



INFORME DE GUÍA PRÁCTICA

I. PORTADA

Tema:	Tratamiento de transacciones
Unidad de Organización Curricular:	PROFESIONAL
Nivel y Paralelo:	5 “A” TI
Alumnos participantes:	Aldas Jordan Wellington Ismael Caguasango Bayas Alex Patricio Gómez Llerena Luis Fernando Paredes Barrera Luis Enrique
Asignatura:	Sistemas de Bases de Datos Distribuidos
Docente:	Ing. Jose Ruben Caiza Caizabuano, Mg.

II. INFORME DE GUÍA PRÁCTICA

2.1 Objetivos

General:

Determinar el comportamiento de un SGBD con transacciones.

Específicos:

- Implementar transacciones en SQL Server aplicando los comandos BEGIN TRANSACTION, COMMIT y ROLLBACK para evidenciar el principio de atomicidad en operaciones de bases de datos.
- Analizar el comportamiento del SGBD frente a la concurrencia de sesiones y niveles de aislamiento, verificando cómo afectan los bloqueos y la consistencia de los datos.
- Evaluar técnicas de manejo de errores (SET XACT_ABORT, TRY...CATCH) para garantizar la integridad de la base de datos ante fallos durante la ejecución de transacciones.

2.2 Modalidad

Presencial.

2.3 Tiempo de duración

Presenciales: 6

No presenciales: 0

2.4 Instrucciones

Conéctese al motor de base de datos

- Cree una BD llamada Universidad
- Cree las tablas A(a char(1) PK, B(b char(1) referenciada a A, C(c char(1)
- Ingrese algunos datos y verifiquemos la integridad referencial
- Creamos transacciones y probamos las características de atomicidad Commit y Rollback
- Habilitamos una nueva sesión para pruebas
- Manejamos errores On_error, set xact_abort y Try



2.5 Listado de equipos, materiales y recursos

Listado de equipos y materiales generales empleados en la guía práctica:

- Inteligencia artificial, TAC
- Computador
- SQL Server

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 por desarrollar

CREACIÓN DE UNA NUEVA INSTANCIA DENTRO DE SQL SERVER

Para iniciar el diseño de la base de datos distribuida simulada, y dentro de ella realizar la practica sobre transacciones se creó una nueva instancia de SQL Server denominada **Transacciones**. Esta instancia es necesaria porque permite trabajar de forma independiente con los esquemas y tablas que representan cada sede.

Ejecución del instalador de SQL Server

Al observar la ilustración 1 presentada a continuación logramos observar que al ejecutar el centro de instalación de SQL Server debemos escoger una carpeta en la cual se realizara la creación de la instancia.

CONSEJO: Escoger la carpeta en la que se encuentren los archivos del programa.

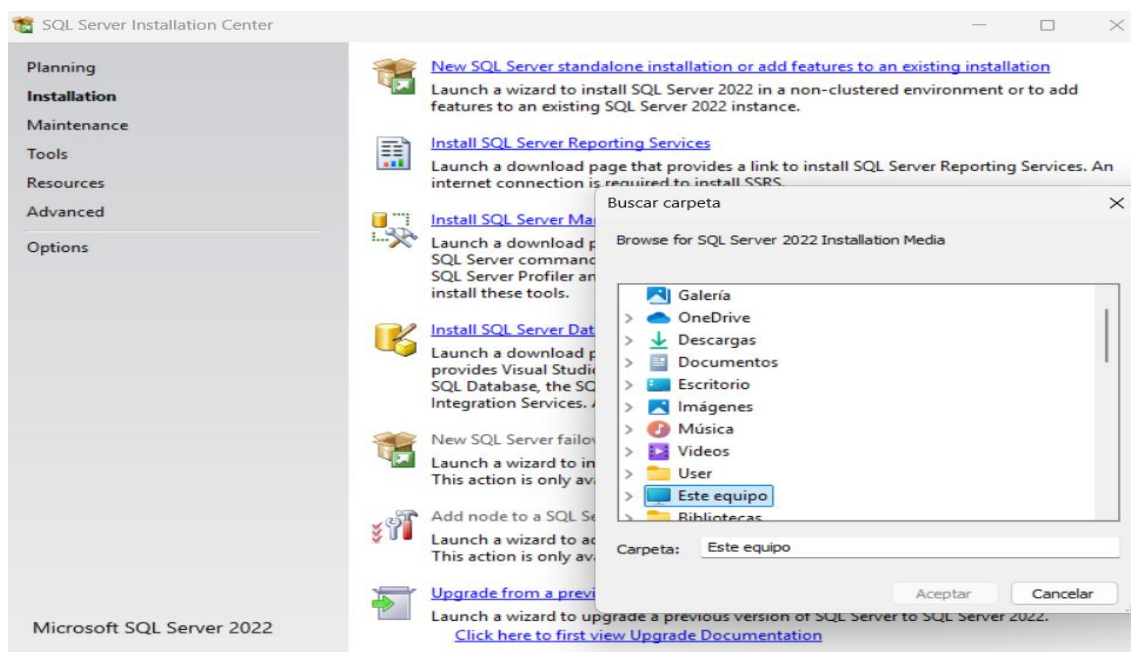


Ilustración 1 Selección de carpeta para la creación de la instancia



Configuración y Creación de la Instancia

Durante la configuración en el apartado de tipo de instalación se escoge la opción Realizar una nueva Instalación de SQL Server, como se puede observar en la ilustración 2. Esto permitirá realizar la creación de la instancia.



Ilustración 2 Tipo de instalación

En el apartado de **configuración de instancia** debemos seleccionar colocar un nombre propio a la nueva instancia y se colocara el nombre que se desee. En este caso SITIO_A, como se observa en la ilustración 3.

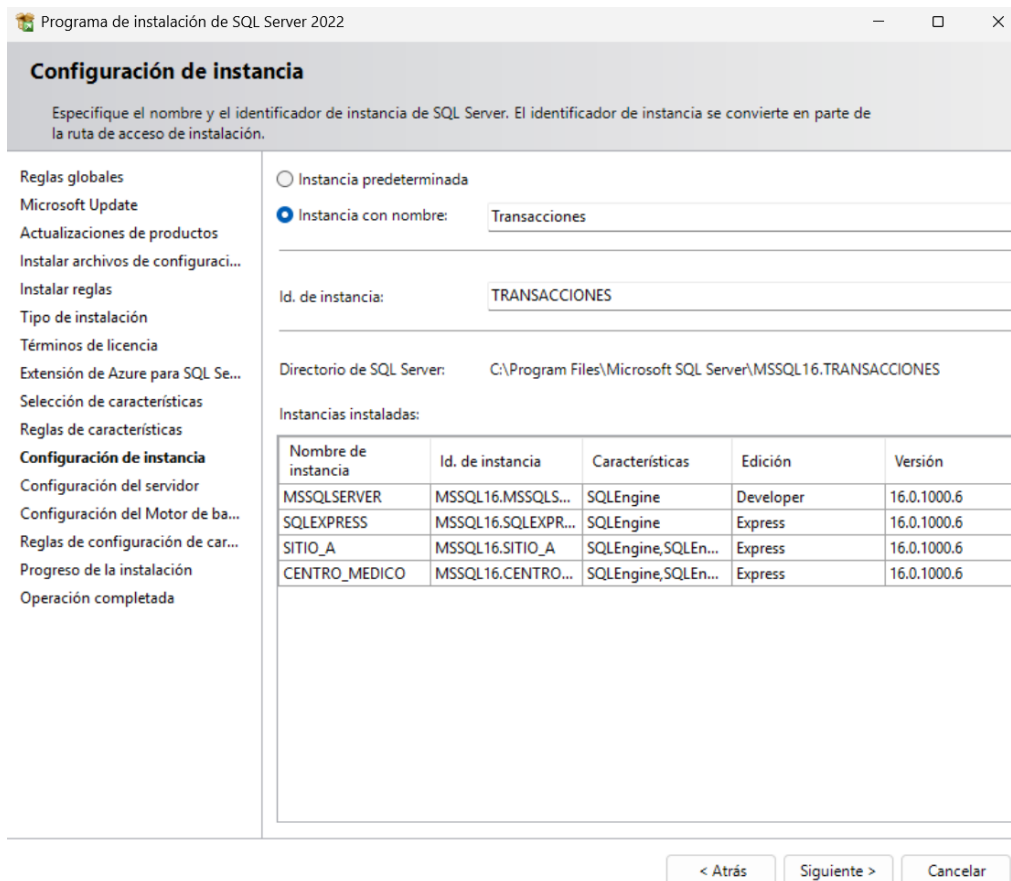


Ilustración 3 Asignar nombre a la nueva instancia



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE Elige un elemento.
CICLO ACADÉMICO: MARZO – JULIO 2025



Una vez configurado todos los apartados se iniciará el proceso de instalación como se observa en la ilustración 4.

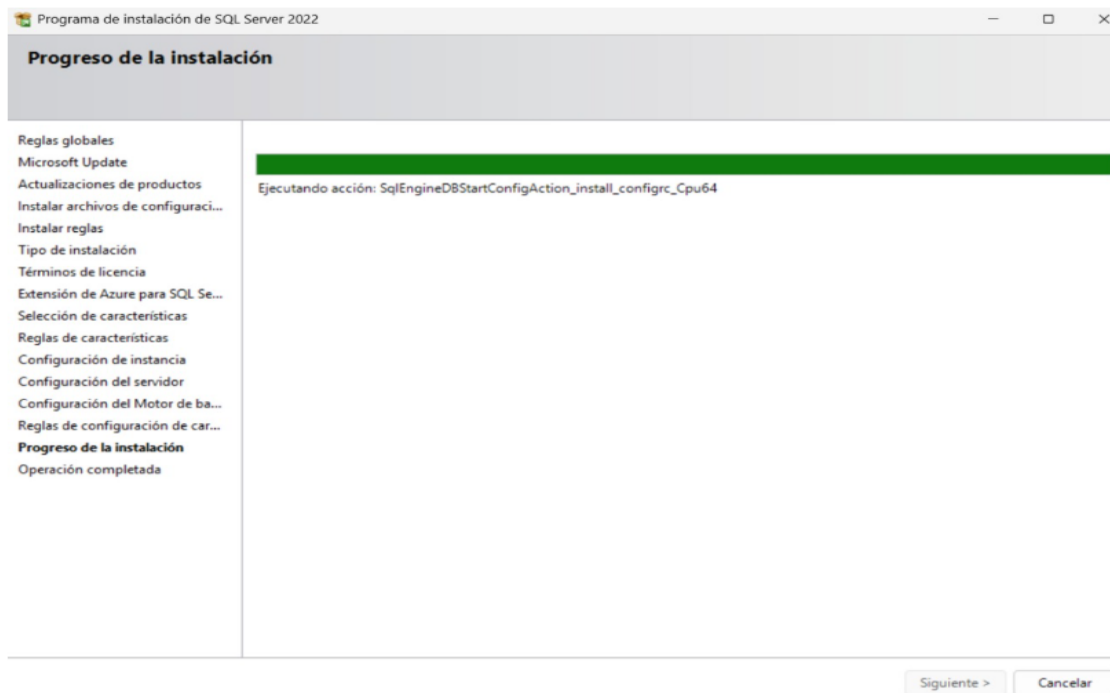


Ilustración 4 Progreso de instalación de la instancia

Como se puede observar en la Ilustración 5, la instancia ha sido creada correctamente y está lista para la posterior creación de la base de datos, esquemas y fragmentos que conformarán la vista global.

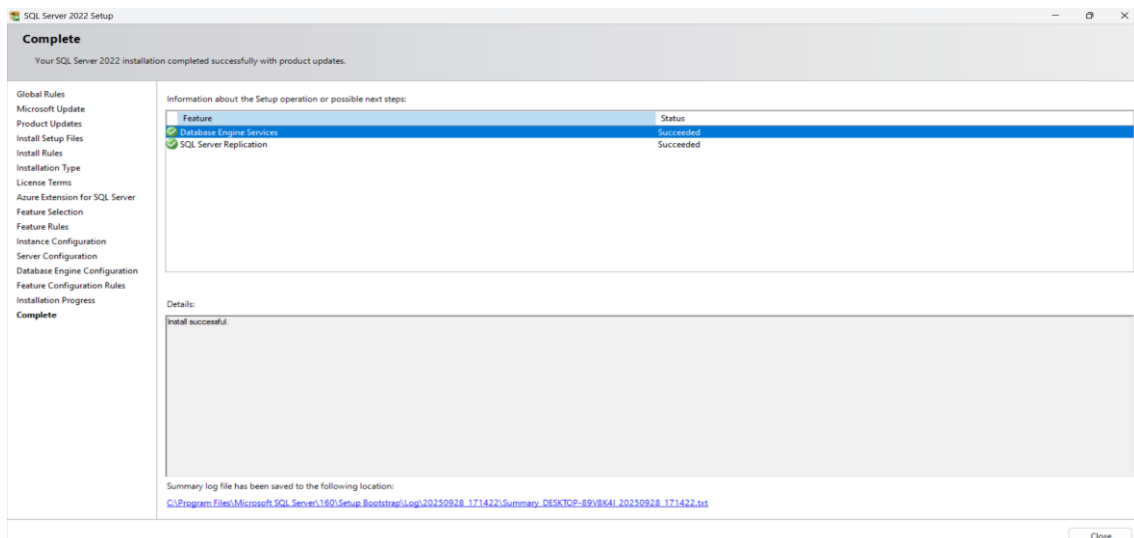


Ilustración 5 Creación Completa de la nueva Instancia

CONECTARSE A SQL SERVER MANAGEMENT STUDIO (SSMS).

¿Cómo ingresar a la instancia creada?

Para lograr ingresar en la instancia que creamos ejecutamos la aplicación de SQL Server y en la ventana en la cual debemos ingresar el nombre del servidor deberemos ingresar el nombre colocado a la instancia durante su creación, este nombre debe ir antecedido por .\



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE Elige un elemento.
CICLO ACADÉMICO: MARZO – JULIO 2025



Connect to Server

SQL Server

Login | Connection Properties | Always Encrypted | Additional Connection Parameters

Server

Server type: Database Engine

Server name: ERMENEL\TRANSACCIONES

Authentication: Windows Authentication

User name: ERMENEL\gatit

Password:

☐ Remember password

Connection Security

Encryption: Mandatory

☐ Trust server certificate

Host name in certificate:

Connect Cancel Help Options <<

Ilustración 6 Ingreso a la Instancia creada

DISEÑO DE LA BASE DE DATOS

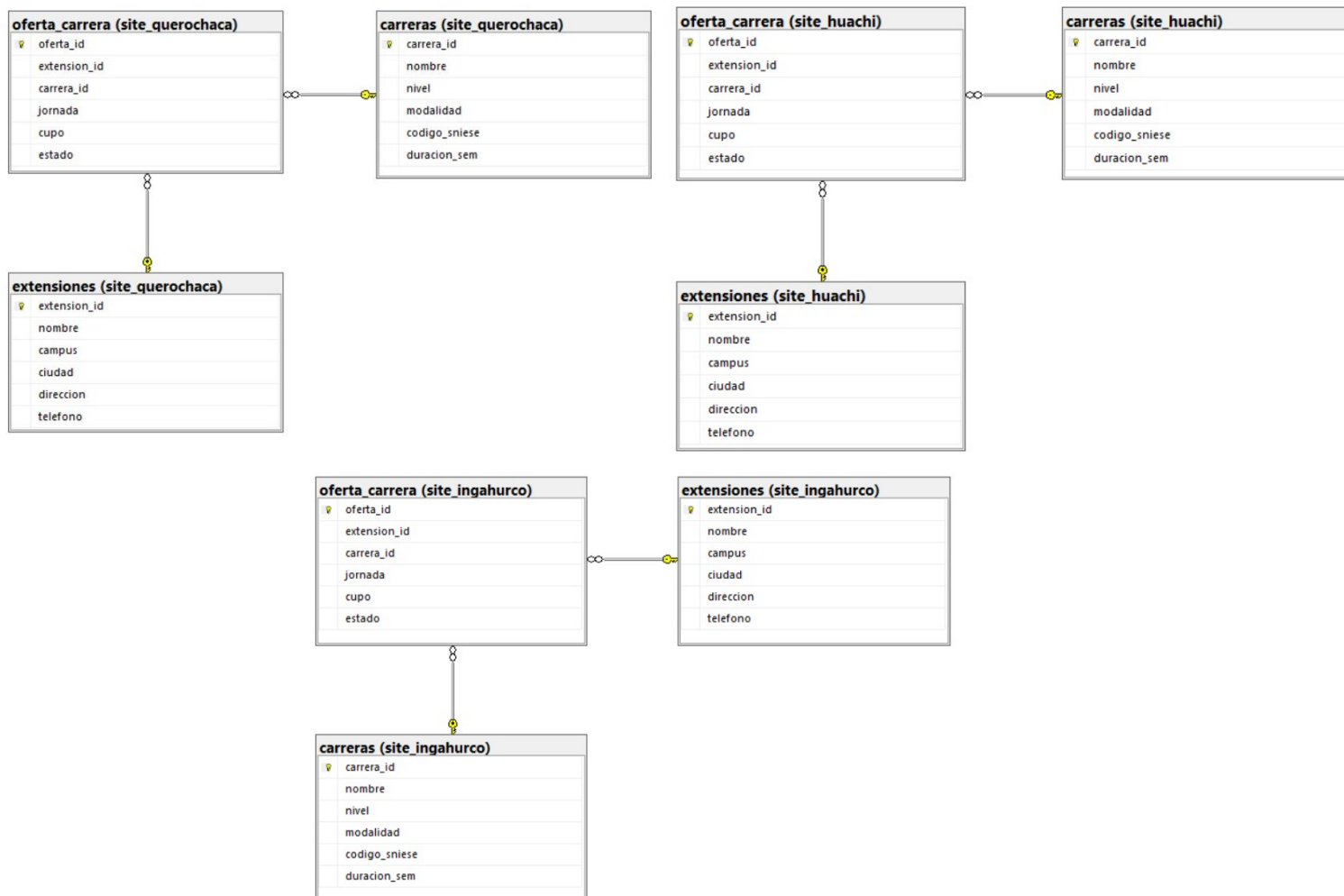


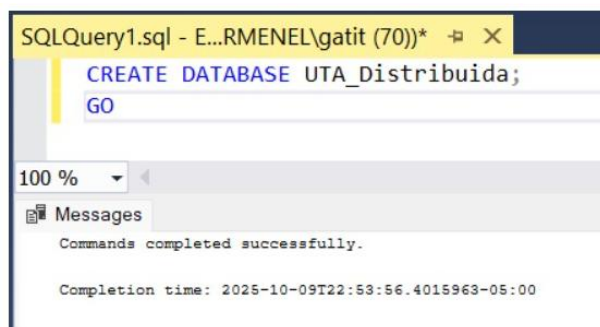
Ilustración 7 Diseño de la Base de Datos a Crear



IMPLEMENTACIÓN TÉCNICA (BD, esquemas, tablas, datos)

Una vez dentro de la instancia procederemos a la creación de nuestra base de datos.

Creación de la Base de Datos dentro de la Instancia:



```
SQLQuery1.sql - E...RMENEL\gatit (70))*
CREATE DATABASE UTA_Distribuida;
GO

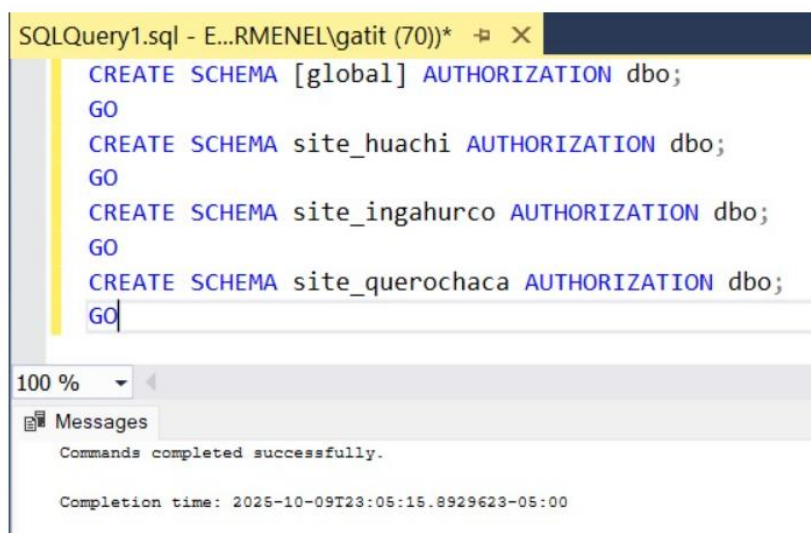
100 %
Messages
Commands completed successfully.
Completion time: 2025-10-09T22:53:56.4015963-05:00
```

Ilustración 8 Creación de la Base de Datos dentro de SQL SERVER

Creación de Esquemas dentro de la base de datos

La creación de esquemas sirve fundamentalmente para organizar y segmentar lógicamente la base de datos, lo cual es crucial para la gestión eficiente de la información en un entorno distribuido. Por ejemplo, al crear esquemas como site_huachi, site_ingahurco, y site_querochaca, se está estableciendo un espacio de nombres independiente para los datos asociados a ubicaciones o sitios específicos.

Esto permite que cada sitio maneje sus propias tablas, vistas y procedimientos almacenados de forma aislada, mejorando la seguridad (al limitar el acceso a los datos de un sitio específico), facilitando la administración (ya que los cambios en un sitio no afectan directamente a los demás)



```
SQLQuery1.sql - E...RMENEL\gatit (70))*
CREATE SCHEMA [global] AUTHORIZATION dbo;
GO
CREATE SCHEMA site_huachi AUTHORIZATION dbo;
GO
CREATE SCHEMA site_ingahurco AUTHORIZATION dbo;
GO
CREATE SCHEMA site_querochaca AUTHORIZATION dbo;
GO

100 %
Messages
Commands completed successfully.
Completion time: 2025-10-09T23:05:15.8929623-05:00
```

Ilustración 9 Creación de Esquemas dentro de la Base

Creación de Tablas:

Una vez creada y puesto en uso la base de datos procedemos con la creación de todas las tablas necesarias:

Creación de tabla site_huachi.carreras



```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
CREATE TABLE site huachi.carreras (
    carrera_id INT NOT NULL PRIMARY KEY,
    nombre VARCHAR(100) NOT NULL,
    nivel VARCHAR(30) NOT NULL,
    modalidad VARCHAR(30) NOT NULL,
    codigo_sniese VARCHAR(20) NULL,
    duracion_sem TINYINT NOT NULL
);
GO

100 %
Messages
Commands completed successfully.
Completion time: 2025-10-09T23:06:00.3788785-05:00
```

Ilustración 10 Creación de tabla site_huachi.carreras

Creación de tabla site_ingahurco.carreras

```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
CREATE TABLE site ingahurco.carreras (
    carrera_id INT NOT NULL PRIMARY KEY,
    nombre VARCHAR(100) NOT NULL,
    nivel VARCHAR(30) NOT NULL,
    modalidad VARCHAR(30) NOT NULL,
    codigo_sniese VARCHAR(20) NULL,
    duracion_sem TINYINT NOT NULL
);
GO

100 %
Messages
Commands completed successfully.
Completion time: 2025-10-09T23:06:40.4000837-05:00
```

Ilustración 11 Creación de tabla site_ingahurco.carreras

Creación de tabla site_querochacas.carreras

```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
CREATE TABLE site querochaca.carreras (
    carrera_id INT NOT NULL PRIMARY KEY,
    nombre VARCHAR(100) NOT NULL,
    nivel VARCHAR(30) NOT NULL,
    modalidad VARCHAR(30) NOT NULL,
    codigo_sniese VARCHAR(20) NULL,
    duracion_sem TINYINT NOT NULL
);
GO

100 %
Messages
Commands completed successfully.
Completion time: 2025-10-09T23:07:33.5914537-05:00
```

Ilustración 12 Creación de tabla site_querochacas.carreras



Creación de tabla site_huachi.extensiones

```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
CREATE TABLE site_huachi.extensiones (
    extension_id INT NOT NULL PRIMARY KEY,
    nombre VARCHAR(120) NOT NULL,
    campus VARCHAR(40) NOT NULL,
    ciudad VARCHAR(60) NOT NULL,
    direccion VARCHAR(160) NULL,
    telefono VARCHAR(30) NULL
);
GO

100 %
Messages
Commands completed successfully.
Completion time: 2025-10-09T23:09:28.3894040-05:00
```

Ilustración 13 Creación de tabla site_huachi.extensiones

Creación de tabla site_ingahurco.extensiones

```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
CREATE TABLE site_ingahurco.extensiones (
    extension_id INT NOT NULL PRIMARY KEY,
    nombre VARCHAR(120) NOT NULL,
    campus VARCHAR(40) NOT NULL,
    ciudad VARCHAR(60) NOT NULL,
    direccion VARCHAR(160) NULL,
    telefono VARCHAR(30) NULL
);
GO

100 %
Messages
Commands completed successfully.
Completion time: 2025-10-09T23:10:21.1562769-05:00
```

Ilustración 14 Creación de tabla site_ingahurco.extensiones

Creación de la tabla site_querachaca.extensiones

```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
CREATE TABLE site_querachaca.extensiones (
    extension_id INT NOT NULL PRIMARY KEY,
    nombre VARCHAR(120) NOT NULL,
    campus VARCHAR(40) NOT NULL,
    ciudad VARCHAR(60) NOT NULL,
    direccion VARCHAR(160) NULL,
    telefono VARCHAR(30) NULL
);
GO

100 %
Messages
Commands completed successfully.
Completion time: 2025-10-09T23:11:01.2178861-05:00
```

Ilustración 15 Creación de la tabla site_querachaca.extensiones



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE Elige un elemento.
CICLO ACADÉMICO: MARZO – JULIO 2025



Creación de tabla site_huachi.oferta_carrera

```
SQLQuery1.sql - E...RMENEL\gatit (70))* - X
CREATE TABLE site_huachi.oferta_carrera (
    oferta_id INT NOT NULL PRIMARY KEY,
    extension_id INT NOT NULL,
    carrera_id INT NOT NULL,
    jornada VARCHAR(20) NOT NULL,
    cupo INT NOT NULL,
    estado VARCHAR(15) NOT NULL,
    CONSTRAINT FK_ofh_ext FOREIGN KEY (extension_id) REFERENCES site_huachi.extensiones(extension_id),
    CONSTRAINT FK_ofh_car FOREIGN KEY (carrera_id) REFERENCES site_huachi.carreras(carrera_id)
);
GO

100 %
Messages
Commands completed successfully.
Completion time: 2025-10-09T23:12:40.6851181-05:00
```

Ilustración 16 Creación de tabla site_huachi.oferta_carrera

Creación de tabla site_ingahurco.oferta_carrera

```
SQLQuery1.sql - E...RMENEL\gatit (70))* - X
CREATE TABLE site_ingahurco.oferta_carrera (
    oferta_id INT NOT NULL PRIMARY KEY,
    extension_id INT NOT NULL,
    carrera_id INT NOT NULL,
    jornada VARCHAR(20) NOT NULL,
    cupo INT NOT NULL,
    estado VARCHAR(15) NOT NULL,
    CONSTRAINT FK_ofi_ext FOREIGN KEY (extension_id) REFERENCES site_ingahurco.extensiones(extension_id),
    CONSTRAINT FK_ofi_car FOREIGN KEY (carrera_id) REFERENCES site_ingahurco.carreras(carrera_id)
);
GO

100 %
Messages
Commands completed successfully.
Completion time: 2025-10-09T23:13:16.4470244-05:00
```

Ilustración 17 Creación de tabla site_ingahurco.oferta_carrera

Creación de tabla site_querchaca.oferta_carrera

```
SQLQuery1.sql - E...RMENEL\gatit (70))* - X
CREATE TABLE site_querchaca.oferta_carrera (
    oferta_id INT NOT NULL PRIMARY KEY,
    extension_id INT NOT NULL,
    carrera_id INT NOT NULL,
    jornada VARCHAR(20) NOT NULL,
    cupo INT NOT NULL,
    estado VARCHAR(15) NOT NULL,
    CONSTRAINT FK_ofq_ext FOREIGN KEY (extension_id) REFERENCES site_querchaca.extensiones(extension_id),
    CONSTRAINT FK_ofq_car FOREIGN KEY (carrera_id) REFERENCES site_querchaca.carreras(carrera_id)
);
GO

100 %
Messages
Commands completed successfully.
Completion time: 2025-10-09T23:13:52.4078204-05:00
```

Ilustración 18 Creación de tabla site_querchaca.oferta_carrera

Verificación de la correcta creación de tablas:

Es una muy buena práctica el revisar que se realizaron correctamente las acciones posteriores a ejecutar cualquier comando, esto evitara errores o confusiones que puedan llegar a ocurrir durante la práctica.



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE Elige un elemento.
CICLO ACADÉMICO: MARZO – JULIO 2025

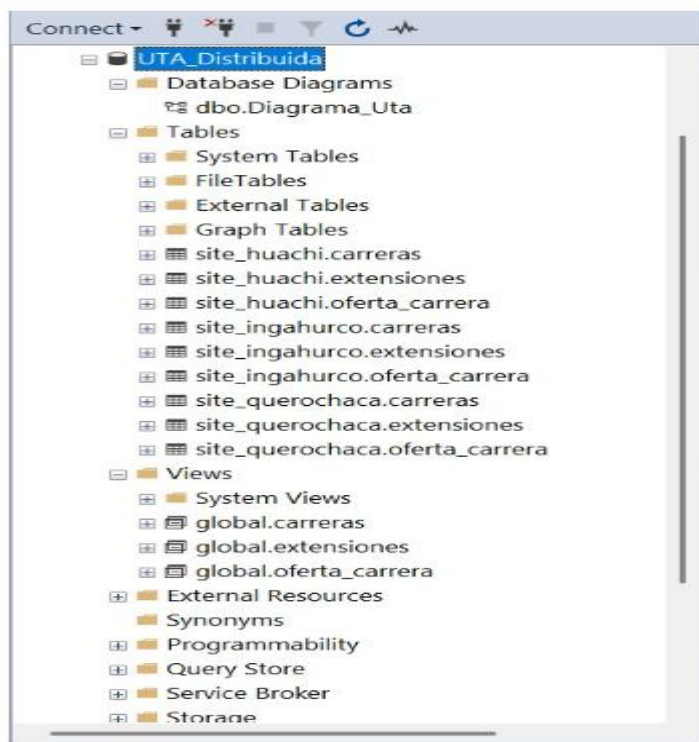


Ilustración 19 Verificación de la correcta creación de tablas

Verificación diagrama de base de datos obtenido:

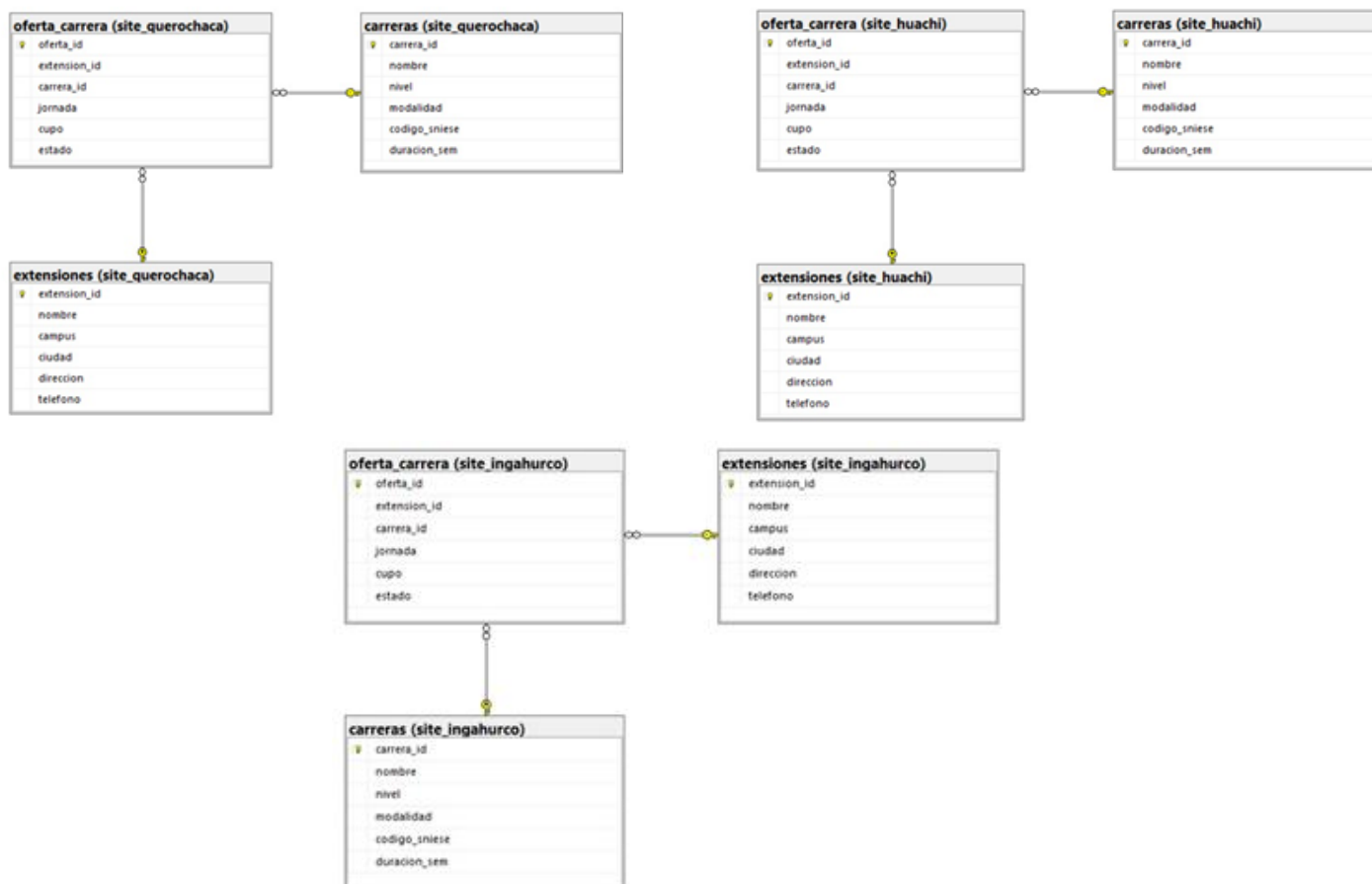


Ilustración 20 Verificación diagrama de base de datos obtenido

INSERCIÓN DE DATOS PARA POSTERIORES PRUEBAS



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE Elige un elemento.
CICLO ACADÉMICO: MARZO – JULIO 2025



Este paso es esencial ya que nos permitirá verificar la integridad de nuestra base de datos mediante la inserción de datos correctos o incorrectos (intentando violar la integridad referencial) posteriormente.

```
SQLQuery1.sql - E...RMENEL\gatit (70))* - X
INSERT INTO site_huachi.carreras VALUES
(1,'Ingeniería en Sistemas','Pregrado','Presencial','UTA-IS',10),
(2,'Ingeniería Civil','Pregrado','Presencial','UTA-IC',10),
(3,'Administración de Empresas','Pregrado','Híbrida','UTA-AE',8),
(4,'Enfermería','Pregrado','Presencial','UTA-ENF',10);
INSERT INTO site_ingahurco.carreras SELECT * FROM site_huachi.carreras;
INSERT INTO site_querchaca.carreras SELECT * FROM site_huachi.carreras;
GO

100 %
Messages
(4 rows affected)
(4 rows affected)
(4 rows affected)
Completion time: 2025-10-09T23:08:32.9553451-05:00
```

Ilustración 21 Inserción de Datos

```
SQLQuery1.sql - E...RMENEL\gatit (70))* - X
INSERT INTO site_huachi.extensiones VALUES
(101,'Universidad Técnica de Ambato - Campus Huachi','Huachi','Ambato','Av. Los Chasquis s/n','(03) 299-0001');
INSERT INTO site_ingahurco.extensiones VALUES
(201,'Universidad Técnica de Ambato - Campus Ingahurco','Ingahurco','Ambato','Av. Colombia y Chile','(03) 299-0002');
INSERT INTO site_querchaca.extensiones VALUES
(301,'Universidad Técnica de Ambato - Campus Querchaca','Querchaca','Cevallos','Vía a Quero km 2','(03) 299-0003');
GO

100 %
Messages
(1 row affected)
(1 row affected)
(1 row affected)
Completion time: 2025-10-09T23:11:41.1649116-05:00
```

Ilustración 22 Inserción de Datos

```
SQLQuery1.sql - E...RMENEL\gatit (70))* - X
INSERT INTO site_huachi.oferta_carrera VALUES
(1101,101,1,'Matutina',60,'Activa'),
(1102,101,3,'Nocturna',40,'Activa');
INSERT INTO site_ingahurco.oferta_carrera VALUES
(2101,201,2,'Matutina',50,'Activa'),
(2102,201,1,'Vespertina',45,'Activa');
INSERT INTO site_querchaca.oferta_carrera VALUES
(3101,301,4,'Vespertina',35,'Activa'),
(3102,301,3,'Nocturna',30,'Activa');
GO

100 %
Messages
(2 rows affected)
(2 rows affected)
(2 rows affected)
Completion time: 2025-10-09T23:14:38.5944623-05:00
```

Ilustración 23 Inserción de Datos



CREACIÓN DE VISTAS

Ayudan a verificar los datos de manera mucha mas rápida y de ser necesario brinda la información filtrada mediante parámetros especificados antes de su ejecución.

```
SQLQuery1.sql - E...RMENEL\gatit (70))*  X
CREATE VIEW global.extensiones AS
SELECT * FROM site_huachi.extensiones
UNION ALL
SELECT * FROM site_ingahurco.extensiones
UNION ALL
SELECT * FROM site_querchaca.extensiones;
GO
```

100 %

Messages

Commands completed successfully.

Completion time: 2025-10-09T23:15:23.1496341-05:00

Ilustración 24 Creación de Vista a las extensiones

```
SQLQuery1.sql - E...RMENEL\gatit (70))*  X
CREATE VIEW global.oferta_carrera AS
SELECT * FROM site_huachi.oferta_carrera
UNION ALL
SELECT * FROM site_ingahurco.oferta_carrera
UNION ALL
SELECT * FROM site_querchaca.oferta_carrera;
GO
```

100 %

Messages

Commands completed successfully.

Completion time: 2025-10-09T23:15:51.0196086-05:00

Ilustración 25 Creación de vista a oferta carreras

```
SQLQuery1.sql - E...RMENEL\gatit (70))*  X
CREATE VIEW global.carreras AS
SELECT * FROM site_huachi.carreras;
GO
```

100 %

Messages

Commands completed successfully.

Completion time: 2025-10-09T23:16:31.2607661-05:00

Ilustración 26 Creación de vista a carreras



VERIFICACIÓN DE ESQUEMAS

The screenshot shows a SQL query window titled "SQLQuery1.sql - E...RMENEL\gatit (70))*". The query is: `SELECT SCHEMA_NAME FROM INFORMATION_SCHEMA.SCHEMATA; GO`. The results pane shows a list of 17 schemas:

	SCHEMA_NAME
1	dbo
2	guest
3	INFORMATION_SCHEMA
4	sys
5	global
6	site_huachi
7	site_ingahurco
8	site_querochaca
9	db_owner
10	db_accessadmin
11	db_securityadmin
12	db_ddladmin
13	db_backupoperator
14	db_datareader
15	db_datawriter
16	db_denydatareader
17	db_denydatawriter

Ilustración 27 VERIFICACIÓN DE ESQUEMAS

VERIFICACIÓN DE CONSULTAS EN LAS TABLAS

The screenshot shows a SQL query window titled "SQLQuery1.sql - E...RMENEL\gatit (70))*". The query is: `SELECT TABLE_SCHEMA, TABLE_NAME FROM INFORMATION_SCHEMA.TABLES WHERE TABLE_TYPE = 'BASE TABLE'; GO`. The results pane shows a list of 9 tables:

	TABLE_SCHEMA	TABLE_NAME
1	site_huachi	carreras
2	site_ingahurco	carreras
3	site_querochaca	carreras
4	site_huachi	extensiones
5	site_ingahurco	extensiones
6	site_querochaca	extensiones
7	site_huachi	oferta_carrera
8	site_ingahurco	oferta_carrera
9	site_querochaca	oferta_carrera

Ilustración 28 VERIFICACIÓN DE CONSULTAS EN LAS TABLAS



VERIFICACIÓN DE LA INTEGRIDAD REFERENCIAL

```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
SELECT * FROM site_huachi.carreras;
SELECT * FROM site_huachi.extensiones;

INSERT INTO site_huachi.oferta_carrera VALUES
(1201,101,2,'Matutina',50,'Activa');

INSERT INTO site_huachi.oferta_carrera VALUES
(1202,199,3,'Nocturna',40,'Activa');
GO
```

100 %

Results Messages

(4 rows affected)
(1 row affected)
(1 row affected)
Msg 547, Level 16, State 9, Line 7
Instrucción INSERT en conflicto con la restricción FOREIGN KEY 'FK_ofh_ext'. El conflicto ha aparecido en la base de datos 'UTA_Distribuida', tabla 'site_huachi.extensiones', column 'extension_id'
Se terminó la instrucción.
Completion time: 2025-10-09T23:44:52.7918100-05:00

Ilustración 29 VERIFICACIÓN DE LA INTEGRIDAD REFERENCIAL

Como se logra observar en la ilustración 29 al intentar violar la integridad referencial la ejecución del comando no se realiza y nos alerta de que el comando ejecutado es incorrecto.

TRANSACCIONES: COMMIT Y ROLLBACK

Las transacciones en bases de datos se tratan de un conjunto de operaciones que se ejecutan como una sola unidad lógica de trabajo. Se realizan para asegurar que los datos permanezcan consistentes y confiables, incluso si ocurre un error durante el proceso.

Dentro de ellas, los comandos más conocidos son **COMMIT** y **ROLLBACK**, debido a que permiten controlar el resultado final de una transacción.

Ejemplo haciendo uso de COMMIT:

```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
BEGIN TRANSACTION;

INSERT INTO site_ingahurco.oferta_carrera VALUES
(2203,201,3,'Matutina',55,'Activa');

COMMIT;
GO
```

100 %

Messages

(1 row affected)

Completion time: 2025-10-09T23:45:50.6041016-05:00

Ilustración 30 Transacción usando COMMIT



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE Elige un elemento.
CICLO ACADÉMICO: MARZO – JULIO 2025



Ejemplo haciendo uso de ROLLBACK:

```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
BEGIN TRANSACTION;
INSERT INTO site_querchaca.oferta_carrera VALUES
(3203,399,2,'Vespertina',30,'Activa');
ROLLBACK;
GO
```

100 %

Messages

Msg 845, Level 16, State 0, Line 3
Instrucción INSERT en conflicto con la restricción FOREIGN KEY 'FK_ofg_ext'. El conflicto ha aparecido en la base de datos 'UTA_Distribuida', tabla 'site_querchaca.extensiones', column 'extension'. Se terminó la instrucción.

Completion time: 2025-10-09T23:46:24.3595913-05:00

Ilustración 31 Transacción usando ROLLBACK

VERIFICACIÓN

SQLQuery1.sql - E...RMENEL\gatit (70))* X

```
SELECT * FROM site_querchaca.oferta_carrera;
GO
```

100 %

Results Messages

	oferta_id	extension_id	carrera_id	jornada	cupo	estado
1	3101	301	4	Vespertina	35	Activa
2	3102	301	3	Nocturna	30	Activa

Ilustración 32 Verificación de los ejemplos mostrados

PRUEBAS DE VISTAS GLOBALES CON TRANSPARENCIA DE ACCESO

SQLQuery1.sql - E...RMENEL\gatit (70))* X

```
SELECT * FROM global.carreras;
SELECT * FROM global.extensiones;
SELECT * FROM global.oferta_carrera;
GO
```

100 %

Results Messages

	carrera_id	nombre	nivel	modalidad	codigo_sniese	duracion_sem
1	1	Ingeniería en Sistemas	Pregrado	Presencial	UTA-IS	10
2	2	Ingeniería Civil	Pregrado	Presencial	UTA-IC	10
3	3	Administración de Empresas	Pregrado	Híbrida	UTA-AE	8
4	4	Enfermería	Pregrado	Presencial	UTA-ENF	10

	extension_id	nombre	campus	ciudad	direccion	telefono
1	101	Universidad Técnica de Ambato - Campus Huachi	Huachi	Ambato	Av. Los Chasquis s/n	(03) 299-0001
2	201	Universidad Técnica de Ambato - Campus Ingahurco	Ingahurco	Ambato	Av. Colombia y Chile	(03) 299-0002
3	301	Universidad Técnica de Ambato - Campus Querchaca	Querchaca	Cevallos	Via a Quero km 2	(03) 299-0003

	oferta_id	extension_id	carrera_id	jornada	cupo	estado
1	1101	101	1	Matutina	60	Activa
2	1102	101	3	Nocturna	40	Activa
3	1201	101	2	Matutina	50	Activa
4	2101	201	2	Matutina	50	Activa
5	2102	201	1	Vespertina	45	Activa
6	2203	201	3	Matutina	55	Activa
7	3101	301	4	Vespertina	35	Activa
8	3102	301	3	Nocturna	30	Activa

Ilustración 33 PRUEBAS DE VISTAS GLOBALES CON TRANSPARENCIA DE ACCESO



PRUEBAS EN SESIONES CONCURRENTES

Las pruebas en sesiones se realizan para observar cómo se comporta la base de datos cuando dos o más usuarios trabajan al mismo tiempo. Este tipo de pruebas permite analizar la concurrencia y los posibles bloqueos que pueden generarse cuando varias transacciones intentan acceder o modificar los mismos datos.

Sesión 1 (Actualización):

```
SQLQuery1.sql - E...RMENEL\gatit (70))*
BEGIN TRANSACTION;
UPDATE site_huachi.oferta_carrera
SET cupo = cupo + 10
WHERE oferta_id = 1101;
GO
```

100 %

Messages

(1 row affected)

Completion time: 2025-10-09T23:48:48.7356582-05:00

Ilustración 34 Prueba Sesión 1 (Actualización)

Sesión 2 (Lectura):

```
SQLQuery2.sql - ERM...(55) Executing...*
SELECT * FROM site_huachi.oferta_carrera WHERE oferta_id = 1101;
GO
```

100 %

Results Messages

Ilustración 35 Prueba Sesión 2 (Lectura)

MANEJO DE ERRORES

El manejo de errores en las transacciones se realiza para prevenir daños o inconsistencias en la base de datos cuando ocurre una falla durante la ejecución de las operaciones. En SQL Server existen varias formas de controlar estos errores, siendo las más utilizadas TRY...CATCH, SET XACT_ABORT, y ON_ERROR. Estas herramientas permiten detectar automáticamente los fallos y decidir si la transacción debe continuar o revertirse mediante un ROLLBACK.

Ejemplo haciendo uso de ON_ERROR y XACT_ABORT:

Los comandos ON_ERROR y SET XACT_ABORT se utilizan para controlar el comportamiento de las transacciones cuando ocurre un error. Con ON_ERROR se puede definir qué acción debe ejecutar SQL Server ante una falla, mientras que SET XACT_ABORT ON hace que la transacción se cancele automáticamente si se produce



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE Elige un elemento.
CICLO ACADÉMICO: MARZO – JULIO 2025



cualquier error durante su ejecución. Esto asegura que la base de datos no quede en un estado inconsistente y que todos los cambios se reviertan de manera inmediata.

```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
SET XACT_ABORT ON;

BEGIN TRANSACTION;
INSERT INTO site_huachi.oferta_carrera VALUES
(1301,999,1,'Nocturna',40,'Activa'); -- Error FK
COMMIT;
GO
```

100 %

Messages

Msg 547, Level 16, State 0, Line 4
Instrucción INSERT en conflicto con la restricción FOREIGN KEY 'FK_ofc_ext'. El conflicto ha aparecido en la base de datos 'UTA_Distribuida', tabla 'site_huachi.extensiones', column 'extension_id'.
Completion time: 2025-10-09T23:51:20.9205446-05:00

Ilustración 36 Ejemplo haciendo uso de ON_ERROR y XACT_ABORT

Ejemplo haciendo uso de **TRY** y **CATCH**:

La estructura TRY...CATCH se utiliza para manejar errores de forma controlada dentro de una transacción. En el bloque TRY se colocan las instrucciones que podrían generar un error, y si ocurre alguno, el control pasa automáticamente al bloque CATCH, donde se pueden realizar acciones como mostrar un mensaje o ejecutar un ROLLBACK.

```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
BEGIN TRY
    BEGIN TRANSACTION;

    INSERT INTO site_ingahurco.oferta_carrera VALUES
    (2305,999,2,'Matutina',45,'Activa'); -- Error FK

    COMMIT;
END TRY
BEGIN CATCH
    PRINT 'Error detectado: ' + ERROR_MESSAGE();
    ROLLBACK;
END CATCH;
GO
```

100 %

Messages

(0 rows affected)
Error detectado: Instrucción INSERT en conflicto con la restricción FOREIGN KEY 'FK_ofi_ext'. El conflicto ha aparecido en la base de datos 'UTA_Distribuida', tabla 'site_ingahurco.extensiones', column 'extension_id'.
Completion time: 2025-10-09T23:52:04.0744344-05:00

Ilustración 37 Ejemplo haciendo uso de TRY y CATCH

CONFIRMACIÓN DE RESULTADOS FINALES

```
SQLQuery1.sql - E...RMENEL\gatit (70))* X
SELECT * FROM global.oferta_carrera;
GO
```

100 %

Results Messages

	oferta_id	extension_id	carrera_id	jornada	cupo	estado
1	1101	101	1	Matutina	60	Activa
2	1102	101	3	Nocturna	40	Activa
3	1201	101	2	Matutina	50	Activa
4	2101	201	2	Matutina	50	Activa
5	2102	201	1	Vespertina	45	Activa
6	2203	201	3	Matutina	55	Activa
7	3101	301	4	Vespertina	35	Activa
8	3102	301	3	Nocturna	30	Activa

Ilustración 38 CONFIRMACIÓN DE RESULTADOS FINALES



2.7 Resultados obtenidos

Un sistema transaccional siempre finaliza con Commit o Rollback, caso contrario puede generar un estado inconsistente de la BD. Para el manejo de errores existen varias alternativas que funciona de acuerdo al contexto.

Durante el desarrollo de la práctica se comprobó que las transacciones en SQL Server garantizan la integridad y coherencia de los datos al permitir que las operaciones se ejecuten completamente o se reviertan en caso de error. Mediante los comandos COMMIT y ROLLBACK se evidenció el control que ofrece el sistema sobre los cambios realizados, asegurando que la base de datos mantenga un estado consistente.

Además, al realizar pruebas en sesiones concurrentes se observó cómo SQL Server maneja los bloqueos y la concurrencia entre usuarios, evitando conflictos en la modificación de datos. También, el uso de TRY...CATCH, ON_ERROR y SET XACT_ABORT permitió manejar errores de manera eficaz, mostrando cómo el motor puede abortar o revertir transacciones automáticamente ante fallos, reforzando la confiabilidad del sistema transaccional.

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

Durante el desarrollo de la práctica se aplicaron diversas habilidades blandas que contribuyeron al correcto manejo y comprensión de las transacciones en la base de datos.

El pensamiento crítico fue esencial para analizar el comportamiento del sistema ante los comandos COMMIT, ROLLBACK y los diferentes métodos de control de errores, permitiendo interpretar los resultados y comprender cómo cada acción afecta la integridad de los datos.

La resolución de conflictos se aplicó al solucionar errores de integridad referencial y bloqueos generados en las sesiones concurrentes, logrando mantener la consistencia de la información sin afectar el trabajo realizado.

La adaptabilidad se demostró al enfrentarse a imprevistos técnicos y ajustes solicitados durante las pruebas de transacciones y manejo de errores, adaptando las estrategias según las necesidades del ejercicio.

La responsabilidad se evidenció en la ejecución ordenada de cada paso de la guía, en la revisión de resultados obtenidos y en el compromiso por mantener la base de datos libre de inconsistencias.

Además, un pilar esencial durante la ejecución de la guía práctica fue el **trabajo en equipo** lo cual ayudo a que la ejecución de las actividades realizadas dentro de la misma fuera mas interactivas entre los miembros del equipo además de reducir la carga de trabajo en un solo



UNIVERSIDAD TÉCNICA DE AMBATO

FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL

CARRERA DE Elige un elemento.

CICLO ACADÉMICO: MARZO – JULIO 2025



integrante y lo mas importa el realizar reuniones y dialogar sobre las actividades realizadas, los problemas encontrados y los resultados obtenidos.

Todo el trabajo en equipo fue distribuido, administrado y dado seguimiento mediante AzureDevops, esta herramienta permitió un gran control sobre las actividades a realizar y además la correcta y clara distribución a cada miembro del equipo.

Resumen de la Práctica Realizada

GUÍA-APE-2---Tratamiento-de-tr...
Enter page title
Guía Ape Número 2 - Tratami...
+ New page

Guía Ape Número 2 - Tratamiento de transacciones

Gomez Llerena Luis Fernando · 21m ago

Qué actividades se realizarán durante la ejecución de la Práctica?

En esta práctica se va a realizar la implementación y análisis del comportamiento de las transacciones en un Sistema Gestor de Base de Datos (SGBD) utilizando Microsoft SQL Server. El objetivo principal es comprobar cómo las transacciones garantizan la integridad y consistencia de los datos dentro de una base de datos relacional.

Para el desarrollo de la práctica se creará una nueva instancia de SQL Server destinada exclusivamente a las pruebas, dentro de la cual se diseñará una base de datos llamada "Universidad". En esta base se implementarán esquemas distribuidos (site_huachi, site_ingahurco y site_querachaca), representando distintas sedes de la institución, y se crearán las tablas necesarias para simular un entorno real de gestión académica.

Posteriormente, se insertarán datos de prueba para verificar la integridad referencial y se crearán vistas que permitirán consultar la información de manera más rápida y organizada. A continuación, se aplicarán transacciones con los comandos BEGIN TRANSACTION, COMMIT y ROLLBACK, con el fin de evidenciar el principio de atomicidad, donde las operaciones se ejecutan completamente o se revierten en caso de error.

También se realizarán pruebas de concurrencia con múltiples sesiones conectadas al mismo tiempo para observar el manejo de bloqueos, niveles de aislamiento y consistencia de los datos cuando varios usuarios acceden simultáneamente al sistema.

Finalmente, se implementarán técnicas de manejo de errores como SET XACT_ABORT, ON_ERROR y la estructura TRY...CATCH, con el propósito de controlar los fallos que puedan presentarse durante la ejecución de las transacciones y asegurar que la base de datos mantenga un estado coherente.

Durante toda la práctica se hará uso de SQL Server Management Studio (SSMS), computadoras personales y herramientas TAC como plataformas educativas y apoyo de inteligencia artificial para la interpretación de resultados y generación de código.

Asignación e Identificación de Tareas a Realizar

GUÍA APE 2 - Tratamie...
Overview
Boards
Work items
Boards
Backlogs
Sprints
Queries
Delivery Plans
Analytics views
Repos

Work items

Recently updated · + New Work Item · Open in Queries · Column Options · Import Work Items · Recycle Bin

Filter by keyword

ID	Title	Assigned To	State	Area Path	Tags
14	✓ Pruebas avanzadas y análisis	Paredes Barrera Luis Enrique	To Do	GUÍA APE 2 - Tratamiento de t...	
13	✓ Transacciones básicas (COMMIT y ROLLBACK)	Gomez Llerena Luis Fernan...	To Do	GUÍA APE 2 - Tratamiento de t...	
12	✓ Inserción de datos, creación de vistas y pruebas de integridad referen	Gomez Llerena Luis Fernan...	To Do	GUÍA APE 2 - Tratamiento de t...	
11	✓ Creación de tablas y verificación	Caguasango Bayas Alex Pa...	To Do	GUÍA APE 2 - Tratamiento de t...	
10	✓ Diseño y creación de la base de datos y sus esquemas	Caguasango Bayas Alex Pa...	To Do	GUÍA APE 2 - Tratamiento de t...	
1	✓ Creación de la instancia y conexión en SQL Server	Aldas Jordan Wellington Is...	To Do	GUÍA APE 2 - Tratamiento de t...	

Repartición de Tareas y Seguimiento de las Mismas

GUÍA APE 2 - Tratamiento de transacciones Team · View as backlog

Board Analytics

To Do

Doing 0/5

Done

9 Creación de la instancia y conexión en SQL Server
Done
Aldas Jordan Wellington Ismael

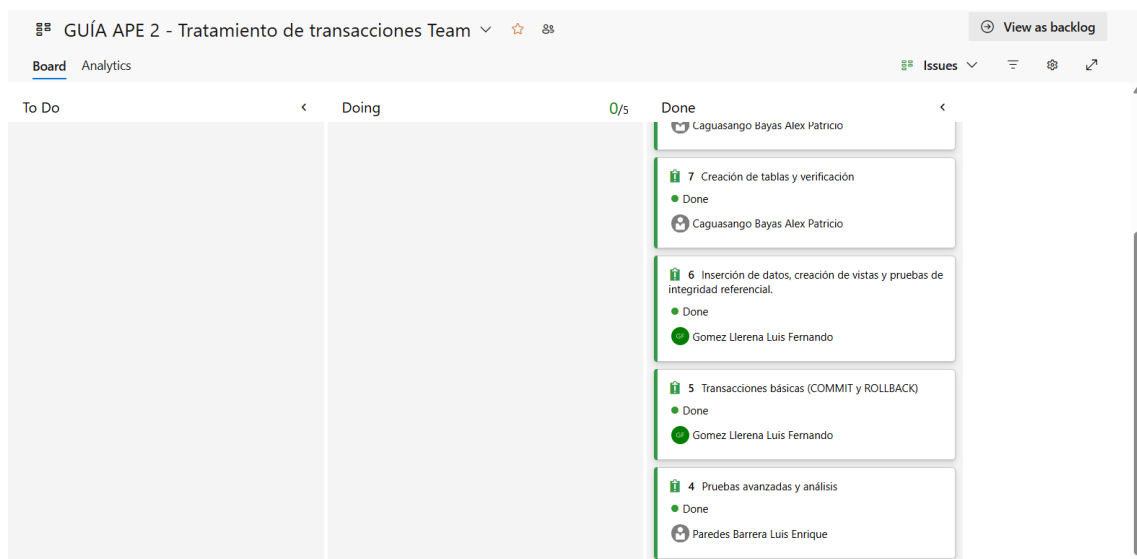
8 Diseño y creación de la base de datos y sus esquemas
Done
Caguasango Bayas Alex Patricio

7 Creación de tablas y verificación
Done
Caguasango Bayas Alex Patricio

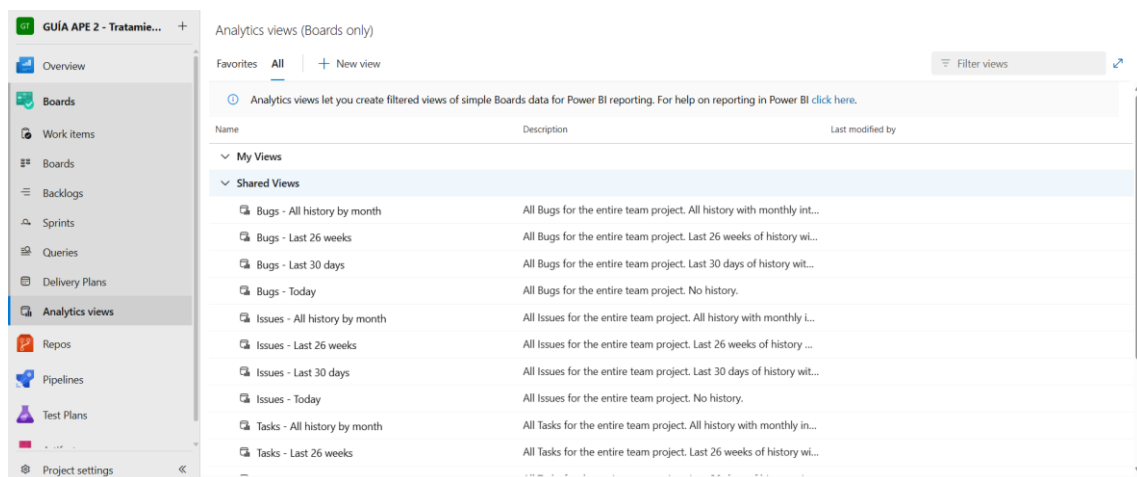
6 Inserción de datos, creación de vistas y pruebas de integridad referencial.
Done
Gomez Llerena Luis Fernando



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE Elige un elemento.
CICLO ACADÉMICO: MARZO – JULIO 2025



Análisis



2.9 Conclusiones

- Se comprobó que el uso de transacciones permite mantener la integridad de los datos, ya que COMMIT y ROLLBACK garantizan que las operaciones se ejecuten completamente o se reviertan en caso de error, evitando estados inconsistentes en la base de datos.
- Las pruebas en sesiones simultáneas demostraron que SQL Server maneja adecuadamente la concurrencia mediante bloqueos y niveles de aislamiento, asegurando que las transacciones de diferentes usuarios no interfieran entre sí ni alteren la consistencia de la información.
- El uso de SET XACT_ABORT y TRY...CATCH permitió controlar errores de forma eficiente, abortando o revirtiendo las transacciones cuando se presentaron fallos, lo que asegura la confiabilidad y estabilidad del sistema transaccional.

2.10 Recomendaciones

- Al manejar transacciones se debe cuidar de no producir un estado inconsistente de la BD. La captura de errores va a depender del tipo de error y su severidad.



UNIVERSIDAD TÉCNICA DE AMBATO
FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E INDUSTRIAL
CARRERA DE Elige un elemento.
CICLO ACADÉMICO: MARZO – JULIO 2025



- Al trabajar con sesiones concurrentes o ambientes distribuidos, se recomienda planificar el nivel de aislamiento y el manejo de errores de manera cuidadosa, para evitar bloqueos, inconsistencias o pérdida de información durante la ejecución de transacciones.
- Siempre documentar y probar las transacciones antes de aplicarlas en entornos reales, ya que esto permite detectar errores y garantizar que los cambios se realicen de forma segura sin afectar la integridad de la base de datos.

2.11 Anexos

