



INFORME DE GUÍA PRÁCTICA

I. PORTADA

Tema:	Examen Practico Primer Parcial
Unidad de Organización Curricular:	PROFESIONAL
Nivel y Paralelo:	5 ^{to} – ‘A’
Alumnos participantes:	Ases Tiban Jeremy Damian Palate Moreta Kevin Damian Poveda Gomez William Alberto Pullupaxi Chango Daniel
Asignatura:	Sistemas De Bases De Datos Distribuidos
Docente:	Ing. José Rubén Caiza

II. INFORME DE GUÍA PRÁCTICA

2.1 Objetivos

General:

Aplicar y demostrar los conceptos fundamentales de los Sistemas de Bases de Datos Distribuidas mediante la implementación práctica de un mini-sistema clínico, haciendo énfasis en la fragmentación horizontal,

Específicos:

- Modelar e Inicializar el Sistema: Crear el esquema de la base de datos distribuida, incluyendo las entidades CentrosMedicos, Medicos y Consultas con sus respectivas claves primarias, foráneas e índices.
- Verificar Operaciones CRUD Locales: Demostrar el correcto funcionamiento de las operaciones básicas.

2.2 Modalidad

Presencial.

2.3 Tiempo de duración

Presenciales: 3 horas.

No presenciales: N/A.

2.4 Instrucciones

- Fase 1: Creación de la Base Local

Preparar el Stack: Elige PostgreSQL o SQL Server.

Modelado (01_schema.sql): Crea las 3 tablas con todas las llaves e índices.

Carga (02_seed.sql): Inserta datos mínimos (4 Centros, 6 Médicos, 20 Consultas).

Pruebas CRUD: Demuestra que puedes insertar, modificar, borrar y realizar las dos consultas analíticas locales. (Capturas)

- Fase 2: Distribución y Transparencia

Configurar Nodos: Crea dos ambientes separados (Nodo_A y Nodo_B).

Fragmentación (03_fragmentacion.sql): Distribuye los datos de las tres tablas entre Nodo_A (Ambato/Quito) y Nodo_B (Cuenca/Guayaquil) usando CentroID. (Capturas de la división)

Capa Global (04_vista_global.sql): Crea la vista (vw_ConsultasGlobal) que combine los fragmentos de la tabla Consultas para simular una sola tabla. (Captura de la consulta a la vista)



Optimización: Ejecuta la consulta global con filtros. Explica en el informe/README cómo garantizaste que se movieran pocos datos (filtrando por nodo).

- Fase 3: Robustez y Entrega

Replicación (Opcional - 05_replicacion.sql): Si lo implementas, replica CentrosMedicos y justifica la política de actualización.

Prueba de Fallo: Simula la caída de un nodo. (Captura)

Describe qué consultas siguen funcionando y cuál sería tu plan de recuperación.

Organizar Entrega:

Junta todos los scripts SQL y las capturas en la estructura de carpeta solicitada.

Crear README.md: Incluye el diagrama de arquitectura, las instrucciones para levantar el entorno y la reflexión final.

Entregar la carpeta completa.

2.5 Listado de equipos, materiales y recursos

- Computadora.
- Docker

TAC (Tecnologías para el Aprendizaje y Conocimiento) empleados en la guía práctica:

- ☐ Plataformas educativas
- ☒ Simuladores y laboratorios virtuales
- ☐ Aplicaciones educativas
- ☐ Recursos audiovisuales
- ☐ Gamificación
- ☒ Inteligencia Artificial

Otros (Especifique): _____

2.6 Actividades desarrolladas

Preparación de entorno:

Se define el archivo de configuración docker-compose.yml para levantar múltiples servicios de bases de datos



```
kevin@Khevin: ~/Examen_BDD$ nano docker-compose.yml
kevin@Khevin:~/Examen_BDD_Grupo3$ cat docker-compose.yml
services:
  nodo_a:
    image: mcr.microsoft.com/mssql/server:2022-latest
    container_name: Nodo_A
    environment:
      ACCEPT_EULA: "Y"
      SA_PASSWORD: "12345Aa!"
    ports:
      - "1434:1433" # Puerto externo 1434 -> interno 1433
    volumes:
      - ./data/nodo_a:/var/opt/mssql
    networks:
      - red_distribuida
    restart: unless-stopped

  nodo_b:
    image: mcr.microsoft.com/mssql/server:2022-latest
    container_name: Nodo_B
    environment:
      ACCEPT_EULA: "Y"
      SA_PASSWORD: "12345Aa!"
    ports:
      - "1435:1433" # Puerto externo 1435 -> interno 1433
    volumes:
      - ./data/nodo_b:/var/opt/mssql
    networks:
      - red_distribuida
    restart: unless-stopped

  nodo_c:
    image: mcr.microsoft.com/mssql/server:2022-latest
    container_name: Nodo_C
    environment:
      ACCEPT_EULA: "Y"
      SA_PASSWORD: "12345Aa!"
    ports:
      - "1436:1433" # Puerto externo 1436 -> interno 1433
    volumes:
      - ./data/nodo_c:/var/opt/mssql
    networks:
      - red_distribuida
    restart: unless-stopped

networks:
  red_distribuida:
    driver: bridge
kevin@Khevin:~/Examen_BDD_Grupo3$ |
```

Fig 1. Archivo docker-compose.yml

Se establecen volúmenes locales de persistencia para asegurar que los datos de las bases de datos no se pierdan al detener o reiniciar los contenedores

```
kevin@Khevin:~/Examen_BDD_Grupo3$ mkdir -p data/nodo_a data/nodo_b data/nodo_c
kevin@Khevin:~/Examen_BDD_Grupo3$ ll
total 16
drwxr-xr-x 3 kevin kevin 4096 Oct 14 08:47 ./
drwxr-xr-x 7 kevin kevin 4096 Oct 14 08:45 ../
drwxr-xr-x 5 kevin kevin 4096 Oct 14 08:47 data/
-rw-r--r-- 1 kevin kevin 1102 Oct 14 08:46 docker-compose.yml
kevin@Khevin:~/Examen_BDD_Grupo3$ chmod -R 777 data
kevin@Khevin:~/Examen_BDD_Grupo3$ |
```

Fig 2. Creación de directorios locales para la persistencia de los datos de los contenedores

Se ejecuta el comando docker-compose up para construir y levantar las instancias de los nodos

```
kevin@Khevin:~/Examen_BDD_Grupo3$ docker-compose up -d
Creating network "examen_bdd_grupo3_red_distribuida" with driver "bridge"
Creating Nodo_A ... done
Creating Nodo_B ... done
Creating Nodo_C ... done
kevin@Khevin:~/Examen_BDD_Grupo3$ |
```

Fig 3. Ejecución del Docker-compose para la creación y despliegue de los contenedores.



Se verifica el estado de los servicios para confirmar que todos los nodos lógicos estén operativos y escuchando en los puertos definidos.

```
kevin@khevin:~/Examen_BDD_Grupo3$ docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS
f324b6e8f7a7   mcr.microsoft.com/mssql/server:2022-latest   "/opt/mssql/bin/laun..." 7 minutes ago   Up 7 minutes   0.0.0.0:1436->1433/tcp, [::]:1436->1433/tcp
ef81e8d40603   mcr.microsoft.com/mssql/server:2022-latest   "/opt/mssql/bin/laun..." 7 minutes ago   Up 7 minutes   0.0.0.0:1435->1433/tcp, [::]:1435->1433/tcp
149c8f92bedd   mcr.microsoft.com/mssql/server:2022-latest   "/opt/mssql/bin/laun..." 7 minutes ago   Up 7 minutes   0.0.0.0:1434->1433/tcp, [::]:1434->1433/tcp
kevin@khevin:~/Examen_BDD_Grupo3$
```

Fig 4. Verificación del estado 'running' de los contenedores Docker del entorno distribuido.

Se establece la conexión con las herramientas de administración a cada una de las instancias/nodos.

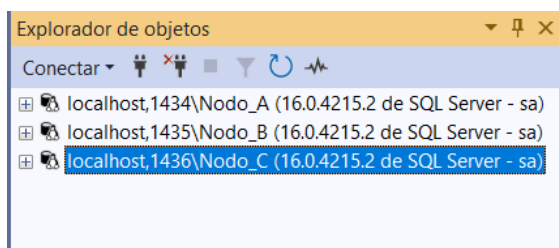


Fig 5. Conexión exitosa a las diferentes instancias de bases de datos.

A. Fundamentos / Base (30 pts)

1. Modelo y DDL (10 pts)

Se procedió a implementar el DDL para crear el modelo relacional distribuido en cada nodo. Se utilizó un esquema por ciudad dentro de cada instancia lógica para simular una mayor granularidad de fragmentación.

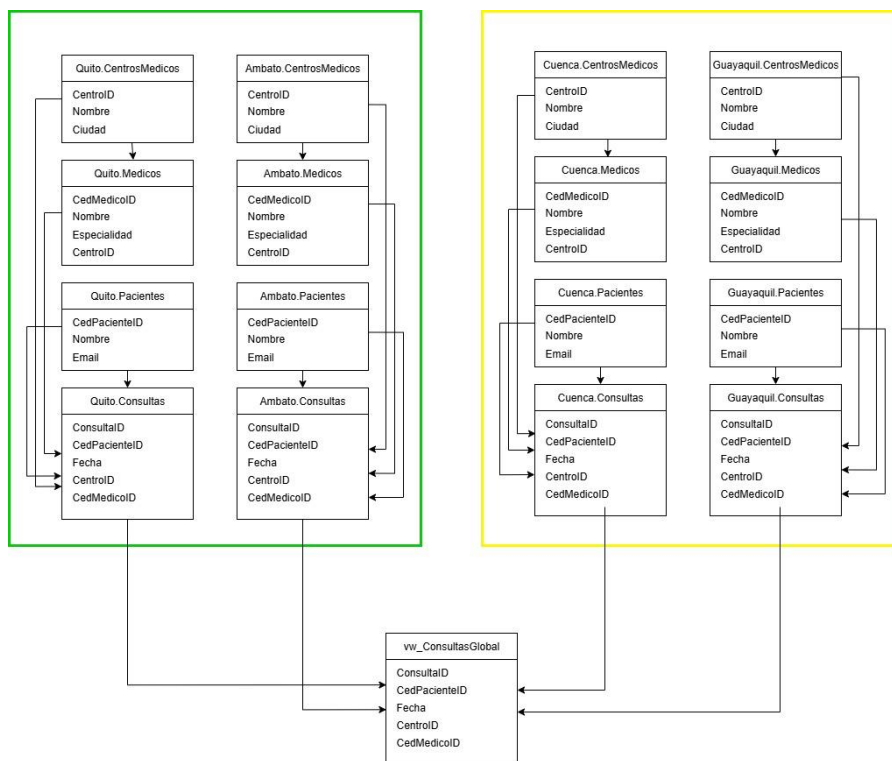


Fig 6. Modelo relacional en draw.io

NODO_A

Crear Base de Datos:

Para iniciar el desarrollo del sistema distribuido, se procedió a crear la base de datos correspondiente al **Nodo_A**, la cual gestionará la información de los centros médicos ubicados en **Ambato** y **Quito**.

Esta base de datos servirá como instancia local que contiene las tablas necesarias para almacenar médicos, centros y consultas asociadas a estas dos ciudades.

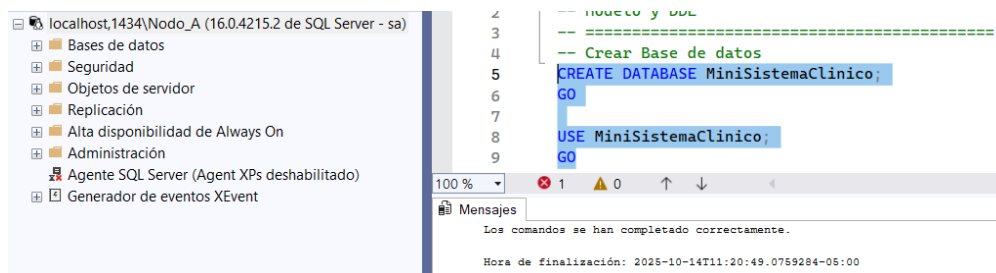


Fig 7. Script de la creación de la base de datos correspondiente al Nodo_A

Crear Esquemas:

Dentro del **Nodo_A** se definieron dos esquemas denominados **Ambato** y **Quito**, los cuales permiten organizar los datos de manera lógica y separada, facilitando la administración y el control de los fragmentos horizontales correspondientes a cada ciudad.

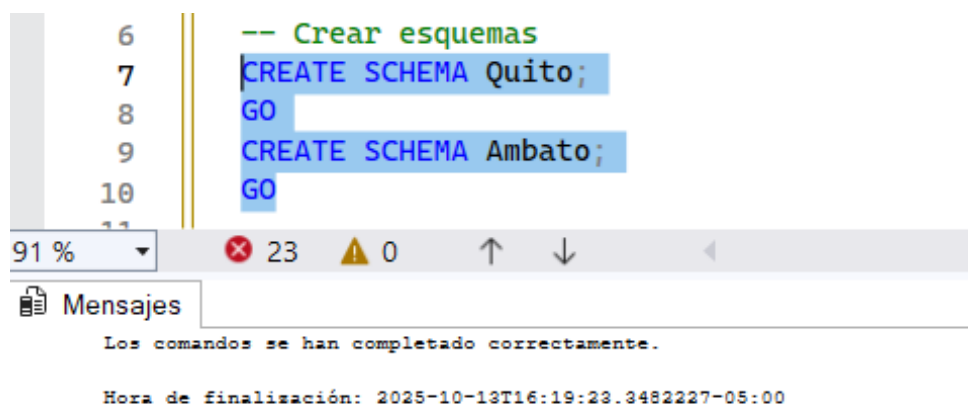


Fig 8. Script de la creación de los esquemas Ambato y Quito en el Nodo_A

Crear Tablas en Esquema Quito:

En el esquema **Quito** se implementaron tres tablas principales:

- **Centro:** almacena la información básica de cada centro médico.
- **Medico:** contiene los datos personales y profesionales de los médicos.
- **Pacientes:** contiene los datos principales de los pacientes.
- **Consulta:** registra las consultas realizadas, incluyendo referencias a las tablas de centro y médico mediante claves foráneas.

Se crea además un index para cada tabla el cual le indica al motor de la base de datos en qué posición están los datos, para no tener que revisar fila por fila cuando haces una consulta.



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN
CICLO ACADÉMICO: AGOSTO 2025 – ENERO 2026



```
17 -- Tablas en esquema Quito
18 CREATE TABLE Quito.CentrosMedicos (
19     CentroID INT PRIMARY KEY,
20     Nombre NVARCHAR(50) NOT NULL,
21     Ciudad NVARCHAR(50) NOT NULL,
22 );
23
24 CREATE TABLE Quito.Medicos (
25     CedMedicoID NVARCHAR(10) PRIMARY KEY,
26     Nombre NVARCHAR(50) NOT NULL,
27     Especialidad NVARCHAR(50) NOT NULL,
28     CentroID INT NOT NULL FOREIGN KEY REFERENCES Quito.CentrosMedicos(CentroID)
29 );
30
31 CREATE TABLE Quito.Pacientes (
32     CedPacienteID NVARCHAR(50) PRIMARY KEY,
33     Nombre NVARCHAR(50) NOT NULL,
34     Email NVARCHAR(100)
35 );
36
37 CREATE TABLE Quito.Consultas (
38     ConsultaID NVARCHAR(50) PRIMARY KEY,
39     CedPacienteID NVARCHAR(50) NOT NULL FOREIGN KEY REFERENCES Quito.Pacientes(CedPacienteID),
40     Fecha DATE NOT NULL,
41     CentroID INT NOT NULL FOREIGN KEY REFERENCES Quito.CentrosMedicos(CentroID),
42     CedMedicoID NVARCHAR(10) NOT NULL FOREIGN KEY REFERENCES Quito.Medicos(CedMedicoID)
43 );
44
45 -- Índices Quito
46 CREATE INDEX idx_Consultas_CentroID_Quito ON Quito.Consultas(CentroID);
47 CREATE INDEX idx_Consultas_Fecha_Quito ON Quito.Consultas(Fecha);
48 CREATE INDEX idx_Consultas_PacienteID_Quito ON Quito.Consultas(CedPacienteID);
49 CREATE INDEX idx_Consultas_MedicoID_Quito ON Quito.Consultas(CedMedicoID);
50
51
```

Mensajes

Los comandos se han completado correctamente.

Hora de finalización: 2025-10-10T21:22:23.4916494-05:00

83 % No se encontraron problemas.

Consulta ejecutada correctamente. KHEVIN\NODO_A (16.0 RTM)

Fig 9. Creación de tablas en el esquema Quito

Crear Tabla en Esquema Ambato:

De igual forma, en el esquema Ambato se crearon las tablas y los index equivalentes, manteniendo la estructura homogénea entre nodos.

```
52 -- Tablas en esquema Ambato
53 CREATE TABLE Ambato.CentrosMedicos (
54     CentroID INT PRIMARY KEY,
55     Nombre NVARCHAR(50) NOT NULL,
56     Ciudad NVARCHAR(50) NOT NULL,
57 );
58
59 CREATE TABLE Ambato.Medicos (
60     CedMedicoID NVARCHAR(10) PRIMARY KEY,
61     Nombre NVARCHAR(50) NOT NULL,
62     Especialidad NVARCHAR(50) NOT NULL,
63     CentroID INT NOT NULL FOREIGN KEY REFERENCES Ambato.CentrosMedicos(CentroID)
64 );
65
66 CREATE TABLE Ambato.Pacientes (
67     CedPacienteID NVARCHAR(50) PRIMARY KEY,
68     Nombre NVARCHAR(50) NOT NULL,
69     Email NVARCHAR(100)
70 );
71
72 CREATE TABLE Ambato.Consultas (
73     ConsultaID NVARCHAR(50) PRIMARY KEY,
74     CedPacienteID NVARCHAR(50) NOT NULL FOREIGN KEY REFERENCES Ambato.Pacientes(CedPacienteID),
75     Fecha DATE NOT NULL,
76     CentroID INT NOT NULL FOREIGN KEY REFERENCES Ambato.CentrosMedicos(CentroID),
77     CedMedicoID NVARCHAR(10) NOT NULL FOREIGN KEY REFERENCES Ambato.Medicos(CedMedicoID)
78 );
79
80 -- Índices Ambato
81 CREATE INDEX idx_Consultas_CentroID_Ambato ON Ambato.Consultas(CentroID);
82 CREATE INDEX idx_Consultas_Fecha_Ambato ON Ambato.Consultas(Fecha);
83 CREATE INDEX idx_Consultas_PacienteID_Ambato ON Ambato.Consultas(CedPacienteID);
84 CREATE INDEX idx_Consultas_MedicoID_Ambato ON Ambato.Consultas(CedMedicoID);
85
86
```

Mensajes

Los comandos se han completado correctamente.

Hora de finalización: 2025-10-10T21:22:35.0465121-05:00

83 % No se encontraron problemas.

Consulta ejecutada correctamente. KHEVIN\NODO_A (16.0 RTM)

Fig 10. Creación de tablas en el esquema Ambato



NODO_B

Crear Base de Datos:

En el Nodo_B se generó la base de datos que contiene los esquemas de las ciudades **Cuenca** y **Guayaquil**, simulando una segunda instancia lógica del sistema distribuido.

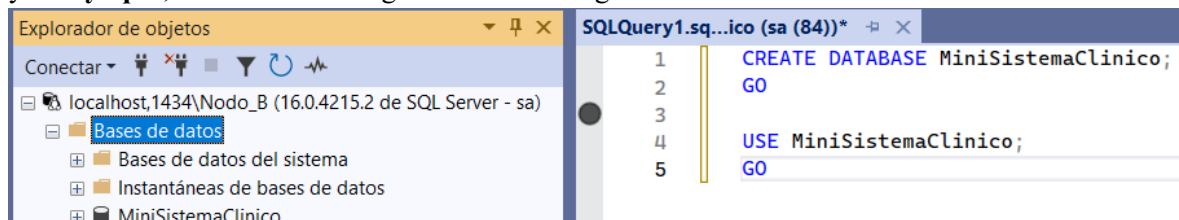


Fig 11. Creación de la base de datos en el Nodo_B

Crear Esquemas:

Al igual que en el Nodo_A, se definieron los esquemas Cuenca y Guayaquil para mantener independencia de datos por ubicación

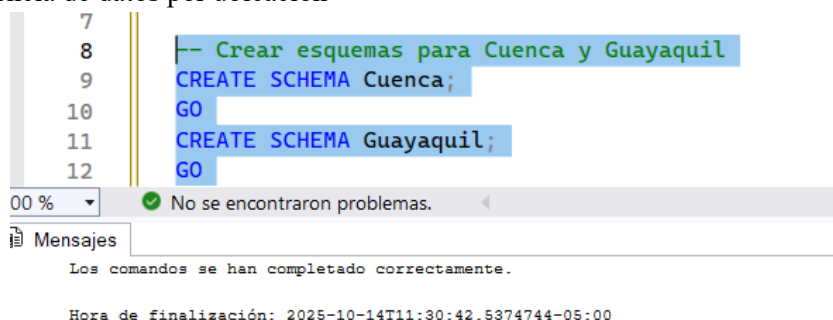


Fig 12. Creación de esquemas Cuenca y Guayaquil en el Nodo_B

Crear Tablas en Esquema Cuenca:

De igual forma, en el esquema Cuenca del Nodo B se crearon las tablas y los index equivalentes, manteniendo la estructura homogénea entre nodos.

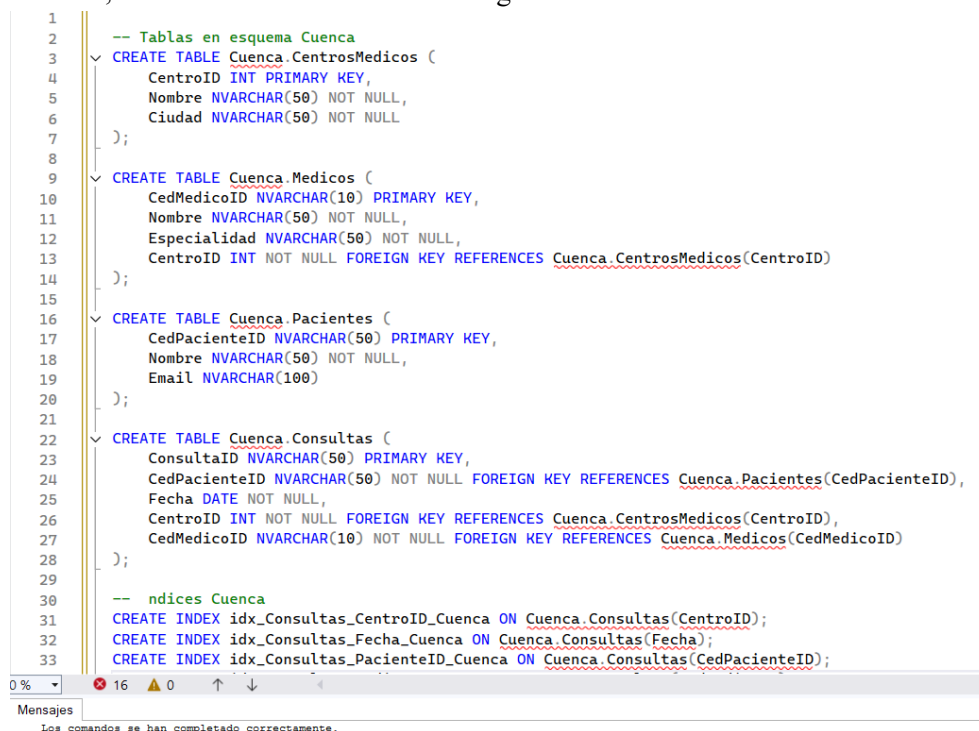


Fig 13. Creación de tablas en esquema Cuenca



Crear Tabla en Esquema Guayaquil:

De igual forma, en el esquema Guayaquil del Nodo B se crearon las tablas y los index equivalentes, manteniendo la estructura homogénea entre nodos.

```
SQLQuery2.sql (sa (53)) *
1  -- Tablas en esquema Guayaquil
2  CREATE TABLE Guayaquil.CentrosMedicos (
3      CentroID INT PRIMARY KEY,
4      Nombre NVARCHAR(50) NOT NULL,
5      Ciudad NVARCHAR(50) NOT NULL
6  );
7
8  CREATE TABLE Guayaquil.Medicos (
9      CedMedicoID NVARCHAR(10) PRIMARY KEY,
10     Nombre NVARCHAR(50) NOT NULL,
11     Especialidad NVARCHAR(50) NOT NULL,
12     CentroID INT NOT NULL FOREIGN KEY REFERENCES Guayaquil.CentrosMedicos(CentroID)
13 );
14
15 CREATE TABLE Guayaquil.Pacientes (
16     CedPacienteID NVARCHAR(50) PRIMARY KEY,
17     Nombre NVARCHAR(50) NOT NULL,
18     Email NVARCHAR(100)
19 );
20
21 CREATE TABLE Guayaquil.Consultas (
22     ConsultaID NVARCHAR(50) PRIMARY KEY,
23     CedPacienteID NVARCHAR(50) NOT NULL FOREIGN KEY REFERENCES Guayaquil.Pacientes(CedPacienteID),
24     Fecha DATE NOT NULL,
25     CentroID INT NOT NULL FOREIGN KEY REFERENCES Guayaquil.CentrosMedicos(CentroID),
26     CedMedicoID NVARCHAR(10) NOT NULL FOREIGN KEY REFERENCES Guayaquil.Medicos(CedMedicoID)
27 );
28
29 -- ndices Guayaquil
30 CREATE INDEX idx_Consultas_CentroID_Guayaquil ON Guayaquil.Consultas(CentroID);
31 CREATE INDEX idx_Consultas_Fecha_Guayaquil ON Guayaquil.Consultas(Fecha);
32 CREATE INDEX idx_Consultas_PacienteID_Guayaquil ON Guayaquil.Consultas(CedPacienteID);
33 CREATE INDEX idx_Consultas_MedicoID_Guayaquil ON Guayaquil.Consultas(CedMedicoID);
```

Mensajes
Los comandos se han completado correctamente.

Fig 14. Creación de tablas en el esquema Guayaquil

2. Carga de datos (10 pts).

NODO_A

Inserción de datos en esquema Ambato:

Se realizó la inserción de datos en ambos nodos, cumpliendo con el mínimo de:

- 4 Centros médicos: Ambato, Quito, Cuenca y Guayaquil.
- 6 Médicos distribuidos entre los distintos centros.
- 20 Consultas médicas, distribuidas proporcionalmente por ciudad.



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN
CICLO ACADÉMICO: AGOSTO 2025 – ENERO 2026



```
89  -- Centros en Ambato
90  INSERT INTO Ambato.CentrosMedicos (CentroID, Nombre, Ciudad) VALUES
91  (1, 'Centro Ambato', 'Ambato');
92
93  -- Médicos en Ambato
94  INSERT INTO Ambato.Medicos (CedMedicoID, Nombre, Especialidad, CentroID) VALUES
95  ('1800000001', 'Dr. Juan Pérez', 'Cardiología', 1),
96  ('1800000002', 'Dra. Ana Gómez', 'Pediatría', 1),
97  ('1800000003', 'Dr. Luis Torres', 'Neurología', 1);
98
99  -- Pacientes (Ambato)
100 INSERT INTO Ambato.Pacientes (CedPacienteID, Nombre, Email) VALUES
101 ('1900000001', 'José Martínez', 'jose@mail.com'),
102 ('1900000002', 'Lucía Herrera', 'lucia@mail.com'),
103 ('1900000003', 'Carlos Andrade', 'carlos@mail.com'),
104 ('1900000004', 'María López', 'maria@mail.com'),
105 ('1900000005', 'Pedro Sánchez', 'pedro@mail.com');
106
107 -- Consultas en Ambato
108 INSERT INTO Ambato.Consultas (ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID) VALUES
109 ('CAMB001', '1900000001', '2025-10-01', 1, '1800000001'),
110 ('CAMB002', '1900000002', '2025-10-02', 1, '1800000001'),
111 ('CAMB003', '1900000003', '2025-10-03', 1, '1800000002'),
112 ('CAMB004', '1900000004', '2025-10-07', 1, '1800000002'),
113 ('CAMB005', '1900000005', '2025-10-09', 1, '1800000003');
114
115
116
```

Mensajes

(1 fila afectada)

(2 filas afectadas)

(5 filas afectadas)

(5 filas afectadas)

Hora de finalización: 2025-10-13T21:24:42.4262543-05:00

91 %

No se encontraron problemas.

Consulta ejecutada correctamente.

KHEVIN\NODO_A (16.0 RTM)

Fig 15. Inserción de datos en el esquema Ambato

Inserción de Datos en esquema Quito:

Del mismo modo, se cargaron los registros de médicos y consultas pertenecientes al centro Quito.

```
116  -- Centros en Quito
117  INSERT INTO Quito.CentrosMedicos (CentroID, Nombre, Ciudad) VALUES
118  (2, 'Centro Quito', 'Quito');
119
120  -- Médicos en Quito
121  INSERT INTO Quito.Medicos (CedMedicoID, Nombre, Especialidad, CentroID) VALUES
122  ('1800000004', 'Dra. Marta Ruiz', 'Dermatología', 2),
123  ('1800000005', 'Dra. Ana Morales', 'Pediatría', 2),
124  ('1800000006', 'Dr. Jose Torres', 'Neurología', 2);
125
126  -- Pacientes (Quito)
127  INSERT INTO Quito.Pacientes (CedPacienteID, Nombre, Email) VALUES
128  ('1900000006', 'Andrea Navas', 'andrea@mail.com'),
129  ('1900000007', 'Luis Mora', 'luis@mail.com'),
130  ('1900000008', 'Fernanda Salas', 'fernanda@mail.com'),
131  ('1900000009', 'Jorge Torres', 'jorge@mail.com'),
132  ('1900000010', 'Carla Rojas', 'carla@mail.com');
133
134  -- Consultas en Quito
135  INSERT INTO Quito.Consultas (ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID) VALUES
136  ('CQUI001', '1900000006', '2025-10-04', 2, '1800000004'),
137  ('CQUI002', '1900000007', '2025-10-05', 2, '1800000005'),
138  ('CQUI003', '1900000008', '2025-10-06', 2, '1800000004'),
139  ('CQUI004', '1900000009', '2025-10-08', 2, '1800000004'),
140  ('CQUI005', '1900000010', '2025-10-10', 2, '1800000006');
141
142
143
```

Mensajes

(1 fila afectada)

(3 filas afectadas)

(5 filas afectadas)

(5 filas afectadas)

Hora de finalización: 2025-10-13T21:25:43.8023995-05:00

91 %

No se encontraron problemas.

Consulta ejecutada correctamente.

KHEVIN\NODO_A (16.0 RTM)

Fig 16. Inserción de datos en el esquema Quito

NODO_B

Inserción de datos en esquema Cuenca:

Del mismo modo, se cargaron los registros de médicos y consultas pertenecientes al centro Cuenca perteneciente al Nodo B.



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN
CICLO ACADÉMICO: AGOSTO 2025 – ENERO 2026



```
1  -- Centros en Cuenca
2  INSERT INTO Cuenca.CentrosMedicos (CentroID, Nombre, Ciudad) VALUES
3  (3, 'Centro Cuenca', 'Cuenca');
4
5  -- Médicos en Cuenca
6  INSERT INTO Cuenca.Medicos (CedMedicoID, Nombre, Especialidad, CentroID) VALUES
7  ('1800000007', 'Dr. Pedro Castro', 'Cardiología', 3),
8  ('1800000008', 'Dra. Silvia Moreno', 'Pediatría', 3),
9  ('1800000009', 'Dr. Carlos Vega', 'Neurología', 3);
10
11 -- Pacientes Cuenca
12 INSERT INTO Cuenca.Pacientes (CedPacienteID, Nombre, Email) VALUES
13 ('1900000011', 'Marcos Pinto', 'marcos@mail.com'),
14 ('1900000012', 'Sofía Ramírez', 'sofia@mail.com'),
15 ('1900000013', 'Daniela Torres', 'daniela@mail.com'),
16 ('1900000014', 'Rafael Mejía', 'raul@mail.com'),
17 ('1900000015', 'Camila López', 'camila@mail.com');
18
19 -- Consultas en Cuenca
20 INSERT INTO Cuenca.Consultas (ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID) VALUES
21 ('CCUE001', '1900000011', '2025-10-01', 3, '1800000007'),
22 ('CCUE002', '1900000012', '2025-10-02', 3, '1800000008'),
23 ('CCUE003', '1900000013', '2025-10-03', 3, '1800000009'),
24 ('CCUE004', '1900000014', '2025-10-04', 3, '1800000007'),
25 ('CCUE005', '1900000015', '2025-10-05', 3, '1800000008');
```

Mensajes

(1 fila afectada)

(3 filas afectadas)

(5 filas afectadas)

(5 filas afectadas)

Hora de finalización: 2025-10-14T11:00:13.8272003-05:00

Fig 17. Inserción de datos en el esquema Cuenca

Inserción de Datos en esquema Guayaquil:

Del mismo modo, se cargaron los registros de médicos y consultas pertenecientes al centro Guayaquil perteneciente al Nodo B.

```
SQLQuery2.sql...ico (sa (53))
1  -- Centros en Guayaquil
2  INSERT INTO Guayaquil.CentrosMedicos (CentroID, Nombre, Ciudad) VALUES
3  (4, 'Centro Guayaquil', 'Guayaquil');
4
5  -- Médicos en Guayaquil
6  INSERT INTO Guayaquil.Medicos (CedMedicoID, Nombre, Especialidad, CentroID) VALUES
7  ('1800000010', 'Dr. Miguel Ramírez', 'Cardiología', 4),
8  ('1800000011', 'Dra. Laura Pérez', 'Pediatría', 4),
9  ('1800000012', 'Dr. Jorge Morales', 'Neurología', 4);
10
11 -- Pacientes Guayaquil
12 INSERT INTO Guayaquil.Pacientes (CedPacienteID, Nombre, Email) VALUES
13 ('1900000016', 'Valeria Torres', 'valeria@mail.com'),
14 ('1900000017', 'Andrés Gómez', 'andres@mail.com'),
15 ('1900000018', 'Natalia Cruz', 'natalia@mail.com'),
16 ('1900000019', 'Javier Paredes', 'javier@mail.com'),
17 ('1900000020', 'Daniel Flores', 'daniel@mail.com');
18
19 -- Consultas en Guayaquil
20 INSERT INTO Guayaquil.Consultas (ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID) VALUES
21 ('CGYE001', '1900000016', '2025-10-01', 4, '1800000010'),
22 ('CGYE002', '1900000017', '2025-10-02', 4, '1800000011'),
23 ('CGYE003', '1900000018', '2025-10-03', 4, '1800000012'),
24 ('CGYE004', '1900000019', '2025-10-04', 4, '1800000010'),
25 ('CGYE005', '1900000020', '2025-10-05', 4, '1800000011');
```

Mensajes

(1 fila afectada)

(3 filas afectadas)

(5 filas afectadas)

(5 filas afectadas)

Hora de finalización: 2025-10-14T11:09:26.5586155-05:00

Fig 18. Inserción de datos en el esquema Guayaquil

3. Pruebas CRUD locales (10 pts).

Demuestra INSERT/UPDATE/DELETE y 2 consultas analíticas (p.ej. consultas por rango de Fecha y top de Especialidades en un centro).

NODO_A

Prueba de INSERT:

Se realizaron inserciones de registros en las tablas del esquema **Ambato** mediante sentencias INSERT INTO.



```
155  -- =====  
156  -- Pruebas CRUD locales  
157  -- =====  
158  -- CREATE (INSERT):  
159  INSERT INTO Ambato.Pacientes (CedPacienteID, Nombre, Email)  
160  VALUES ('1900000011', 'Mario Zamora', 'mario@mail.com');  
100 % 14 0  
Mensajes  
(1 fila afectada)  
Hora de finalización: 2025-10-13T19:30:43.9399427-05:00
```

Fig 19. Ejecución de operación INSERT en el esquema Ambato

Prueba de SELECT:

Se realizaron consultas básicas para verificar la correcta inserción y estructura de los datos en las tablas. Extrayendo información específica del sistema, como el listado de pacientes registrados en una ciudad determinada, en este caso Ambato.

```
162  -- READ (SELECT):  
163  SELECT * FROM Ambato.Pacientes;  
164  
100 % 14 0  
Resultados Mensajes  


|   | CedPacienteID | Nombre         | Email           |
|---|---------------|----------------|-----------------|
| 1 | 1900000001    | José Martínez  | jose@mail.com   |
| 2 | 1900000002    | Lucía Herrera  | lucia@mail.com  |
| 3 | 1900000003    | Carlos Andrade | carlos@mail.com |
| 4 | 1900000004    | María López    | maria@mail.com  |
| 5 | 1900000005    | Pedro Sánchez  | pedro@mail.com  |
| 6 | 1900000011    | Mario Zamora   | mario@mail.com  |


```

Fig 20. Ejecución de operación SELECT de los PACIENTES en el esquema Ambato

Prueba de UPDATE:

La operación UPDATE se utilizó para modificar información ya almacenada, como en la Fig 15 un cambio del email.

```
164  -- UPDATE:  
165  UPDATE Ambato.Pacientes  
166  SET Email = 'mario.zamora@mail.com'  
167  WHERE CedPacienteID = '1900000011';  
168  
100 % 14 0  
Mensajes  
(1 fila afectada)  
Hora de finalización: 2025-10-13T19:32:49.9522134-05:00
```

Fig 21. Ejecución de operación UPDATE de un paciente en el esquema Ambato



Prueba de DELETE:

Se llevaron a cabo operaciones de eliminación controlada de registros, utilizando cláusulas WHERE, como en la Fig 16 con la cedula.

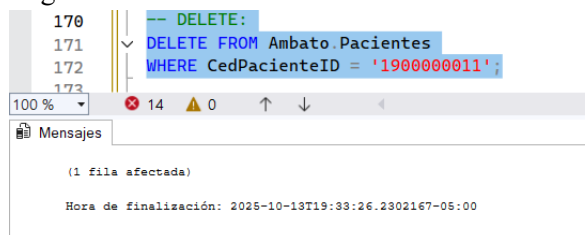


Fig 22. Ejecución de operación DELETE a un paciente en el esquema Ambato

Consulta Analítica 1:

Esta consulta tiene como objetivo analizar la cantidad de consultas realizadas en un período de tiempo específico. Es útil para evaluar la carga de trabajo o estacionalidad en la atención médica.

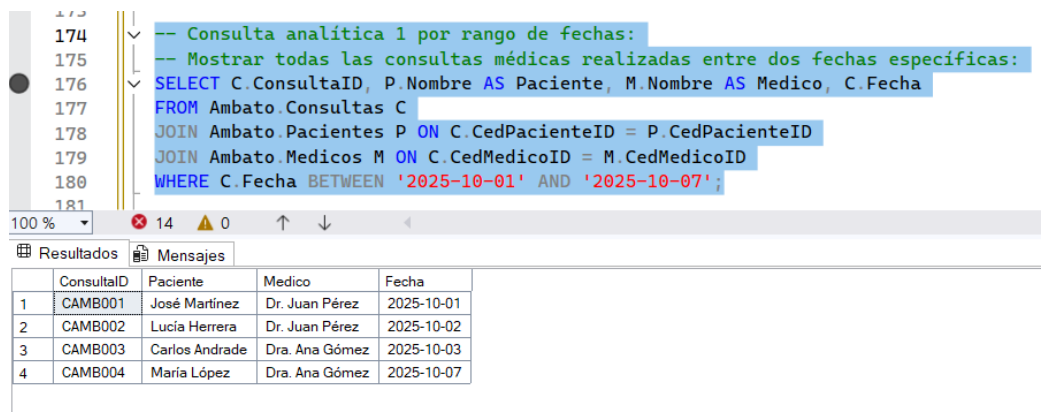


Fig 23. Consulta que muestra diferentes datos de consultas filtradas por fechas

Consulta Analítica 2:

El propósito de esta consulta es obtener un TOP de las especialidades médicas más demandadas en un centro específico. Se utiliza una combinación de JOIN y GROUP BY para agrupar los resultados por especialidad.

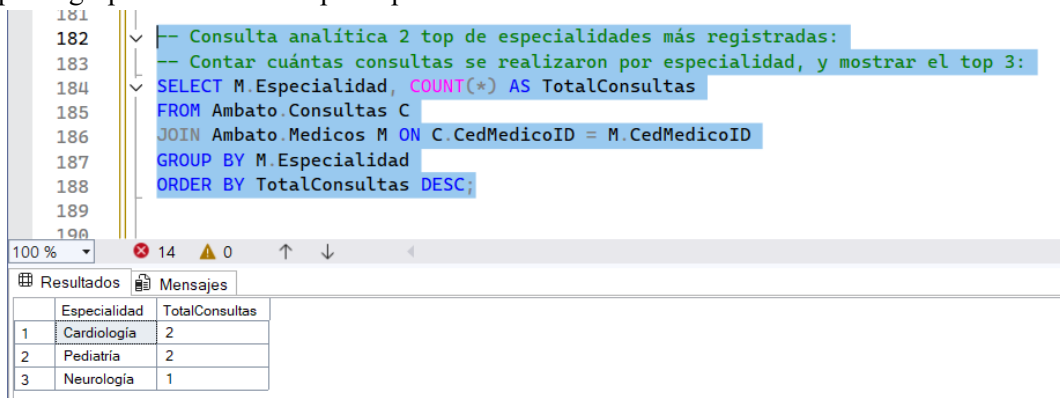


Fig 24. Consulta que devuelve top 3, la cantidad de consultas realizadas por especialidad



NODO_B

Prueba de INSERT:

Se realizaron inserciones de registros en las tablas del esquema CUENCA mediante sentencias INSERT INTO.

```
-- CREATE (INSERT):
INSERT INTO Cuenca.Pacientes (CedPacienteID, Nombre, Email)
VALUES ('1900000021', 'Patricio Vega', 'patricio@mail.com');
```

(1 fila afectada)

Hora de finalización: 2025-10-14T11:40:36.3955666-05:00

Fig 25. Ejecución de operación SELECT de los PACIENTES en el esquema Cuenca

Prueba de SELECT:

Se realizaron consultas básicas extrayendo información específica del sistema, como el listado de pacientes registrados en la ciudad de Cuenca.

```
-- READ (SELECT):
SELECT * FROM Cuenca.Pacientes;
```

	CedPacienteID	Nombre	Email
1	1900000011	Marcos Pinto	marcos@mail.com
2	1900000012	Sof a Ram rez	sofia@mail.com
3	1900000013	Daniela Torres	daniela@mail.com
4	1900000014	Ra l Mej a	raul@mail.com
5	1900000015	Camila L pez	camila@mail.com
6	1900000021	Patricio Vega	patricio@mail.com

Fig 26. Ejecución de operación SELECT de los PACIENTES en el esquema Cuenca

Prueba de UPDATE:

La operación UPDATE se utilizó para modificar información ya almacenada, como en la Fig 21 un cambio del email.

```
-- UPDATE:
UPDATE Cuenca.Pacientes
SET Email = 'patricio.vega@mail.com'
WHERE CedPacienteID = '1900000021';
```

(1 fila afectada)

Hora de finalización: 2025-10-14T11:43:00.7309595-05:00

Fig 27. Ejecución de operación UPDATE de un paciente en el esquema Cuenca



Prueba de DELETE:

Se llevaron a cabo operaciones de eliminación controlada de registros, utilizando cláusulas WHERE.

```
12  |
13  |
14  |  -- DELETE:
15  |  DELETE FROM Cuenca.Pacientes
    |  WHERE CedPacienteID = '1900000021';
```

100 % 10 0

Mensajes

Hora de finalización: 2025-10-14T11:43:00.7309595-05:00

Fig 28. Ejecución de operación DELETE a un paciente en el esquema Cuenca

Consulta Analítica 1:

Esta consulta tiene como objetivo analizar la cantidad de consultas realizadas en un período de tiempo específico. Es útil para evaluar la carga de trabajo o estacionalidad en la atención médica.

```
6  |
7  |  -- Consulta analítica 1 - Rango de fechas:
8  |  -- Consultas médicas entre fechas específicas
9  |  SELECT C.ConsultaID, P.Nombre AS Paciente, M.Nombre AS Medico, C.Fecha
10 |  FROM Cuenca.Consultas C
11 |  JOIN Cuenca.Pacientes P ON C.CedPacienteID = P.CedPacienteID
12 |  JOIN Cuenca.Medicos M ON C.CedMedicoID = M.CedMedicoID
13 |  WHERE C.Fecha BETWEEN '2025-10-01' AND '2025-10-07';
```

90 % No se encontraron problemas.

Resultados Mensajes

	ConsultaID	Paciente	Medico	Fecha
1	CCUE001	Marcos Pinto	Dr. Pedro Castro	2025-10-01
2	CCUE002	Sofía Ramírez	Dra. Silvia Moreno	2025-10-02
3	CCUE003	Daniela Torres	Dr. Carlos Vega	2025-10-03
4	CCUE004	Raúl Mejía	Dr. Pedro Castro	2025-10-04
5	CCUE005	Camila López	Dra. Silvia Moreno	2025-10-05

Fig 29. Consulta analítica de volumen de consultas filtradas por rango de fechas en Cuenca.

Consulta Analítica 2:

El propósito de esta consulta es obtener un TOP de las especialidades médicas más demandadas en el centro específico de Cuenca.

```
14 |
15 |  -- Consulta analítica 2 - Top especialidades más registradas:
16 |  SELECT M.Especialidad, COUNT(*) AS TotalConsultas
17 |  FROM Cuenca.Consultas C
18 |  JOIN Cuenca.Medicos M ON C.CedMedicoID = M.CedMedicoID
19 |  GROUP BY M.Especialidad
20 |  ORDER BY TotalConsultas DESC;
21 |  GO
```

90 % No se encontraron problemas.

Resultados Mensajes

	Especialidad	TotalConsultas
1	Cardiología	2
2	Pediatría	2
3	Neurología	1

Fig 30. Consulta \$TOP\$ 3 de especialidades médicas más demandadas en el centro de Cuenca.

B. Fragmentación y capa global (40 pts)

4. Fragmentación horizontal (15 pts)

Implementación de dos nodos lógicos:

Se realiza **Dos bases/instancias** que contendrán una base de datos con dos esquemas para la representación de la distribución de datos, en la cual se distribuirá de la siguiente manera:



- **Nodo_A: Centros = Ambato y Quito**

El Nodo_A almacena todos los registros de las tablas Centros, Médicos y Consultas que pertenecen únicamente a las ciudades de Ambato y Quito. Esta fragmentación permite que las transacciones locales en estas sedes se ejecuten sin depender del Nodo_B.

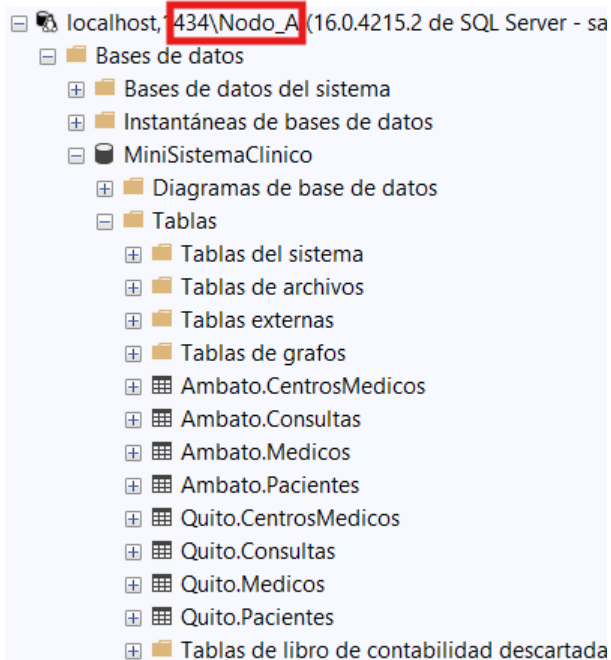


Fig 31. Representación de la Fragmentación Horizontal en el \$Nodo_A\$, mostrando solo los datos de Ambato y Quito.

- **Nodo_B: Centros = Cuenca y Guayaquil**

El Nodo_B contiene los fragmentos correspondientes a los centros médicos de Cuenca y Guayaquil. Esto completa la distribución de todos los datos del sistema, garantizando que todos los registros de las tablas originales han sido divididos.

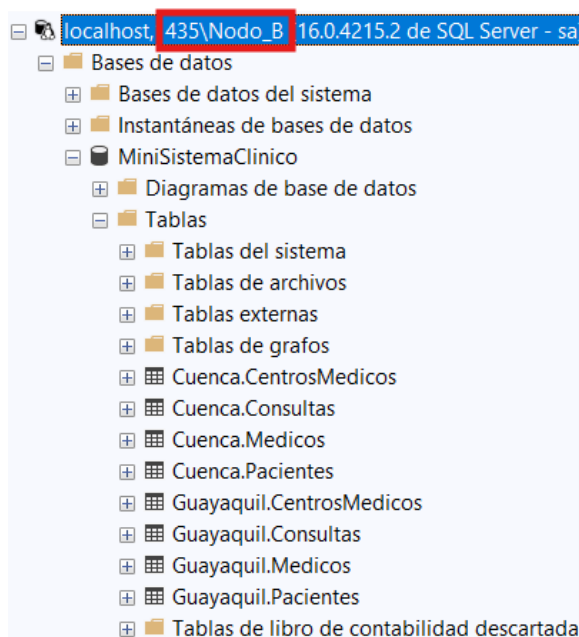


Fig 32. Configuración de Servidores Enlazados en SQL Server para permitir la comunicación entre los nodos lógicos.



Se ejecutaron los procedimientos `sp_addlinkedserver` y `sp_addlinkedsrvlogin` en el `Nodo_A` para crear un enlace hacia el `Nodo_B`. Esta configuración permite al `Nodo_A` referenciar y consultar objetos del `Nodo_B`, cumpliendo el requisito de transparencia de localización a nivel de motor.

```
5  -- Fragmentación horizontal
6  --
7  --
8  -- En Nodo_A crear linked server hacia la instancia Nodo_B
9  EXEC sp_addlinkedserver
10     @server = 'Nodo_B',          -- Nombre con el que se identifica el servidor remoto
11     @srvproduct = '',           -- Nombre del producto (puede dejarse vacío para SQL Server)
12     @provider = 'MSOLEDBSQL',    -- Proveedor OLE DB que permite la conexión (SQL Native Client)
13     @datasrc = 'nodo_b';        -- Dirección IP o nombre del servidor remoto
14
15  -- login mapping (usar autenticación SQL)
16  EXEC sp_addlinkedsrvlogin
17     @rmtsrvname = 'Nodo_B',
18     @useself = 'false',
19     @locallogin = NULL,
20     @rmtuser = 'sa',             -- o el usuario SQL del otro servidor
21     @rmtpassword = '12345Aa!';
22
```

90 % No se encontraron problemas.

	SRV_NAME	SRV_PROVIDERNAME	SRV_PRODUCT	SRV_DATASOURCE	SRV_PROVIDERSTRING	SRV_LOCATION	SRV_CAT
1	23991923odcc	SQLNCLI	SQL Server	23991923odcc	NULL	NULL	NULL
2	Nodo_B	MSOLEDBSQL		nodo_b	NULL	NULL	NULL

Fig 33. Configuración del Servidor Enlazado en el `Nodo_A` hacia el `Nodo_B` mediante `sp_addlinkedserver` y `sp_addlinkedsrvlogin`.

5. Capa global con transparencia (15 pts)

Se implementa una vista global que permite consultar datos como si fuera una sola tabla, en la que mostrara todos los datos de todos los esquemas (Quito, Ambato, Cuenca y Guayaquil):

- `vw_ConsultasGlobal` = `SELECT ... FROM Nodo_A.Consultas UNION ALL SELECT ... FROM Nodo B.Consultas`

```
216  --
217  -- Capa global con transparencia
218  --
219  -- Vista global
220  CREATE VIEW vw_ConsultasGlobal AS
221  -- Nodo_A local
222  SELECT ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID
223  FROM Ambato.Consultas
224  UNION ALL
225  SELECT ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID
226  FROM Quito.Consultas
227  -- Nodo_B via linked server
228  UNION ALL
229  SELECT ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID
230  FROM Nodo_B.MiniSistemaClinico.Cuenca.Consultas
231  UNION ALL
232  SELECT ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID
233  FROM Nodo_B.MiniSistemaClinico.Guayaquil.Consultas;
234
```

100 % 28 0

Mensajes

Los comandos se han completado correctamente.

Hora de finalización: 2025-10-13T20:58:51.0754489-05:00

Fig 34. Script de la creación de la vista global `vw_ConsultasGlobal`, uniendo los fragmentos de la tabla `Consultas` de los esquemas locales y remotos mediante `UNION ALL`.



- Se crea consultas que demuestren transparencia de localización (usuario consulta la vista sin saber dónde están los datos). (Transparencia de localización y de fragmentación.)

Usuario consulta la vista sin saber dónde están los datos:

	ConsultaID	CedPacienteID	Fecha	CentrolID	CedMedicolID
1	CAMB001	1900000001	2025-10-01	1	1800000001
2	CAMB002	1900000002	2025-10-02	1	1800000001
3	CAMB003	1900000003	2025-10-03	1	1800000002
4	CAMB004	1900000004	2025-10-07	1	1800000002
5	CAMB005	1900000005	2025-10-09	1	1800000003
6	CQUI001	1900000006	2025-10-04	2	1800000004
7	CQUI002	1900000007	2025-10-05	2	1800000005
8	CQUI003	1900000008	2025-10-06	2	1800000004
9	CQUI004	1900000009	2025-10-08	2	1800000004
10	CQUI005	1900000010	2025-10-10	2	1800000006
11	CCUE001	1900000011	2025-10-01	3	1800000007
12	CCUE002	1900000012	2025-10-02	3	1800000008
13	CCUE003	1900000013	2025-10-03	3	1800000009
14	CCUE004	1900000014	2025-10-04	3	1800000007
15	CCUE005	1900000015	2025-10-05	3	1800000008
16	CGYE001	1900000016	2025-10-01	4	1800000010
17	CGYE002	1900000017	2025-10-02	4	1800000011
18	CGYE003	1900000018	2025-10-03	4	1800000012
19	CGYE004	1900000019	2025-10-04	4	1800000010
20	CGYE005	1900000020	2025-10-05	4	1800000011

Fig 35. El resultado de la consulta SELECT sobre vw_ConsultasGlobal consolida los datos de Ambato, Quito, Cuenca y Guayaquil, demostrando la Transparencia de Localización y Fragmentación al usuario.

Transparencia de localización y de fragmentación:

	ConsultaID	CedPacienteID	Fecha	CentrolID	CedMedicolID
1	CAMB001	1900000001	2025-10-01	1	1800000001
2	CAMB002	1900000002	2025-10-02	1	1800000001
3	CAMB003	1900000003	2025-10-03	1	1800000002
4	CQUI001	1900000006	2025-10-04	2	1800000004
5	CQUI002	1900000007	2025-10-05	2	1800000005
6	CCUE001	1900000011	2025-10-01	3	1800000007
7	CCUE002	1900000012	2025-10-02	3	1800000008
8	CCUE003	1900000013	2025-10-03	3	1800000009
9	CCUE004	1900000014	2025-10-04	3	1800000007
10	CCUE005	1900000015	2025-10-05	3	1800000008
11	CGYE001	1900000016	2025-10-01	4	1800000010
12	CGYE002	1900000017	2025-10-02	4	1800000011
13	CGYE003	1900000018	2025-10-03	4	1800000012
14	CGYE004	1900000019	2025-10-04	4	1800000010
15	CGYE005	1900000020	2025-10-05	4	1800000011

Fig 36. Consulta a la vista vw_ConsultasGlobal con filtro por rango de Fecha.

La consulta sigue siendo transparente para el usuario final, quien solo usa la vista, y demuestra que el filtro se aplica correctamente a todos los fragmentos.

El usuario no sabe que los datos están fragmentados (usa solo vw_ConsultasGlobal), por lo tanto, se demuestra la **Transparencia de Fragmentación**.

Además, no sabe dónde están físicamente (Nodo_B o Nodo_A), por lo que demuestra la **Transparencia de Localización**.



6. Consulta distribuida optimizada (10 pts)

El objetivo de esta sección es demostrar la ejecución de una consulta que involucra datos de ambos nodos y explicar cómo se minimiza el movimiento de datos a través de la red, que es el mayor factor de costo en los sistemas distribuidos.

Consulta Distribuida:

Se diseñó una consulta que accede a la vista global (\$vw_ConsultasGlobal\$) pero incluye un filtro por un rango de fecha y una ciudad específica.

```
246 -- =====
247 -- Consulta distribuida optimizada
248 -- =====
249 -- Ejemplo de consulta: Se desea conocer las consultas entre el 1 y el 5 de octubre de 2025 solo para Quito o Cuenca.
250 SELECT ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID
251 FROM Quito.Consultas
252 WHERE Fecha BETWEEN '2025-10-01' AND '2025-10-05' -- Nodo_A: Quito
253 AND CentroID = 2 -- Quito CentroID
254
255 UNION ALL
256
257 SELECT ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID
258 FROM Nodo_B_MiniSistemaClinico.Cuenca.Consultas
259 WHERE Fecha BETWEEN '2025-10-01' AND '2025-10-05' -- Nodo_B: Cuenca
260 AND CentroID = 3;
261
262 -- Nota: No incluimos Guayaquil porque no nos interesa según el filtro de ciudad
263
264
```

ConsultaID	CedPacienteID	Fecha	CentroID	CedMedicoID	
1	CQUI001	1900000006	2025-10-04	2	1800000004
2	CQUI002	1900000007	2025-10-05	2	1800000005
3	CCUE001	1900000011	2025-10-01	3	1800000007
4	CCUE002	1900000012	2025-10-02	3	1800000008
5	CCUE003	1900000013	2025-10-03	3	1800000009
6	CCUE004	1900000014	2025-10-04	3	1800000007
7	CCUE005	1900000015	2025-10-05	3	1800000008

Fig 37. Ejecución de una consulta distribuida que utiliza la vista global vw_ConsultasGlobal.

Consulta global con datos distribuidos

Al aplicar un filtro que involucra un predicado de fragmentación (implícitamente, la ciudad) y un predicado de restricción (rango de fecha), la consulta demuestra la capacidad de proceso distribuido.

```
265 -- Misma consulta aplicando filtro a vw.ConsultasGlobal
266 SELECT ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID
267 FROM vw.ConsultasGlobal
268 WHERE Fecha BETWEEN '2025-10-01' AND '2025-10-05'
269 AND CentroID IN (2, 3); -- Quito y Cuenca
270
271
```

ConsultaID	CedPacienteID	Fecha	CentroID	CedMedicoID	
1	CQUI001	1900000006	2025-10-04	2	1800000004
2	CQUI002	1900000007	2025-10-05	2	1800000005
3	CCUE001	1900000011	2025-10-01	3	1800000007
4	CCUE002	1900000012	2025-10-02	3	1800000008
5	CCUE003	1900000013	2025-10-03	3	1800000009
6	CCUE004	1900000014	2025-10-04	3	1800000007
7	CCUE005	1900000015	2025-10-05	3	1800000008

Fig 38. Consulta global con filtros por rango de \$Fecha\$ y ciudad. El resultado consolida los datos filtrados de los fragmentos.

Explicación: La estrategia para **minimizar el movimiento de datos** en una consulta global distribuida se centra en aplicar las **operaciones más restrictivas** lo más cerca posible de la fuente de datos es decir el nodo local (Nodo_A), mediante el uso de cláusulas en **nodos** y **esquemas específicos** como el la Fig 37, pedir solo las **columnas estrictamente necesarias** como en la Fig 38 y tener en cuenta el orden de operaciones para asegurar el menor costo de procesamiento distribuido.



C. Replicación y tolerancia a fallos (opcional fuerte / 20 pts)

7. Replicación de referencia (10 pts)

Se realiza una **Réplica Simple** de todas las tablas hacia otro nodo (**Nodo_C**) para copiar todos los datos que están en el **Nodo_A** y **Nodo_B** y garantizar la **disponibilidad de los datos**.

Preparación de nodos:

Se realiza la configuración para tener un Servidor Enlazado **Nodo_C** al **Nodo_A**, lo cual permite que la **instancia local** (**Nodo_A**) de SQL Server pueda **consultar y manipular datos** que residen en una **instancia remota** (**Nodo_C**).

```
Nodo_A.sql - ...ster (sa (53))  x  Nodo_B.sql - master (sa (78))  SQLQuery1.sql ...nico (sa (76))*  SQLQuery2.sql ...nico (sa (76))

273
274
275 -- Replicación de referencia
276
277 -- En Nodo_A crear linked server hacia la instancia Nodo_C para BackUp
278 EXEC sp_addlinkedserver
279     @server = 'Nodo_C',          -- Nombre con el que se identifica el servidor remoto
280     @srvproduct = '',           -- Nombre del producto (puede dejarse vacío para SQL Server)
281     @provider = 'MSOLEDBSQL',   -- Proveedor OLE DB que permite la conexión (SQL Native Client)
282     @datasrc = 'nodo_c';       -- Dirección IP o nombre del servidor remoto
283
284 -- login mapping (usar autenticación SQL)
285 EXEC sp_addlinkedsrvlogin
286     @rmtsrvname = 'Nodo_C',
287     @useself = 'false',
288     @locallogin = NULL,
289     @rmtuser = 'sa',
290     @rmtpassword = '12345Aa!';
```

Fig 39. Configuración del Servidor Enlazado el **Nodo_A** hacia la instancia de réplica para el respaldo y la alta disponibilidad.

De igual forma se configura un Servidor Enlazado **Nodo_C** al **Nodo_B**:

```
Nodo_A.sql - master (sa (53))  x  Nodo_B.sql - master (sa (78))  x  SQLQuery1.sql ...nico (sa (76))*  SQLQuery2.sql ...nico (sa (76))

190
191
192 -- Replicación de referencia
193
194 -- En Nodo_A crear linked server hacia la instancia Nodo_C para BackUp
195 EXEC sp_addlinkedserver
196     @server = 'Nodo_C',          -- Nombre con el que se identifica el servidor remoto
197     @srvproduct = '',           -- Nombre del producto (puede dejarse vacío para SQL Server)
198     @provider = 'MSOLEDBSQL',   -- Proveedor OLE DB que permite la conexión (SQL Native Client)
199     @datasrc = 'nodo_c';       -- Dirección IP o nombre del servidor remoto
200
201 -- login mapping (usar autenticación SQL)
202 EXEC sp_addlinkedsrvlogin
203     @rmtsrvname = 'Nodo_C',
204     @useself = 'false',
205     @locallogin = NULL,
206     @rmtuser = 'sa',
207     @rmtpassword = '12345Aa!';
```

Fig 40. Configuración del Servidor Enlazado en el **Nodo_B** hacia la instancia de réplica para asegurar la disponibilidad total de los datos.



Creación de Base de datos que contendrá la replicación de datos en Nodo_C:
El Nodo_C contendrá los datos de ambas instancias por lo que el modelado de la base que replicara los datos serán una unión de todos los nodos y esquemas:

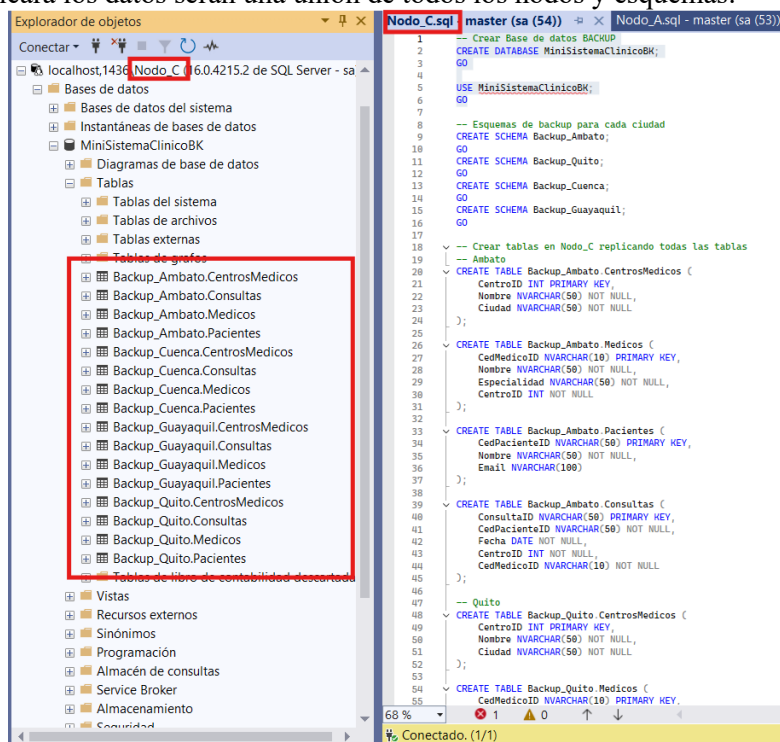


Fig 41. Creación de la base de datos MiniSistemaClinico en el Nodo_C y definición de esquemas de backup y tablas consolidadas para recibir la réplica total de los datos distribuidos.

Replicación Inicial:

Se copia los datos iniciales del Nodo_A y Nodo_B hacia el Nodo_C para mantener la replicación consistente de todos los datos iniciales mediante la sentencia INSERT hacia el nodo remoto Nodo_C desde las tablas y esquemas locales.

```
293 -- Copiar DATOS INICIALES de Ambato
294 -- CentrosMedicos
295 INSERT INTO Nodo_C_MiniSistemaClinicoBK.Backup_Ambato.CentrosMedicos (CentroID, Nombre, Ciudad)
296 SELECT CentroID, Nombre, Ciudad
297 FROM Ambato.CentrosMedicos;
298
299 -- Medicos
300 INSERT INTO Nodo_C_MiniSistemaClinicoBK.Backup_Ambato.Medicos (CedMedicoID, Nombre, Especialidad, CentroID)
301 SELECT CedMedicoID, Nombre, Especialidad, CentroID
302 FROM Ambato.Medicos;
303
304 -- Pacientes
305 INSERT INTO Nodo_C_MiniSistemaClinicoBK.Backup_Ambato.Pacientes (CedPacienteID, Nombre, Email)
306 SELECT CedPacienteID, Nombre, Email
307 FROM Ambato.Pacientes;
308
309 -- Consultas
310 INSERT INTO Nodo_C_MiniSistemaClinicoBK.Backup_Ambato.Consultas (ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID)
311 SELECT ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID
312 FROM Ambato.Consultas;
313
314 -- Copiar DATOS INICIALES de Quito
315 -- CentrosMedicos
316 INSERT INTO Nodo_C_MiniSistemaClinicoBK.Backup_Quito.CentrosMedicos (CentroID, Nombre, Ciudad)
317 SELECT CentroID, Nombre, Ciudad
318 FROM Quito.CentrosMedicos;
319
320 -- Medicos
321 INSERT INTO Nodo_C_MiniSistemaClinicoBK.Backup_Quito.Medicos (CedMedicoID, Nombre, Especialidad, CentroID)
322 SELECT CedMedicoID, Nombre, Especialidad, CentroID
323 FROM Quito.Medicos;
324
325 -- Pacientes
326 INSERT INTO Nodo_C_MiniSistemaClinicoBK.Backup_Quito.Pacientes (CedPacienteID, Nombre, Email)
327 SELECT CedPacienteID, Nombre, Email
328 FROM Quito.Pacientes;
329
330 -- Consultas
331 INSERT INTO Nodo_C_MiniSistemaClinicoBK.Backup_Quito.Consultas (ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID)
332 SELECT ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID
333 FROM Quito.Consultas;
```

Fig 42. Script de la Replicación Inicial de Datos del Nodo_A hacia el Nodo_C, copiando las estructuras de CentrosMedicos, Medicos, Pacientes y Consultas para establecer una base de réplica consistente.



Se ejecutó la misma lógica de inserción para trasladar los datos del Nodo_B hacia el Nodo_C. Esto asegura que el Nodo_C contenga una copia completa y consistente de todos los fragmentos del sistema distribuido antes de implementar los mecanismos de sincronización en tiempo real.

```
213 -- Copiar DATOS INICIALES de Cuenca
214 -- CentrosMedicos
215 INSERT INTO Nodo_C_MiniSistemaClinicoBK_Backup_Cuenca.CentrosMedicos (CentroID, Nombre, Ciudad)
216 SELECT CentroID, Nombre, Ciudad
217 FROM Cuenca.CentrosMedicos;
218
219 -- Medicos
220 INSERT INTO Nodo_C_MiniSistemaClinicoBK_Backup_Cuenca.Medicos (CedMedicoID, Nombre, Especialidad, CentroID)
221 SELECT CedMedicoID, Nombre, Especialidad, CentroID
222 FROM Cuenca.Medicos;
223
224 -- Pacientes
225 INSERT INTO Nodo_C_MiniSistemaClinicoBK_Backup_Cuenca.Pacientes (CedPacienteID, Nombre, Email)
226 SELECT CedPacienteID, Nombre, Email
227 FROM Cuenca.Pacientes;
228
229 -- Consultas
230 INSERT INTO Nodo_C_MiniSistemaClinicoBK_Backup_Cuenca.Consultas (ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID)
231 SELECT ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID
232 FROM Cuenca.Consultas;
233
234 -- Copiar DATOS INICIALES de Guayaquil
235 -- CentrosMedicos
236 INSERT INTO Nodo_C_MiniSistemaClinicoBK_Backup_Guayaquil.CentrosMedicos (CentroID, Nombre, Ciudad)
237 SELECT CentroID, Nombre, Ciudad
238 FROM Guayaquil.CentrosMedicos;
239
240 -- Medicos
241 INSERT INTO Nodo_C_MiniSistemaClinicoBK_Backup_Guayaquil.Medicos (CedMedicoID, Nombre, Especialidad, CentroID)
242 SELECT CedMedicoID, Nombre, Especialidad, CentroID
243 FROM Guayaquil.Medicos;
244
245 -- Pacientes
246 INSERT INTO Nodo_C_MiniSistemaClinicoBK_Backup_Guayaquil.Pacientes (CedPacienteID, Nombre, Email)
247 SELECT CedPacienteID, Nombre, Email
248 FROM Guayaquil.Pacientes;
249
250 -- Consultas
251 INSERT INTO Nodo_C_MiniSistemaClinicoBK_Backup_Guayaquil.Consultas (ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID)
252 SELECT ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID
253 FROM Guayaquil.Consultas;
```

Fig 43. Script para la Replicación Inicial de Datos del Nodo_B (Cuenca y Guayaquil) hacia el Nodo_C, consolidando todos los fragmentos del sistema en el nodo de réplica.

Replicación de datos mediante gatillos:

Se implementó un esquema de **Replicación de Datos** utilizando **Triggers** en las tablas. Esta estrategia asegura la **sincronización inmediata** de las operaciones de INSERT, UPDATE y DELETE entre el nodo primario Nodo_A o Nodo_B y el Nodo_C, mejorando la **disponibilidad** de los datos y permitiendo la distribución de la carga de lectura.

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Explorador de objetos' (Object Explorer) pane displays the database structure for 'Nodo_A' (16.0.4215.2 de SQL Server - sa). The 'MiniSistemaClinico' database is expanded, showing 'Tablas' (Tables) and 'Triggers'. The 'Triggers' folder is expanded, showing triggers for 'Ambato' and 'Quito' schemas. The 'Ambato' schema triggers are listed as follows:

NombreDelTrigger	EsquemaDeLaTabla	NombreDeLaTabla
trg_Insert_CentrosMedicos_Ambato	Ambato	CentrosMedicos
trg_Update_CentrosMedicos_Ambato	Ambato	CentrosMedicos
trg_Delete_CentrosMedicos_Ambato	Ambato	CentrosMedicos
trg_Insert_Medicos_Ambato	Ambato	Medicos
trg_Update_Medicos_Ambato	Ambato	Medicos
trg_Delete_Medicos_Ambato	Ambato	Medicos
trg_Insert_Pacientes_Ambato	Ambato	Pacientes
trg_Update_Pacientes_Ambato	Ambato	Pacientes
trg_Delete_Pacientes_Ambato	Ambato	Pacientes
trg_Insert_Consultas_Ambato	Ambato	Consultas
trg_Update_Consultas_Ambato	Ambato	Consultas
trg_Delete_Consultas_Ambato	Ambato	Consultas

The 'Quito' schema triggers are listed as follows:

NombreDelTrigger	EsquemaDeLaTabla	NombreDeLaTabla
trg_Insert_CentrosMedicos_Quito	Quito	CentrosMedicos
trg_Update_CentrosMedicos_Quito	Quito	CentrosMedicos
trg_Delete_CentrosMedicos_Quito	Quito	CentrosMedicos
trg_Insert_Medicos_Quito	Quito	Medicos
trg_Update_Medicos_Quito	Quito	Medicos
trg_Delete_Medicos_Quito	Quito	Medicos
trg_Insert_Pacientes_Quito	Quito	Pacientes
trg_Update_Pacientes_Quito	Quito	Pacientes
trg_Delete_Pacientes_Quito	Quito	Pacientes
trg_Insert_Consultas_Quito	Quito	Consultas
trg_Update_Consultas_Quito	Quito	Consultas
trg_Delete_Consultas_Quito	Quito	Consultas

The right pane shows the 'SQLQuery4.sql...ico (sa (58))' window with the following SQL query:

```
USE MiniSistemaClinico;
GO
-- Ver triggers
SELECT
    name AS NombreDelTrigger,
    OBJECT_SCHEMA_NAME(parent_id) AS EsquemaDeLaTabla,
    OBJECT_NAME(parent_id) AS NombreDeLaTabla
FROM
    sys.triggers
WHERE
    is_ms_shipped = 0; -- Excluye triggers internos del sistema
GO
```

The bottom status bar indicates 'Consulta ejecutada correctamente.'

Fig 44. Lista de Triggers implementados en las tablas de los esquemas Ambato y Quito del Nodo_A. Estos gatillos aseguran la sincronización inmediata de las operaciones INSERT, UPDATE y DELETE hacia el Nodo_C.



The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'Explorador de objetos' (Object Explorer) pane displays the database structure for 'Nodo_B'. The 'Bases de datos' (Databases) folder is expanded, showing 'MiniSistemaClinico'. Under 'Tablas' (Tables), the 'Cuenca' and 'Guayaquil' schemas are listed. The 'Triggers' folder is expanded, showing a list of triggers for both schemas. On the right, the 'SQLQuery5.sql' window displays the following query:

```
USE MiniSistemaClinico;
GO
-- Ver triggers
SELECT
    name AS NombreDelTrigger,
    OBJECT_SCHEMA_NAME(parent_id) AS EsquemaDeLaTabla,
    OBJECT_NAME(parent_id) AS NombreDeLaTabla
FROM
    sys.triggers
WHERE
    is_ms_shipped = 0; -- Excluye triggers internos del sistema
GO
```

The query results are displayed in a table with the following columns: 'NombreDelTrigger', 'EsquemaDeLaTabla', and 'NombreDeLaTabla'. The results show 24 triggers, grouped by schema (Cuenca and Guayaquil) and table (CentrosMedicos, Medicos, Pacientes, Consultas). The status bar at the bottom indicates 'Consulta ejecutada correctamente.' (Query executed successfully).

Fig 45. Lista de Triggers implementados en las tablas de los esquemas Cuenca y Guayaquil del Nodo_B. Estos triggers aseguran la sincronización inmediata de las operaciones INSERT, UPDATE y DELETE hacia el Nodo_C

Verificación de replicación inmediata:

Para la verificación se realizará una inserción de una Consulta en el Nodo_A

The screenshot shows the SQL Server Enterprise Manager interface. The 'SQLQuery5.sql' window displays the following query:

```
-- Visualizacion de datos ANTES de la insercion
SELECT * FROM vw_ConsultasGlobal WHERE ConsultaID = 'CAMB006';

-- Consulta ingresada en Nodo_A
INSERT INTO Ambato.Consultas (ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID) VALUES
('CAMB006', '1900000005', '2025-10-15', 1, '1800000001');

-- Visualizacion de datos DESPUES de la insercion
SELECT * FROM vw_ConsultasGlobal WHERE ConsultaID = 'CAMB006';
```

The query results are displayed in a table with the following columns: 'ConsultaID', 'CedPacienteID', 'Fecha', 'CentroID', and 'CedMedicoID'. The results show one row of data, corresponding to the inserted record.

Fig 46. Verificación de los datos mediante las operaciones de SELECT e INSERT en el Nodo_A (Ambato y Quito)

Y se realiza una consulta en el Nodo_C para verificar la sincronización de datos:



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN
CICLO ACADÉMICO: AGOSTO 2025 – ENERO 2026



	ConsultaID	CedPacienteID	Fecha	ControlID	CedMedicoID
1	CAMB001	1900000001	2025-10-01	1	1800000001
2	CAMB002	1900000002	2025-10-02	1	1800000001
3	CAMB003	1900000003	2025-10-03	1	1800000002
4	CAMB004	1900000004	2025-10-07	1	1800000002
5	CAMB005	1900000005	2025-10-09	1	1800000003
6	CAMB006	1900000005	2025-10-15	1	1800000001

Fig 47 Verificación de la sincronización en el Nodo_C (respaldo) mediante una consulta en la tabla de Ambato

8. Prueba de fallo y recuperación (10 pts)

Para probar la tolerancia a fallos y la transparencia de replicación/disponibilidad, se simuló la caída de uno de los nodos, ya que gracias a utilizar docker es fácil la Simulación de la caída de un NODO.

Detener el servicio SQL del Nodo_A:

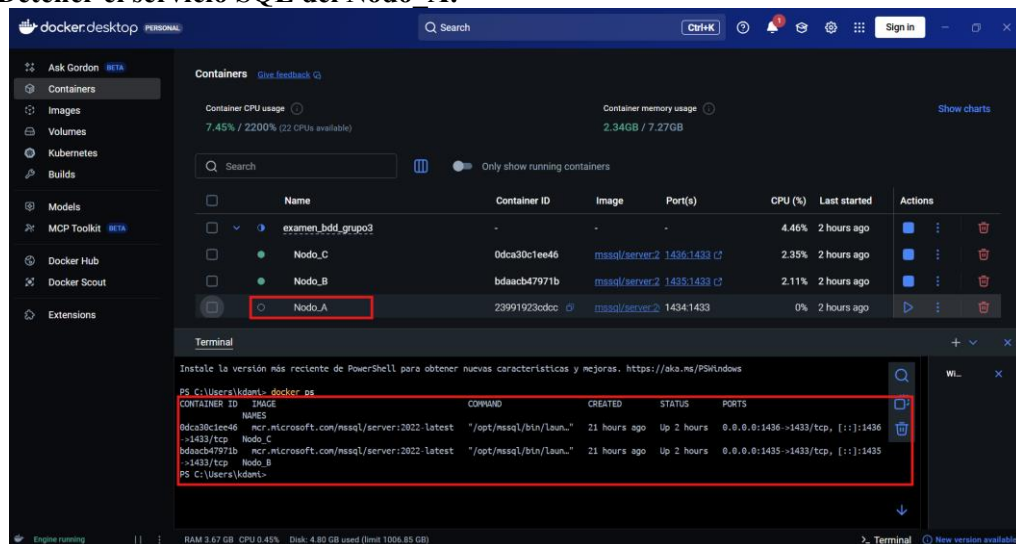


Fig 48. Servidor del Nodo_A desactivado en la aplicación Docker Desktop

Mostrar qué consultas siguen funcionando:

Ya que en el Nodo_A se encuentran datos de los esquemas de Quito y Ambato y mientras Nodo_A está caído:

- Las tablas locales del **Nodo_B** siguen activas: Cuenca y Guayaquil, por lo tanto, se pueden seguir haciendo consultas.



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN
CICLO ACADÉMICO: AGOSTO 2025 – ENERO 2026



```
Nodo_C.sql - master (sa (60))
Nodo_B.sql - ...nico (sa (58))
Nodo_A.sql - master (sa (60))

142 USE MiniSistemaClinico;
143 GO
144 SELECT * FROM Cuenca CentrosMedicos;
145 SELECT * FROM Cuenca Medicos;
146 SELECT * FROM Cuenca Pacientes;
147 SELECT * FROM Cuenca Consultas;
148
149 SELECT * FROM Guayaquil CentrosMedicos;
150 SELECT * FROM Guayaquil Medicos;
151 SELECT * FROM Guayaquil Pacientes;
152 SELECT * FROM Guayaquil Consultas;
```

82 % 56 0

Resultados Mensajes

CentroID	Nombre	Ciudad
1	Centro Cuenca	Cuenca

CedMedicoID	Nombre	Especialidad	CentroID
1	1800000007	Dr. Pedro Castro	Cardiología
2	1800000008	Dra. Silvia Moreno	Pediatría
3	1800000009	Dr. Carlos Vega	Neurología

CedPacienteID	Nombre	Email
1	1900000011	Marcos Pinto marcos@mail.com
2	1900000012	Sofia Ramirez sofia@mail.com
3	1900000013	Daniela Torres daniela@mail.com
4	1900000014	Raul Mejia raul@mail.com
5	1900000015	Camila López camila@mail.com

ConsultaID	CedPacienteID	Fecha	CentroID	CedMedicoID
1	CCUE001	1900000011	2025-10-01	3
2	CCUE002	1900000012	2025-10-02	3
3	CCUE003	1900000013	2025-10-03	3
4	CCUE004	1900000014	2025-10-04	3
5	CCUE005	1900000015	2025-10-05	3

CentroID	Nombre	Ciudad
1	Centro Guayaquil	Guayaquil

CedMedicoID	Nombre	Especialidad	CentroID
1	1800000010	Dr. Miguel Ramirez	Cardiología
2	1800000011	Dra. Laura Pérez	Pediatría
3	1800000012	Dr. Jorge Morales	Neurología

CedPacienteID	Nombre	Email
1	1900000016	Valeria Torres valeria@mail.com

Consulta ejecutada correctamente.

localhost:1435/Nodo_B (16.0... sa (58) MiniSistemaClinico 00:00:00 28 filas

Fig 49. Verificación que los esquemas y las tablas del **Nodo_B** (Cuenca y Guayaquil) todavía sigue activos a pesar de que el **Nodo_A** fue apagado

- Las consultas globales usando el **Nodo_C** muestran los datos de los nodos que aún están activos y ya replicados:

138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE TECNOLOGÍAS DE LA INFORMACIÓN
CICLO ACADÉMICO: AGOSTO 2025 – ENERO 2026



The screenshot shows a database query execution window. The SQL code is as follows:

```
-- ANTES de caída de nodo A
431 USE MiniSistemaClinico;
432 GO
433 SELECT * FROM Ambato.CentrosMedicos;
434 SELECT * FROM Ambato.Medicos;
435 SELECT * FROM Ambato.Pacientes;
436 SELECT * FROM Ambato.Consultas;
437
438
```

The results are displayed in a table with the following data:

CentroID	Nombre	Ciudad
1	Centro Ambato	Ambato

CedMedicoID	Nombre	Especialidad	CentroID
1	Dr. Juan Pérez	Cardiología	1
2	Dra. Ana Gómez	Pediatría	1
3	Dr. Luis Torres	Neurología	1

CedPacienteID	Nombre	Email
1	José Martínez	jose@mail.com
2	Lucía Herrera	lucia@mail.com
3	Carlos Andr...	carlos@mail.c...
4	Maria I. óñez	maria@mail.c...

The status bar at the bottom indicates: "Consulta ejecutada correctamente. localhost,1434\Nodo_A (16.0... sa (53) MiniSistemaClinico 00:00:00 15 filas". A red arrow points to the status bar.

Fig 51. Nodo_A antes de ser apagado, con 15 filas de datos

Luego de una inserción al esquema Backup_Ambato del Nodo_C:

The screenshot shows a database query execution window. The SQL code is as follows:

```
165 -- Prueba de fallo y recuperación
166 -- Prueba de fallo y recuperación
167
168 -- Inserción de un dato en Ambato
169 INSERT INTO Backup_Ambato Consultas (ConsultaID, CedPacienteID, Fecha, CentroID, CedMedicoID) VALUES
170 ('CAMB007', '1900000001', '2025-10-15', 1, '1800000001');
```

The status bar at the bottom indicates: "1 fila afectada". The message pane shows: "Hora de finalización: 2025-10-15T17:40:10.2180807-05:00".

Fig 52. Inserción al esquema del Nodo_C llamado Ambato con el Nodo_A (Ambato y Quito) apagado

Cuando el nodo a vuelva estar disponible ejecutamos el **procedimiento de recuperación** de datos que fueron modificados en el Nodo_C y **mantenemos la sincronización de replicas** cuando **falla un Nodo** o se **pierde la conexión**.



```
454
455 -- Recuperar tras caída
456 EXEC sp_resincronizar_nodoA;
1157
120 % 99+ 0 120 % Línea: 455 Carácter: 1 SPC
Mensajes
--- Iniciando resincronización del Nodo A desde Nodo C (MiniSistemaClinicoBK.Backup_Ambato) ---
--- Resincronización completada exitosamente. ---
Hora de finalización: 2025-10-15T17:48:47.8726292-05:00
```

```
120 % No se encontraron problemas. Línea: 5 Carácter: 1 TABULACIONES
Consulta ejecutada correctamente. localhost,1434\Nodo_A (16.0... sa (97) MiniSistemaClinico 00:00:00
```

Fig 53. Nodo_A activado se realizará el proceso de recuperación de datos modificados en el respaldo (Nodo_C)

Y como podemos observar los datos vuelven a estar disponibles en el Nodo_A:

```
474 -- DESPUES de caída de nodo A
475 USE MiniSistemaClinico;
476 GO
477 SELECT * FROM Ambato.CentrosMedicos;
478 SELECT * FROM Ambato.Medicos;
479 SELECT * FROM Ambato.Pacientes;
480 SELECT * FROM Ambato.Consultas;
```

82 % 99+ 0 82 % Línea: 474 Carácter: 1 SPC CRLF

ControlID	Nombre	Ciudad
1	Centro Ambato	Ambato

CedMedicoID	Nombre	Especialidad	ControlID
1	Dr. Juan Pérez	Cardiología	1
2	Dra. Ana Gómez	Pediatría	1
3	Dr. Luis Torres	Neurología	1

CedPacienteID	Nombre	Email
1	José Martínez	jose@mail.com
2	Lucía Herrera	lucia@mail.com
3	Carlos Andr...	carlos@mail.c...
4	María I. Áñez	maria@mail.c...

Consulta ejecutada correctamente. localhost,1434\Nodo_A (16.0... sa (97) MiniSistemaClinico 00:00:00 16 filas

Fig 54. Nodo_A con el procedimiento de recuperación de datos ya listo, mostrando 16 filas después de hacer un SELECT a todas sus tablas, dando a entender que los datos vuelven a estar disponibles y sin ningún fallo mientras el nodo_A estaba apagado

2.7 Resultados obtenidos

El proyecto ha logrado implementar con éxito un sistema de bases de datos distribuido y tolerante a fallos, virtualizado mediante Docker. La estrategia central se basó en la fragmentación horizontal de los datos por ciudades a través de dos nodos lógicos (Nodo_A y Nodo_B), garantizando así la independencia local de las transacciones. Para asegurar una mejor experiencia para el usuario, se implementó una capa global de transparencia mediante vistas. Además, se validó la capacidad de optimizar el rendimiento de las consultas distribuidas. Finalmente, se estableció un tercer nodo de réplica (Nodo_C) con sincronización inmediata vía triggers, demostrando la alta disponibilidad y la tolerancia a fallos del sistema al verificar que las operaciones de lectura se mantuvieron activas y la consistencia de los datos se recuperó tras la simulación de la caída de un nodo principal.

2.8 Habilidades blandas empleadas en la práctica

- ☐ Liderazgo
- ☒ Trabajo en equipo
- ☐ Comunicación asertiva
- ☐ La empatía
- ☐ Pensamiento crítico
- ☐ Flexibilidad



- ☐ La resolución de conflictos
- ☐ Adaptabilidad
- ☒ Responsabilidad

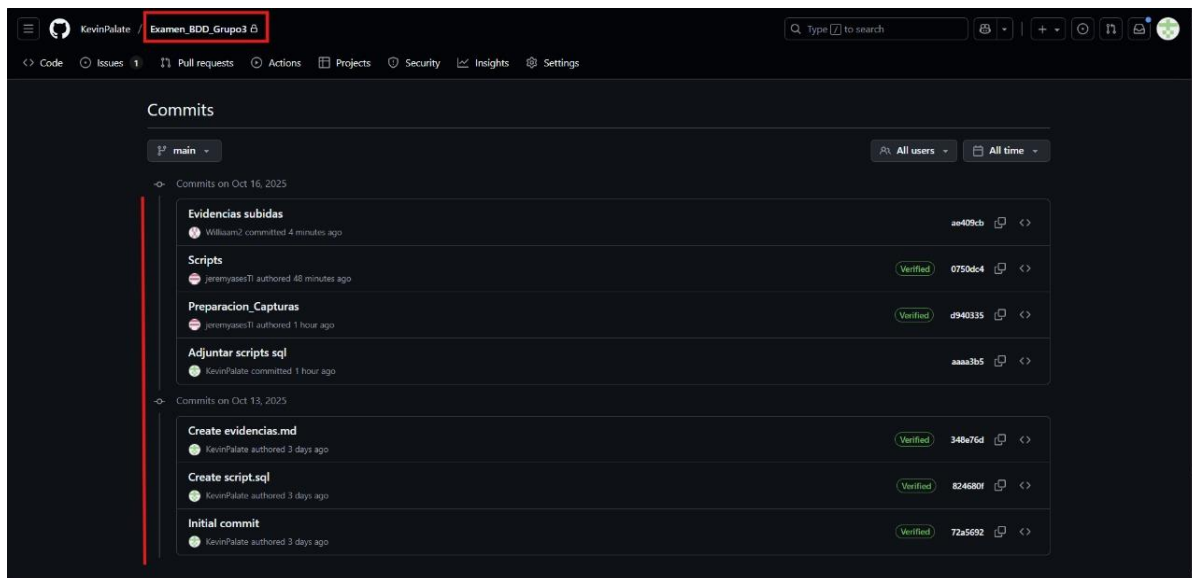


Fig 1. Evidencias Colaborativas

2.9 Conclusiones

En conclusión, el proyecto implementó exitosamente una arquitectura de bases de datos distribuida y de alta disponibilidad. Logramos la fragmentación horizontal de los datos por ciudad en dos nodos (Nodo_A y Nodo_B), lo que optimiza las operaciones locales. El éxito principal fue la creación de una capa de transparencia que, mediante vistas, permite a los usuarios acceder a todos los datos de forma unificada sin conocer su ubicación física. Finalmente, la integración de un tercer nodo de réplica con sincronización inmediata garantizó la tolerancia a fallos. Esto se verificó al simular la caída de un nodo, demostrando que el sistema mantiene la continuidad del servicio y recupera la consistencia de los datos, resultando en una solución robusta y escalable.

2.10 Recomendaciones

Repartir mejor el trabajo: Enviar las consultas de solo lectura (como ver reportes) a la copia de seguridad. Así, los servidores principales se concentran en guardar nuevos datos y el sistema se vuelve más rápido en general.

Usar métodos de copia de seguridad más eficientes: Actualmente usamos triggers para hacer las copias instantáneas, pero estos a veces pueden hacer más lento el guardado de nuevos datos. Lo ideal es cambiar a las herramientas de copia que la base de datos ya trae, que son más rápidas y ligeras al momento de registrar información.



2.11 Referencias bibliográficas

- [1] J. (vijayasimha Br), “Running Microsoft SQL database server with Docker”, *Medium*, 04-nov-2024. [En línea]. Disponible en: <https://vijayasimhabr.medium.com/running-microsoft-sql-database-server-with-docker-5a879ef06adb>. [Consultado: 16-oct-2025].
- [2] WilliamDAssafMSFT, “Crear servidores vinculados - SQL Server”, Microsoft.com. [En línea]. Disponible en: <https://learn.microsoft.com/es-es/sql/relational-databases/linked-servers/create-linked-servers-sql-server-database-engine?view=sql-server-ver17>. [Consultado: 16-oct-2025].
- [3] “CREATE TRIGGER (transact-SQL)”, Microsoft.com. [En línea]. Disponible en: <https://learn.microsoft.com/en-us/sql/t-sql/statements/create-trigger-transact-sql?view=sql-server-ver17>. [Consultado: 16-oct-2025].