFICIE	N189			
11-07-23 5:00	MÁS DE UI	GLOBAL	51E1010055	LETOGILIN	250 mg	NO APLIC	NO APLIC	NO APLIC	CAPISULA	PETINAME	250 CAJAS X 3	CONVULS	NO REPORTADO				
11-07-23 5:00	URGENCIA	SANOFI	AZ	1V100140	VANDETA	100 mg	NO APLIC	NO APLIC	NO APLIC	TABLETA	CAPRELSA	18 CAJAS X 3	TUMOR M	C73X			
11-07-23 5:00	PACIENTE	GLOBAL	SI	11022923	LEVOTIRO	0,05 mg/ml	NO APLIC	NO APLIC	NO APLIC	POLVO UIC	NO REPO	24 CAJAS X 1	TUMOR M	C73X			
11-07-23 5:00	MÁS DE UI	HOLLAND	11020993	INNAJUNOC	50 mg/ml	NO APLIC	NO APLIC	NO APLIC	SOLUCIÓN	PROSELO	600 CAJA X 1	DEFECTOS	0693				

Figure 32: Base de datos extraída

6.2 Transformación

1. Se carga la base en Google Colab

	FECHA_DE_AUTORIZACIÓN	TIPO_DE_SOLICITUD	SOLICITANTE/EXPORTADOR	TIPO	PRINCIPIO_ACTIV1	CONCENTRACIÓN_DEL MEDICAMENTO1	UNIDAD_MEDIDA1	PRINCIPIO_ACTIV2	CONCENTRACIÓN_DEL MEDICAMENTO2	UNIDAD_MEDIDA2
0	11/09/2023 05:00:00 AM	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	1G102616100100	GOLOORSIN	50	mg/ml	NO APLICA		
1	11/09/2023 05:00:00 AM	PACIENTE ESPECIFICO	AUDIFARMA S.A.	1A100171000100	ATALUREN	125	mg	NO APLICA		
2	11/09/2023 05:00:00 AM	PACIENTE ESPECIFICO	AUDIFARMA S.A.	1A1000141000100	ATALUREN	250	mg	NO APLICA		
3	11/09/2023 05:00:00 AM	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	1E102551000101	ETEPURSEN	50	mg/ml	NO APLICA		
4	11/09/2023 05:00:00 AM	PACIENTE ESPECIFICO	LABORATORIOS BIOPAS S.A.	1I1035001000101	IDEBENONA	150	mg	NO APLICA		
...
6036	01/11/2018 05:00:00 AM	MÁS DE UN PACIENTE	MEDICAL KIT LTDA	1S101161000100	FABOTERAPICO POLINALENTE ANTVALACRAN	1.8	mg	NO APLICA		
6036	01/09/2018 05:00:00 AM	URGENCIA CLINICA	HE HUMAN BIOSCIENCE S.A.S	1D1000031000100	DIALNORUBICINA LIPOSOMAL			NO APLICA		

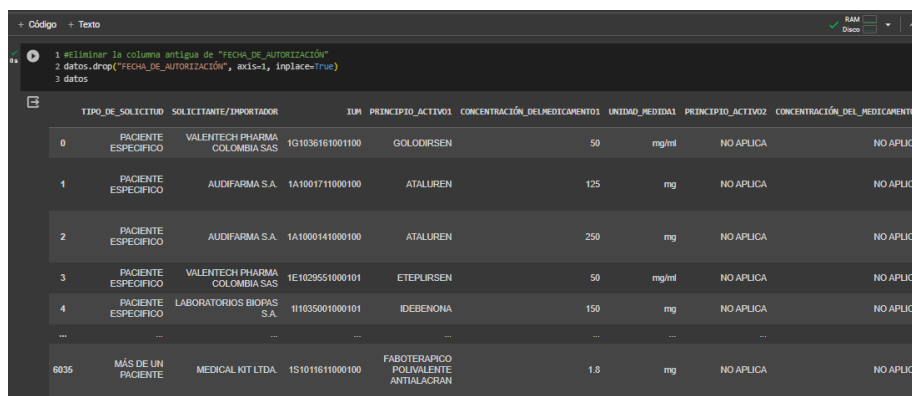
Figure 33: Base

2. Eliminación de valores NULL, validación de inexistencias de valores nulos. Adicional se ajusto el campo "FECHA AUTORIZACIÓN", convirtiendolo a formato date dd/mm/yyyy.

	FECHA_DE_AUTORIZACIÓN	FECHA_DE_AUTORIZACION
0	2023-11-09 05:00:00	09/11/2023
1	2023-11-09 05:00:00	09/11/2023
2	2023-11-09 05:00:00	09/11/2023
3	2023-11-09 05:00:00	09/11/2023
4	2023-11-09 05:00:00	09/11/2023
...
6035	2018-01-11 05:00:00	11/01/2018

Figure 34: Transformación

3. Eliminación del campo antiguo de "FECHA ACTUALIZACIÓN" con formato fecha/hora.



```

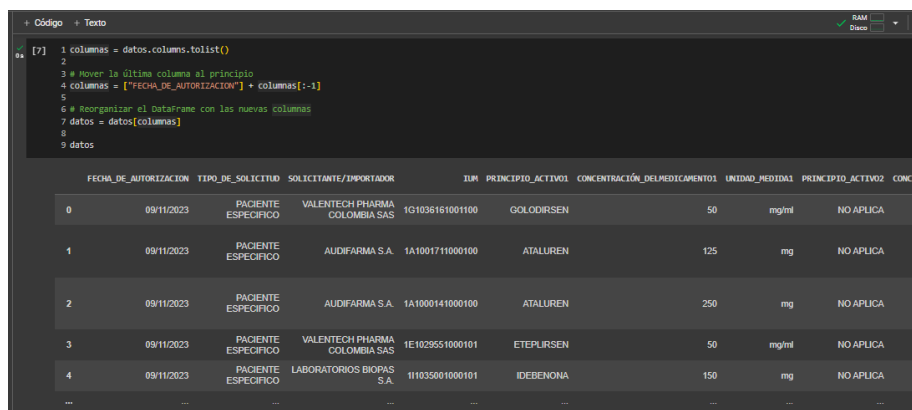
1 #eliminar la columna antigua de "FECHA_ACTUALIZACIÓN"
2 datos.drop("FECHA_ACTUALIZACIÓN", axis=1, inplace=True)
3 datos

```

	TIPO_DE_SOLICITUD	SOLICITANTE/IMPORTADOR	ILM	PRINCIPIO_ACTIV01	CONCENTRACIÓN_DELMEDICAMENTO1	UNIDAD_MEDIDA1	PRINCIPIO_ACTIV02	CONCENTRACIÓN_DEL_MEDICAMENTO2
0	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	1G1036161001100	GOLODIRSEN	50	mg/ml	NO APLICA	NO APLICA
1	PACIENTE ESPECIFICO	AUDIFARMA S.A.	1A1001711000100	ATALUREN	125	mg	NO APLICA	NO APLICA
2	PACIENTE ESPECIFICO	AUDIFARMA S.A.	1A1000141000100	ATALUREN	250	mg	NO APLICA	NO APLICA
3	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	1E1029551000101	ETEPURSEN	50	mg/ml	NO APLICA	NO APLICA
4	PACIENTE ESPECIFICO	LABORATORIOS BIOPAS S.A.	1I1035001000101	IDEBENONA	150	mg	NO APLICA	NO APLICA
...
6035	MÁS DE UN PACIENTE	MEDICAL KIT LTDA.	1S1011611000100	FABOTERAPICO POLIVALENTE ANTALACRAN	1.8	mg	NO APLICA	NO APLICA

Figure 35: Transformación

4. Reorganización de las columnas, con el fin de asignar como primera el campo de fecha que se transformo anteriormente.



```

[7] 1 columnas = datos.columns.tolist()
2
3 # Mover la última columna al principio
4 columnas = ["FECHA_DE_ACTUALIZACIÓN"] + columnas[1:]
5
6 # Reorganizar el Dataframe con las nuevas columnas
7 datos = datos[columnas]
8
9 datos

```

	FECHA_DE_ACTUALIZACIÓN	TIPO_DE_SOLICITUD	SOLICITANTE/IMPORTADOR	ILM	PRINCIPIO_ACTIV01	CONCENTRACIÓN_DELMEDICAMENTO1	UNIDAD_MEDIDA1	PRINCIPIO_ACTIV02	CONCENTRACIÓN_DEL_MEDICAMENTO2
0	09/11/2023	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	1G1036161001100	GOLODIRSEN	50	mg/ml	NO APLICA	NO APLICA
1	09/11/2023	PACIENTE ESPECIFICO	AUDIFARMA S.A.	1A1001711000100	ATALUREN	125	mg	NO APLICA	NO APLICA
2	09/11/2023	PACIENTE ESPECIFICO	AUDIFARMA S.A.	1A1000141000100	ATALUREN	250	mg	NO APLICA	NO APLICA
3	09/11/2023	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	1E1029551000101	ETEPURSEN	50	mg/ml	NO APLICA	NO APLICA
4	09/11/2023	PACIENTE ESPECIFICO	LABORATORIOS BIOPAS S.A.	1I1035001000101	IDEBENONA	150	mg	NO APLICA	NO APLICA
...

Figure 36: Transformación

5. Reemplazo de un caracter especial en los encabezados y validación de data lista para usar.

```

1 # Reemplazar '0' por '00' solo en columnas de tipo cadena
2 datos.columns = datos.columns.str.replace('0', '00')
3 datos

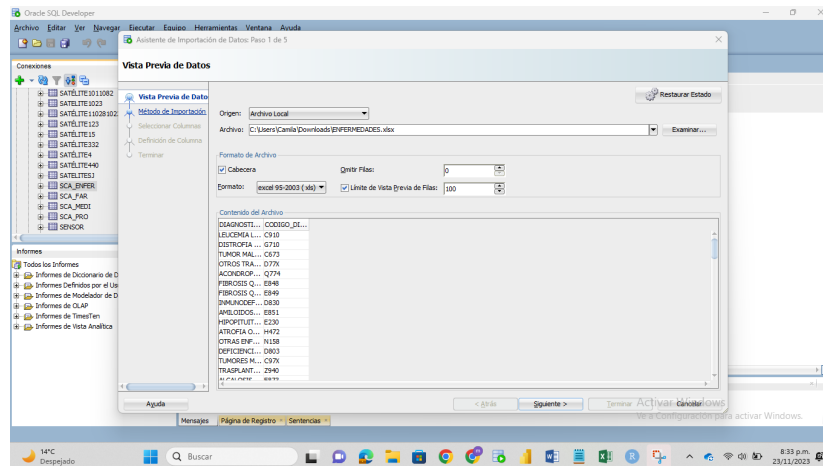
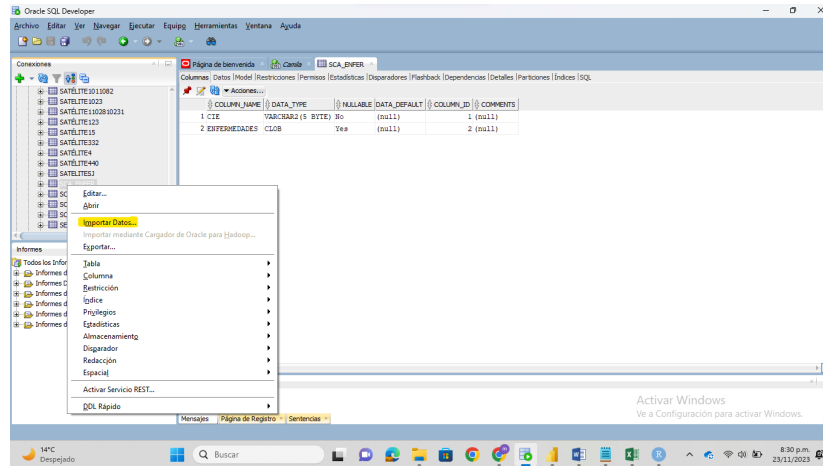
```

	FECHA_DE_AUTORIZACION	TIPO_DE_SOLICITUD	SOLICITANTE/IMPORTADOR	IUM	PRINCIPIO_ACTIV01	CONCENTRACION_DELMEDICAMENTO1	UNIDAD_MEDIDA1	PRINCIPIO_ACTIV02	CONCENTRACION_DELMEDICAMENTO2
0	09/11/2023	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	1G1038161001100	GOLODIRSEN	50	mg/ml	NO APLICA	
1	09/11/2023	PACIENTE ESPECIFICO	AUDIFARMA S.A.	1A1001711000100	ATALUREN	125	mg	NO APLICA	
2	09/11/2023	PACIENTE ESPECIFICO	AUDIFARMA S.A.	1A1000141000100	ATALUREN	250	mg	NO APLICA	
3	09/11/2023	PACIENTE ESPECIFICO	VALENTECH PHARMA COLOMBIA SAS	1E1029551000101	ETEPLURSEN	50	mg/ml	NO APLICA	
4	09/11/2023	PACIENTE ESPECIFICO	LABORATORIOS BIOPAS S.A.	1I1035001000101	IDEBENONA	150	mg	NO APLICA	
...
6035	11/01/2018	MÁS DE UN PACIENTE	MEDICAL KIT LTDA.	1S1011611000100	FABOTERAPICO POLIVALENTE ANTIALACRAN	1.8	mg	NO APLICA	
6036	09/01/2018	URGENCIA CLINICA	HB HUMAN BIOSCIENCE S.A.S.	1D1000031000100	DAUNORUBICINA LIBESOMAL	2	mg/ml	NO APLICA	

Figure 37: Transformación

6.3 Carga

1. Tabla enfermedades.



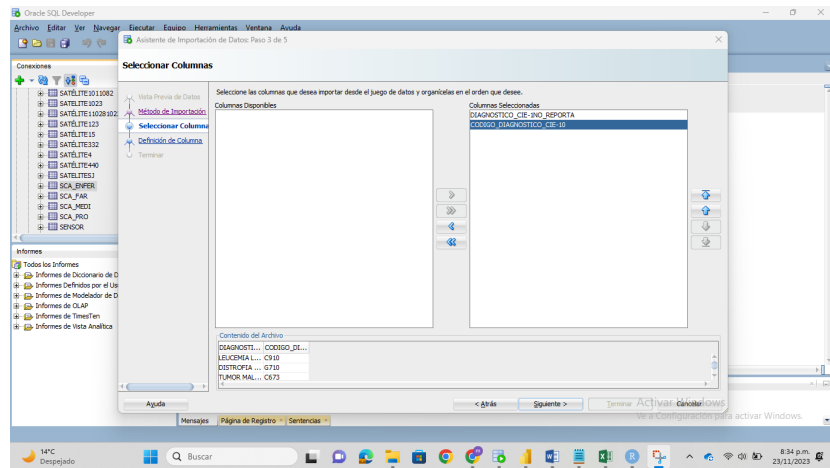


Figure 40: 3. Incluir las variables

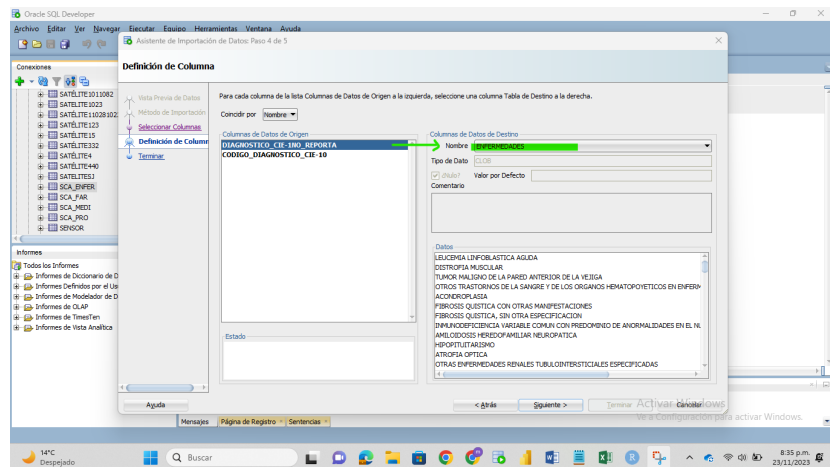
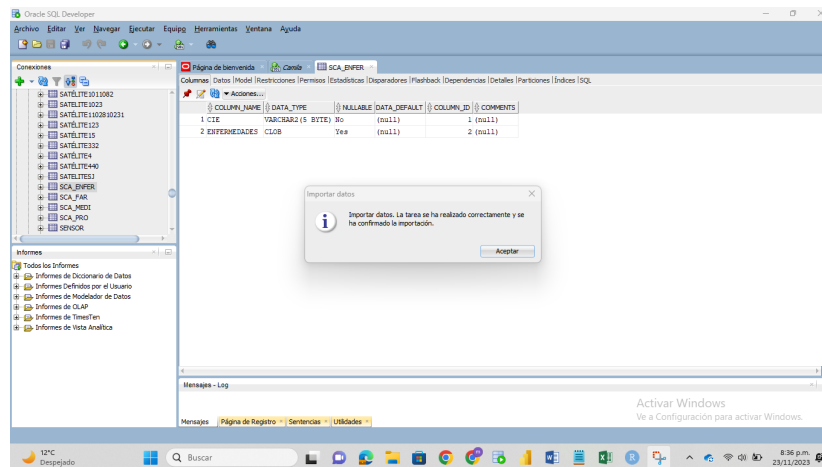
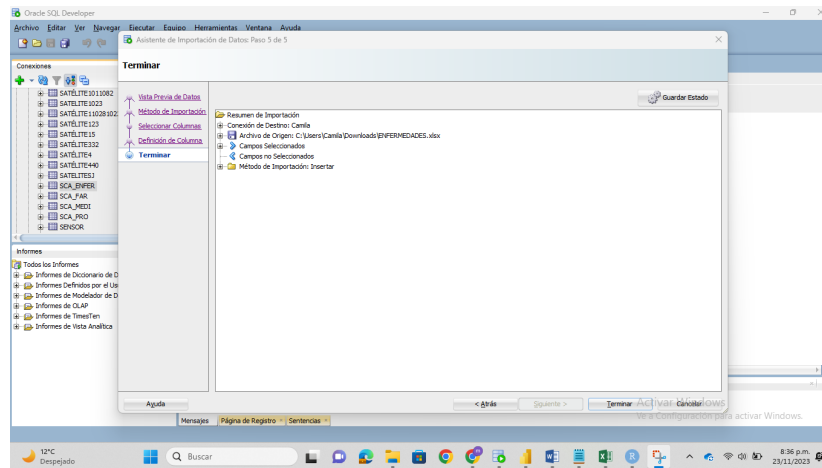


Figure 41: 4. Asignar la columna a cada variable



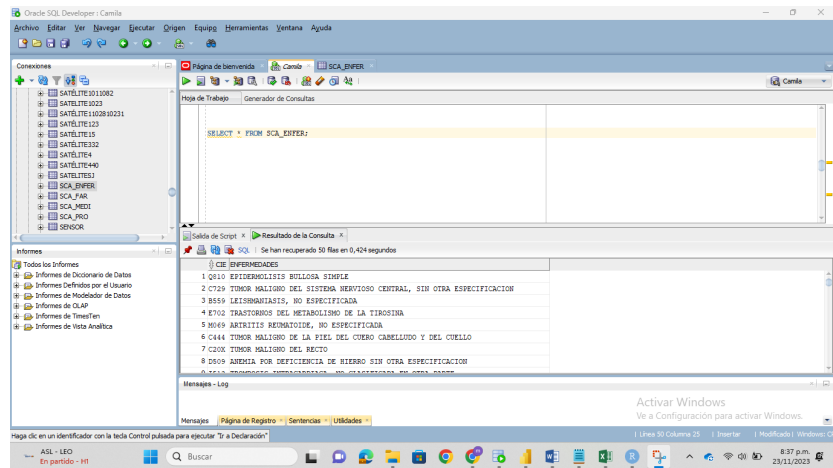


Figure 44: 7. Asegurarse de que la base se subio

2. Tabla Medicamentos.

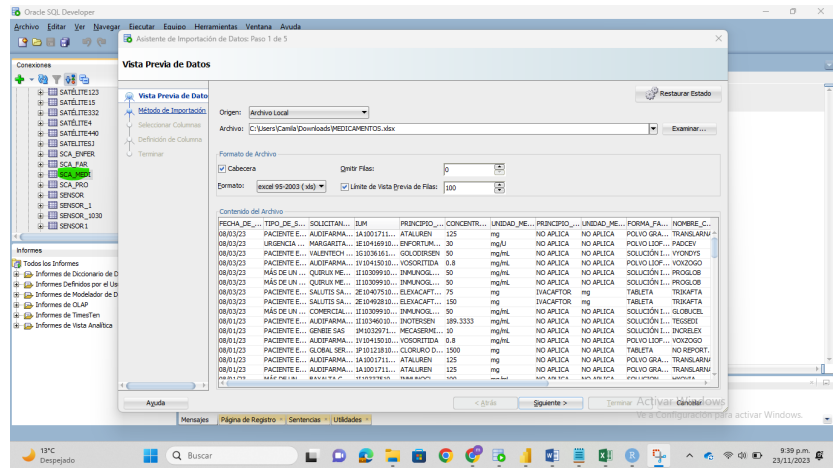


Figure 45: 1. Inclure tabla

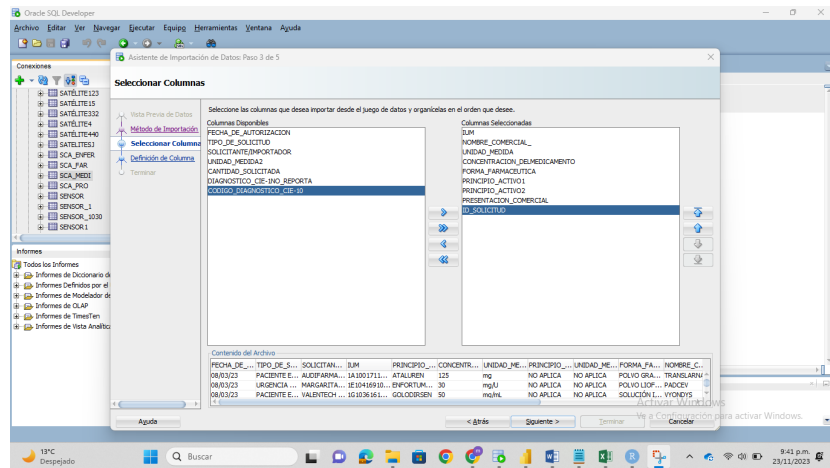


Figure 46: 2. Inclusión variables

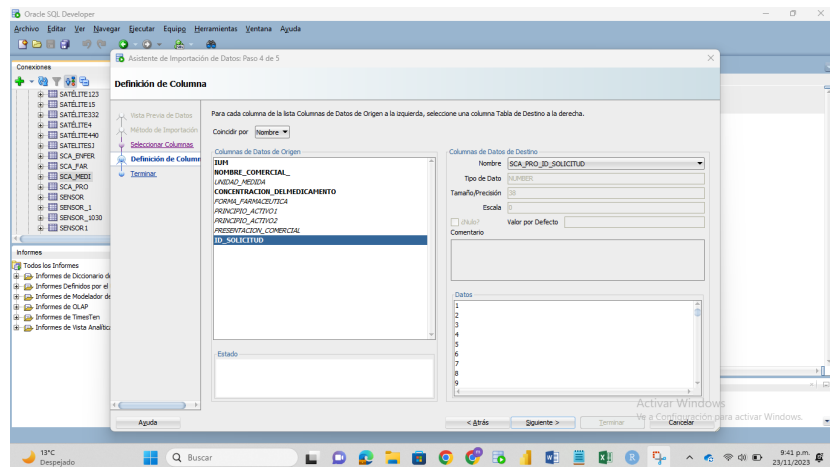


Figure 47: 3. Asigancion de variables a columnas

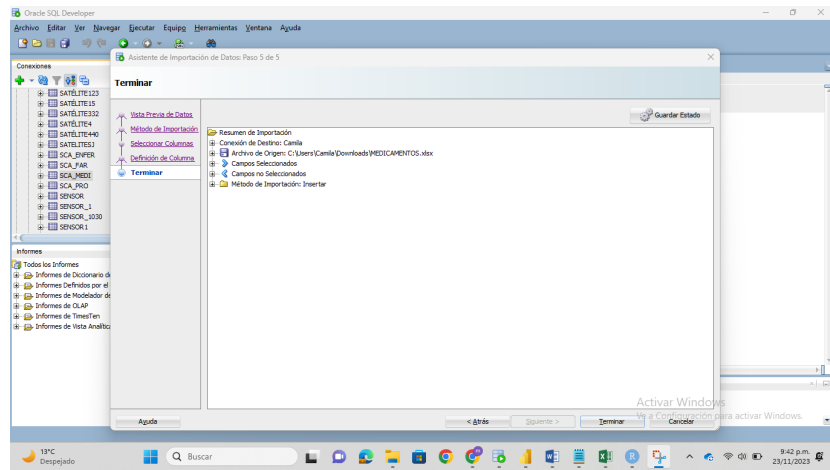


Figure 48: 4. Finalizar inclusion data

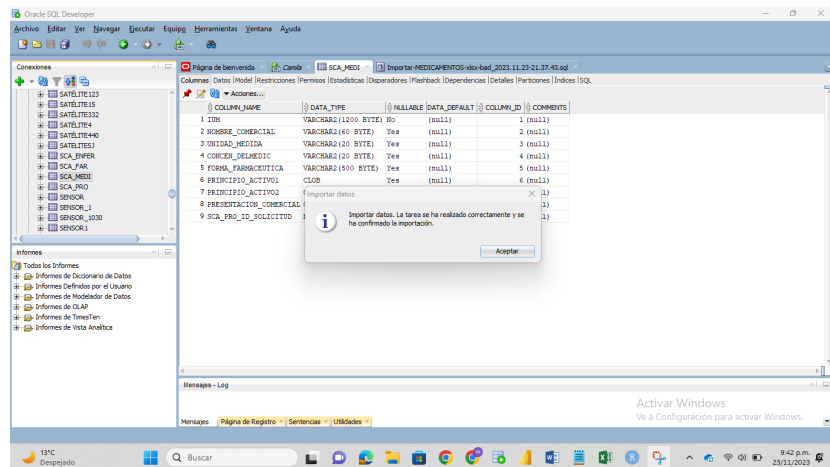


Figure 49: 5. Finalizar proceso

3. Tabla productos

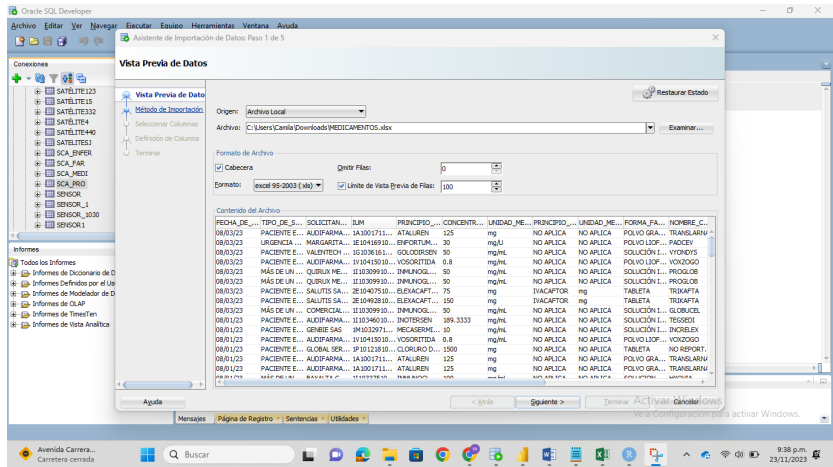


Figure 50: 1. Insertar base de productos

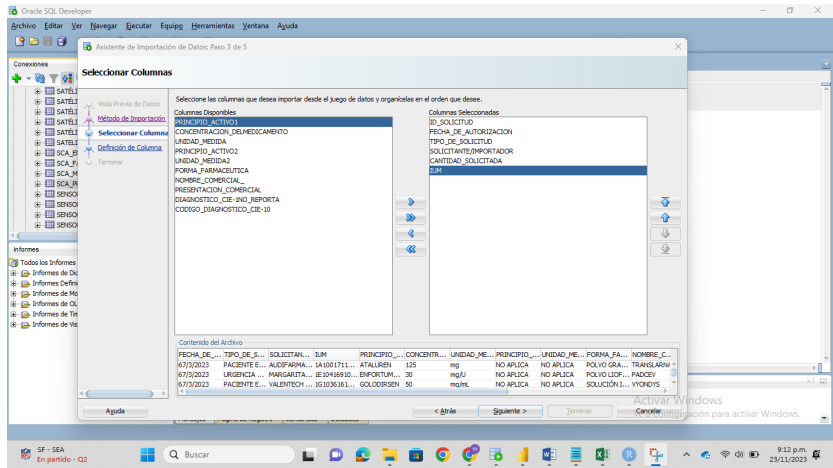


Figure 51: 2. Incluir las variables

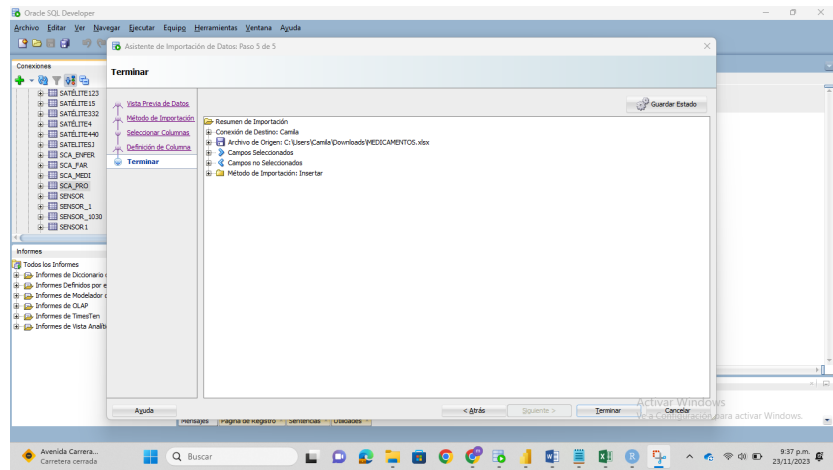


Figure 52: 3. Terminar proceso de subir bases

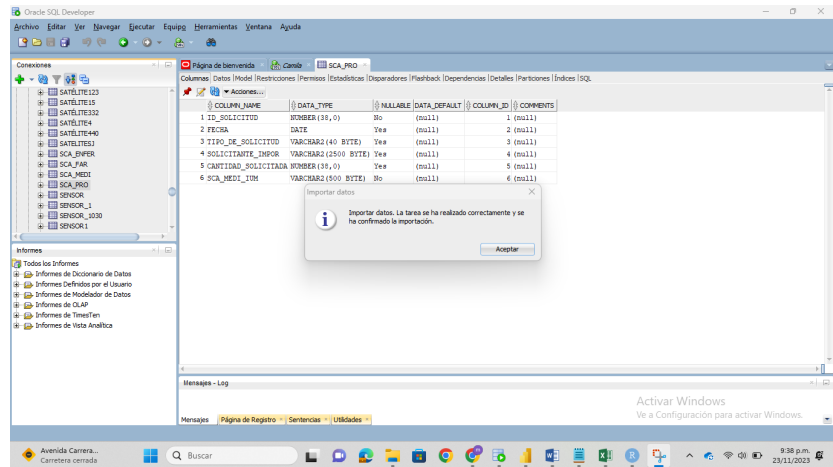


Figure 53: 4. Inclusión finalizada

7 Proximos pasos

- Se requieren una mayor cantidad de datos para mejorar los modelos y así construir modelos mas robustos y de mejor calidad para predecir el comportamiento de los medicamentos vitales no disponibles en Colombia.
- Se deben añadir variables adicionales que puedan explicar mejor el fenómeno.

8 Lecciones aprendidas

- Entender la importancia del modelado de datos para representar la realidad de manera estructurada.
- Aprender a diseñar y crear modelos de datos utilizando herramientas como el Modelo Entidad-Relación (ER).
- Comprender los principios de la normalización y su importancia para la eficiencia y la integridad de la base de datos.
- Aplicar técnicas de normalización para reducir la redundancia y mejorar la consistencia de los datos.
- Adquirir habilidades en el uso del lenguaje SQL para realizar consultas, actualizar datos y gestionar bases de datos. Entender cómo diseñar consultas eficientes y complejas para extraer información específica.
- Conocer cómo diseñar y crear bases de datos relacionales, incluyendo la definición de tablas, claves primarias y foráneas.
- Comprender la importancia de la integridad referencial.
- Aprender técnicas para optimizar consultas SQL y mejorar el rendimiento de las operaciones de la base de datos.
- Conocer la importancia de los índices y cómo utilizarlos adecuadamente.
- Comprender los principios de seguridad en bases de datos, incluyendo la gestión de usuarios, roles y privilegios.
- Familiarizarse con conceptos NoSQL y entender cuándo es apropiado utilizar bases de datos NoSQL en lugar de las relacionales.
- Aprender de los diferentes sistemas de gestión de bases de datos como Oracle, MYSQL, MongoDB, PostgreSQL, entre muchas más.
- Conocer apartir de las noticias como esta que se esta moviendo la IA en la actualidad y su importancia para el futuro.
- Aplicar conocimientos adquiridos en un proyecto práctico, lo que puede incluir diseñar y construir una base de datos desde cero, resolver problemas de rendimiento o estructura relacional.
- Experimentar con la colaboración y trabajo en equipo en proyectos de bases de datos.

9 Bibliografía

- Oracle. (2021). Oracle Data Modeler Documentation.
Recuperado de <https://www.oracle.com/database/technologies/appdev/datamodeler.html>
- Instituto Nacional de Vigilancia de Medicamentos y Alimentos. (2018).
MEDICAMENTOS VITALES NO DISPONIBLES [Data set]. Recuperado
de <https://www.datos.gov.co/Salud-y-Proteccion-Social/MEDICAMENTOS-VITALES-NO-DISPONIBLES/sdmr-tfmf>
- Medicamentos Vitales No Disponibles - Instituto Nacional de Vigilancia
de Medicamentos y Alimentos. (s/f). Gov.co. Recuperado de:
- <https://www.invima.gov.co/medicamentos-vitales-no-disponibles>