
Guía de Instalación y Configuración del Entorno "LogiTrack" para Ubuntu

Objetivo: Instalar y configurar un entorno de base de datos distribuida heterogénea utilizando MySQL y SQL Server.

Sección 1: Instalación de Software Esencial

Abre una terminal (Ctrl + Alt + T) y ejecuta los siguientes comandos.

1.1. Instalar MySQL Server y Workbench

- **Paso 1: Instalar el servidor de MySQL**

```
Bash
sudo apt update
sudo apt install mysql-server
```

- **Paso 2: Descargar MySQL Workbench**

Ve a la página oficial y descarga el paquete .DEB para Ubuntu:
<https://dev.mysql.com/downloads/workbench/>

- **Paso 3: Instalar MySQL Workbench**

Navega a tu carpeta de Descargas en la terminal y ejecuta la instalación (reemplaza el nombre del archivo).

```
Bash
cd ~/Descargas
sudo apt install ./mysql-workbench-community_..._amd64.deb
```

Si da un error de dependencias, solucióvalo con:

```
Bash
sudo apt --fix-broken install
```

1.2. Instalar Docker

- **Paso 1: Instalar dependencias de Docker**

```
Bash
sudo apt-get install ca-certificates curl gnupg
```

- **Paso 2: Añadir la clave GPG oficial de Docker**

Bash

```
sudo install -m 0755 -d /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o
/etc/apt/keyrings/docker.gpg
sudo chmod a+r /etc/apt/keyrings/docker.gpg
```

- **Paso 3: Configurar el repositorio de Docker**

Bash

```
echo \
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg]
https://download.docker.com/linux/ubuntu \
$(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

- **Paso 4: Instalar el Motor de Docker**

Bash

```
sudo apt-get update
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin
docker-compose-plugin
```

1.3. Instalar Azure Data Studio y Drivers ODBC

- Paso 1: Descargar Azure Data Studio

Ve a la página oficial y descarga el paquete .DEB:

<https://learn.microsoft.com/es-es/sql/azure-data-studio/download-azure-data-studio>

- **Paso 2: Instalar Azure Data Studio**

Bash

```
cd ~/Descargas
sudo apt install ./azuredatstudio-linux-....deb
```

- **Paso 3: Instalar los drivers de Microsoft**

Bash

```
curl https://packages.microsoft.com/keys/microsoft.asc | sudo apt-key add -
curl https://packages.microsoft.com/config/ubuntu/22.04/prod.list | sudo tee
/etc/apt/sources.list.d/mssql-release.list
sudo apt-get update
sudo apt-get install -y msodbcsql18 mssql-tools18
```

Sección 2: Configuración de los Servidores

2.1. Configurar MySQL Server

- Paso 1: Configurar la contraseña de root
Por defecto, el usuario root de MySQL en Ubuntu no usa contraseña. Para poder conectar con Workbench, debemos asignarle una.

Bash

```
# Entra a MySQL sin contraseña
```

```
sudo mysql
```

```
# Dentro de la consola mysql>, ejecuta:
```

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY  
'TuPasswordSeguroParaMySQL';
```

```
FLUSH PRIVILEGES;
```

```
EXIT;
```

- Paso 2: Asegurar la instalación

Bash

```
sudo mysql_secure_installation
```

(Introduce la contraseña que acabas de crear y responde 'Y' a las preguntas de seguridad).

- Paso 3: Habilitar el servicio
Para que MySQL se inicie automáticamente:

Bash

```
sudo systemctl enable mysql
```

```
sudo systemctl start mysql
```

2.2. Levantar el Contenedor de SQL Server

- Ejecuta el siguiente comando. **IMPORTANTE: Elige y anota una contraseña segura.**

Bash

```
sudo docker run \
```

```
-e "ACCEPT_EULA=Y" \
```

```
-e "MSSQL_SA_PASSWORD=TuPasswordSeguroParaSQLServer" \
```

```
-p 1433:1433 \
```

```
--name sql_server_quito \
```

```
-d \
```

```
mcr.microsoft.com/mssql/server:2022-latest
```

Sección 3: Creación de la Base de Datos y Población de Datos

3.1. En MySQL (Nodos Guayaquil y Cuenca)

- Abre **MySQL Workbench** y conéctate a tu servidor local (localhost:3306, usuario root, y la contraseña que estableciste).
- Abre una nueva pestaña de consulta y ejecuta el siguiente script completo:

SQL

```
CREATE DATABASE IF NOT EXISTS AndesExpress;  
USE AndesExpress;
```

```
DROP VIEW IF EXISTS Entrega_QTO, Entrega_GYE, Entrega_CUE;  
DROP TABLE IF EXISTS Entrega, Ruta, Mantenimiento, Vehiculo_Mantenimiento,  
Vehiculo_Operativo, Conductor, Centro_Operacion;
```

```
CREATE TABLE Centro_Operacion (id_centro_op INT PRIMARY KEY NOT NULL  
AUTO_INCREMENT, nombre_centro VARCHAR(100) NOT NULL UNIQUE, ciudad VARCHAR(50)  
NOT NULL, direccion VARCHAR(255) NULL, telefono_contacto VARCHAR(15) NULL);  
CREATE TABLE Conductor (id_conductor INT PRIMARY KEY NOT NULL AUTO_INCREMENT,  
cedula VARCHAR(10) NOT NULL UNIQUE, nombre VARCHAR(100) NOT NULL, apellido  
VARCHAR(100) NOT NULL, tipo_licencia ENUM('C', 'E') NOT NULL, fecha_contratacion DATE  
NOT NULL, Conductor_id_centro_op INT NOT NULL, FOREIGN KEY  
(Conductor_id_centro_op) REFERENCES Centro_Operacion(id_centro_op));  
CREATE TABLE Vehiculo_Operativo (id_vehiculo INT PRIMARY KEY NOT NULL  
AUTO_INCREMENT, placa VARCHAR(8) NOT NULL UNIQUE, Vehiculo_id_centro_op INT NOT  
NULL, FOREIGN KEY (Vehiculo_id_centro_op) REFERENCES  
Centro_Operacion(id_centro_op));  
CREATE TABLE Vehiculo_Mantenimiento (id_vehiculo INT PRIMARY KEY NOT NULL, marca  
VARCHAR(50) NOT NULL, modelo VARCHAR(50) NOT NULL, ano_fabricacion INT NOT NULL,  
capacidad_carga_kg DECIMAL(10,2) NOT NULL, FOREIGN KEY (id_vehiculo) REFERENCES  
Vehiculo_Operativo(id_vehiculo));  
CREATE TABLE Mantenimiento (id_mantenimiento INT PRIMARY KEY NOT NULL  
AUTO_INCREMENT, Mantenimiento_id_vehiculo INT NOT NULL, fecha_mantenimiento DATE  
NOT NULL, kilometraje INT NOT NULL, tipo_mantenimiento ENUM('Preventivo', 'Correctivo')  
NOT NULL, descripcion TEXT NOT NULL, costo DECIMAL(10,2) NOT NULL, FOREIGN KEY  
(Mantenimiento_id_vehiculo) REFERENCES Vehiculo_Operativo(id_vehiculo));  
CREATE TABLE Ruta (id_ruta INT PRIMARY KEY NOT NULL AUTO_INCREMENT,  
Ruta_id_vehiculo_asignado INT NOT NULL, Ruta_id_conductor_asignado INT NOT NULL,  
fecha_ruta DATE NOT NULL, estado_ruta ENUM('Programada', 'En curso', 'Completada') NOT  
NULL, kilometros_estimados INT NULL, FOREIGN KEY (Ruta_id_vehiculo_asignado)
```

```

REFERENCES Vehiculo_Operativo(id_vehiculo), FOREIGN KEY
(Ruta_id_conductor_asignado) REFERENCES Conductor(id_conductor));
CREATE TABLE Entrega (id_entrega INT PRIMARY KEY NOT NULL AUTO_INCREMENT,
Entrega_id_ruta_asignada INT NOT NULL, descripcion_paquete TEXT NULL, peso_kg
DECIMAL(10,2) NOT NULL, ciudad_origen VARCHAR(50) NOT NULL, ciudad_destino
VARCHAR(50) NOT NULL, direccion_destino VARCHAR(255) NOT NULL, estado_entrega
ENUM('En bodega','En tránsito','Entregado','Cancelado') NOT NULL, fecha_creacion
TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP, FOREIGN KEY
(Entrega_id_ruta_asignada) REFERENCES Ruta(id_ruta));
CREATE VIEW Entrega_QTO AS SELECT * FROM Entrega WHERE ciudad_destino = 'Quito';
CREATE VIEW Entrega_GYE AS SELECT * FROM Entrega WHERE ciudad_destino = 'Guayaquil';
CREATE VIEW Entrega_CUE AS SELECT * FROM Entrega WHERE ciudad_destino = 'Cuenca';

INSERT INTO Centro_Operacion (nombre_centro, ciudad, direccion) VALUES ('AndesExpress
- Hub Quito', 'Quito', 'Av. de los Shyris N35-174'), ('AndesExpress - Puerto GYE', 'Guayaquil', 'Av. 9
de Octubre 510'), ('AndesExpress - Austro CUE', 'Cuenca', 'Calle Larga 8-27');
INSERT INTO Conductor (cedula, nombre, apellido, tipo_licencia, fecha_contratacion,
Conductor_id_centro_op) VALUES ('1712345678', 'Juan', 'Perez', 'E', '2022-01-15', 1),
('0987654321', 'Carlos', 'Gomez', 'C', '2023-05-20', 2), ('0102030405', 'Ana', 'Martinez', 'E',
'2021-11-10', 3);
INSERT INTO Vehiculo_Operativo (placa, Vehiculo_id_centro_op) VALUES ('PBA-1111', 1),
('GBA-2222', 2), ('ABC-3333', 3);
INSERT INTO Vehiculo_Mantenimiento (id_vehiculo, marca, modelo, ano_fabricacion,
capacidad_carga_kg) VALUES (1, 'Hino', '500 Series', 2021, 5000.00), (2, 'Chevrolet',
'N-Series', 2023, 2500.00), (3, 'Mercedes-Benz', 'Actros', 2022, 7000.00);
INSERT INTO Ruta (Ruta_id_vehiculo_asignado, Ruta_id_conductor_asignado, fecha_ruta,
estado_ruta) VALUES (1, 1, '2025-07-13', 'Programada'), (2, 2, '2025-07-13', 'Programada'), (3, 3,
'2025-07-14', 'Programada'), (4, 1, '2025-07-15', 'Programada');
INSERT INTO Entrega (Entrega_id_ruta_asignada, descripcion_paquete, peso_kg,
ciudad_origen, ciudad_destino, direccion_destino, estado_entrega) VALUES (2, 'Repuestos
Electrónicos', 150.5, 'Guayaquil', 'Quito', 'Eloy Alfaro y República', 'En tránsito'), (3, 'Artesanías',
80.0, 'Cuenca', 'Quito', 'Amazonas y Patria', 'En bodega'), (1, 'Suministros Médicos', 300.0,
'Quito', 'Guayaquil', 'Malecón 2000, Local 5', 'En tránsito'), (3, 'Textiles', 250.75, 'Cuenca',
'Guayaquil', 'Urdesa Central, Calle Principal', 'En bodega'), (1, 'Libros y Papelería', 120.0, 'Quito',
'Cuenca', 'Av. Remigio Crespo Toral', 'En tránsito'), (4, 'Equipos de Sonido', 450.0, 'Quito',
'Guayaquil', 'Av. Victor Emilio Estrada 712', 'En bodega');

```

3.2. En SQL Server (Nodo Quito)

- Abre **Azure Data Studio** y conéctate al servidor SQL Server (127.0.0.1:1433, usuario sa, y la contraseña que estableciste).
- Abre una nueva pestaña de consulta y ejecuta el siguiente script T-SQL completo:
SQL

```
IF NOT EXISTS (SELECT * FROM sys.databases WHERE name = 'AndesExpress') CREATE
DATABASE AndesExpress;
GO
USE AndesExpress;
GO
```

```
IF OBJECT_ID('Entrega_QTO', 'V') IS NOT NULL DROP VIEW Entrega_QTO;
IF OBJECT_ID('Entrega', 'U') IS NOT NULL DROP TABLE Entrega;
IF OBJECT_ID('Ruta', 'U') IS NOT NULL DROP TABLE Ruta;
IF OBJECT_ID('Mantenimiento', 'U') IS NOT NULL DROP TABLE Mantenimiento;
IF OBJECT_ID('Vehiculo_Mantenimiento', 'U') IS NOT NULL DROP TABLE
Vehiculo_Mantenimiento;
IF OBJECT_ID('Vehiculo_Operativo', 'U') IS NOT NULL DROP TABLE Vehiculo_Operativo;
IF OBJECT_ID('Conductor', 'U') IS NOT NULL DROP TABLE Conductor;
IF OBJECT_ID('Centro_Operacion', 'U') IS NOT NULL DROP TABLE Centro_Operacion;
GO
```

```
CREATE TABLE Centro_Operacion (id_centro_op INT PRIMARY KEY IDENTITY(1,1) NOT NULL,
nombre_centro VARCHAR(100) NOT NULL UNIQUE, ciudad VARCHAR(50) NOT NULL,
direccion VARCHAR(255) NULL, telefono_contacto VARCHAR(15) NULL);
CREATE TABLE Conductor (id_conductor INT PRIMARY KEY IDENTITY(1,1) NOT NULL, cedula
VARCHAR(10) NOT NULL UNIQUE, nombre VARCHAR(100) NOT NULL, apellido VARCHAR(100)
NOT NULL, tipo_licencia VARCHAR(1) NOT NULL CHECK (tipo_licencia IN ('C', 'E')),
fecha_contratacion DATE NOT NULL, Conductor_id_centro_op INT NOT NULL, FOREIGN KEY
(Conductor_id_centro_op) REFERENCES Centro_Operacion(id_centro_op));
CREATE TABLE Vehiculo_Operativo (id_vehiculo INT PRIMARY KEY IDENTITY(1,1) NOT NULL,
placa VARCHAR(8) NOT NULL UNIQUE, Vehiculo_id_centro_op INT NOT NULL, FOREIGN KEY
(Vehiculo_id_centro_op) REFERENCES Centro_Operacion(id_centro_op));
CREATE TABLE Vehiculo_Mantenimiento (id_vehiculo INT PRIMARY KEY NOT NULL, marca
VARCHAR(50) NOT NULL, modelo VARCHAR(50) NOT NULL, ano_fabricacion INT NOT NULL,
capacidad_carga_kg DECIMAL(10,2) NOT NULL, FOREIGN KEY (id_vehiculo) REFERENCES
Vehiculo_Operativo(id_vehiculo));
CREATE TABLE Mantenimiento (id_mantenimiento INT PRIMARY KEY IDENTITY(1,1) NOT NULL,
Mantenimiento_id_vehiculo INT NOT NULL, fecha_mantenimiento DATE NOT NULL,
kilometraje INT NOT NULL, tipo_mantenimiento VARCHAR(20) NOT NULL CHECK
(tipo_mantenimiento IN ('Preventivo','Correctivo')), descripcion TEXT NOT NULL, costo
DECIMAL(10,2) NOT NULL, FOREIGN KEY (Mantenimiento_id_vehiculo) REFERENCES
Vehiculo_Operativo(id_vehiculo));
CREATE TABLE Ruta (id_ruta INT PRIMARY KEY IDENTITY(1,1) NOT NULL,
Ruta_id_vehiculo_asignado INT NOT NULL, Ruta_id_conductor_asignado INT NOT NULL,
fecha_ruta DATE NOT NULL, estado_ruta VARCHAR(20) NOT NULL CHECK (estado_ruta IN
('Programada','En curso','Completada')), kilometros_estimados INT NULL, FOREIGN KEY
(Ruta_id_vehiculo_asignado) REFERENCES Vehiculo_Operativo(id_vehiculo), FOREIGN KEY
```

```

(Ruta_id_conductor_asignado) REFERENCES Conductor(id_conductor));
CREATE TABLE Entrega (id_entrega INT PRIMARY KEY IDENTITY(1,1) NOT NULL,
Entrega_id_ruta_asignada INT NOT NULL, descripcion_paquete TEXT NULL, peso_kg
DECIMAL(10,2) NOT NULL, ciudad_origen VARCHAR(50) NOT NULL, ciudad_destino
VARCHAR(50) NOT NULL, direccion_destino VARCHAR(255) NOT NULL, estado_entrega
VARCHAR(20) NOT NULL CHECK (estado_entrega IN ('En bodega','En
tránsito','Entregado','Cancelado'))), fecha_creacion DATETIME2 NOT NULL DEFAULT
GETDATE(), FOREIGN KEY (Entrega_id_ruta_asignada) REFERENCES Ruta(id_ruta));
GO
CREATE VIEW Entrega_QTO AS SELECT * FROM Entrega WHERE ciudad_destino = 'Quito';
GO

```

```

SET IDENTITY_INSERT Centro_Operacion ON; INSERT INTO Centro_Operacion
(id_centro_op, nombre_centro, ciudad, direccion) VALUES (1, 'AndesExpress - Hub Quito',
'Quito', 'Av. de los Shyris N35-174'), (2, 'AndesExpress - Puerto GYE', 'Guayaquil', 'Av. 9 de Octubre
510'), (3, 'AndesExpress - Austro CUE', 'Cuenca', 'Calle Larga 8-27'); SET IDENTITY_INSERT
Centro_Operacion OFF;
SET IDENTITY_INSERT Conductor ON; INSERT INTO Conductor (id_conductor, cedula,
nombre, apellido, tipo_licencia, fecha_contratacion, Conductor_id_centro_op) VALUES (1,
'1712345678', 'Juan', 'Perez', 'E', '2022-01-15', 1), (2, '0987654321', 'Carlos', 'Gomez', 'C',
'2023-05-20', 2), (3, '0102030405', 'Ana', 'Martinez', 'E', '2021-11-10', 3); SET IDENTITY_INSERT
Conductor OFF;
SET IDENTITY_INSERT Vehiculo_Operativo ON; INSERT INTO Vehiculo_Operativo
(id_vehiculo, placa, Vehiculo_id_centro_op) VALUES (1, 'PBA-1111', 1), (2, 'GBA-2222', 2), (3,
'ABC-3333', 3); SET IDENTITY_INSERT Vehiculo_Operativo OFF;
INSERT INTO Vehiculo_Mantenimiento (id_vehiculo, marca, modelo, ano_fabricacion,
capacidad_carga_kg) VALUES (1, 'Hino', '500 Series', 2021, 5000.00), (2, 'Chevrolet',
'N-Series', 2023, 2500.00), (3, 'Mercedes-Benz', 'Actros', 2022, 7000.00);
SET IDENTITY_INSERT Ruta ON; INSERT INTO Ruta (id_ruta, Ruta_id_vehiculo_asignado,
Ruta_id_conductor_asignado, fecha_ruta, estado_ruta) VALUES (1, 1, 1, '2025-07-13',
'Programada'), (2, 2, 2, '2025-07-13', 'Programada'), (3, 3, 3, '2025-07-14', 'Programada'), (4, 1, 1,
'2025-07-15', 'Programada'); SET IDENTITY_INSERT Ruta OFF;
INSERT INTO Entrega (Entrega_id_ruta_asignada, descripcion_paquete, peso_kg,
ciudad_origen, ciudad_destino, direccion_destino, estado_entrega) VALUES (2, 'Repuestos
Electrónicos', 150.5, 'Guayaquil', 'Quito', 'Eloy Alfaro y República', 'En tránsito'), (3, 'Artesanías',
80.0, 'Cuenca', 'Quito', 'Amazonas y Patria', 'En bodega'), (1, 'Suministros Médicos', 300.0,
'Quito', 'Guayaquil', 'Malecón 2000, Local 5', 'En tránsito'), (3, 'Textiles', 250.75, 'Cuenca',
'Guayaquil', 'Urdesa Central, Calle Principal', 'En bodega'), (1, 'Libros y Papelería', 120.0, 'Quito',
'Cuenca', 'Av. Remigio Crespo Toral', 'En tránsito'), (4, 'Equipos de Sonido', 450.0, 'Quito',
'Guayaquil', 'Av. Victor Emilio Estrada 712', 'En bodega');

```

Sección 4: Verificación Final

Al finalizar, el nuevo usuario debería poder conectarse a ambos servidores y ejecutar consultas de prueba, confirmando que el entorno está 100% replicado.