



Instituto Tecnológico y de Estudios Superiores de Monterrey

Campus Monterrey

“Yo, como integrante de la comunidad estudiantil del Tecnológico de Monterrey, soy consciente de que la trampa y el engaño afectan mi dignidad como persona, mi aprendizaje y mi formación, por ello me comprometo a actuar honestamente, respetar y dar crédito al valor y esfuerzo con el que se elaboran las ideas propias, las de los compañeros y de los autores, así como asumir mi responsabilidad en la construcción de un ambiente de aprendizaje justo y confiable”

“Inteligencia artificial avanzada para la ciencia de datos I”

Momento de Retroalimentación 2

Datos

Equipo:

Frida Cano Falcón A01752953

Jorge Javier Sosa Briseño A01749489

Guillermo Romeo Cepeda Medina A01284015

Daniel Saldaña Rodríguez A00829752

Profesor:

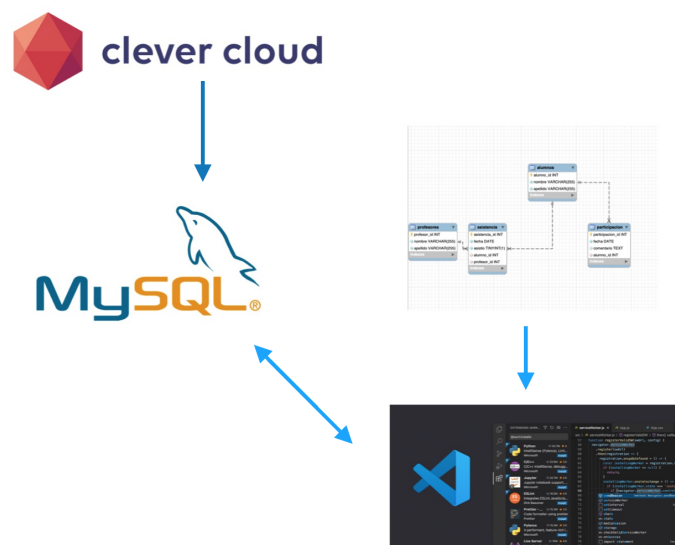
Fecha de entrega: 11 de octubre de 2023

Herramientas tecnológicas:

- **MySQL:** Es una de las bases de datos relacionales más populares. Decidimos usarla debido a su eficiencia, robustez y facilidad de uso. Además, nos permite estructurar nuestros datos de manera relacional y garantizar la integridad de los datos mediante claves primarias y foráneas.
- **Python y mysql-connector:** Python es un lenguaje de programación versátil y ampliamente adoptado en el análisis de datos y la ciencia de datos. La biblioteca mysql-connector nos permitió interactuar con nuestra base de datos MySQL directamente desde Python con el IDE de Visual Studio Code.
- **Clever Cloud:** Este sistema de almacenamiento en la nube se brinda de manera gratuita, aunado a esto esta plataforma nos permite demostrar desarrollar una solución de alojamiento en la nube orientada a nuestro proyecto ya que buscamos desplegar y administrar de forma eficiente la infraestructura de los datos.

Modelo de Almacenamiento:

En el siguiente esquema se presentará la manera en la que tenemos organizada la estructura de información.



Primeramente, se realizó una conexión con la nube del servidor de Clever Cloud en el cual nos permitió gracias a su estructura crear una base de datos en MySQL. Así mismo, para llevar a cabo todas las modificaciones de creación de tablas e inserto de datos, utilizamos el lenguaje de programación Python con el IDE de Visual Studio Code ya que es un lenguaje flexible y sencillo para el análisis y el manejo de los datos.

Diseñamos un esquema relacional en MySQL para atender las necesidades específicas del reto. El modelo consta de las siguientes tablas:

- *Alumnos*

Representa a cada estudiante individualmente. Los atributos incluyen:

- matrícula (clave primaria): de tipo caracter
- nombre: de tipo caracter
- apellido: de tipo caracter
- edad: de tipo entero
- carrera: de tipo caracter
- hora de inicio: de tipo time
- fin: de tipo time

- *Profesores*

Representa a cada profesor individualmente. Los atributos incluyen:

- matrícula (clave primaria): de tipo caracter
- nombre: de tipo caracter
- apellido: de tipo caracter
- edad: de tipo entero

- *Asistencia*

Registra la asistencia tanto de alumnos como de profesores.

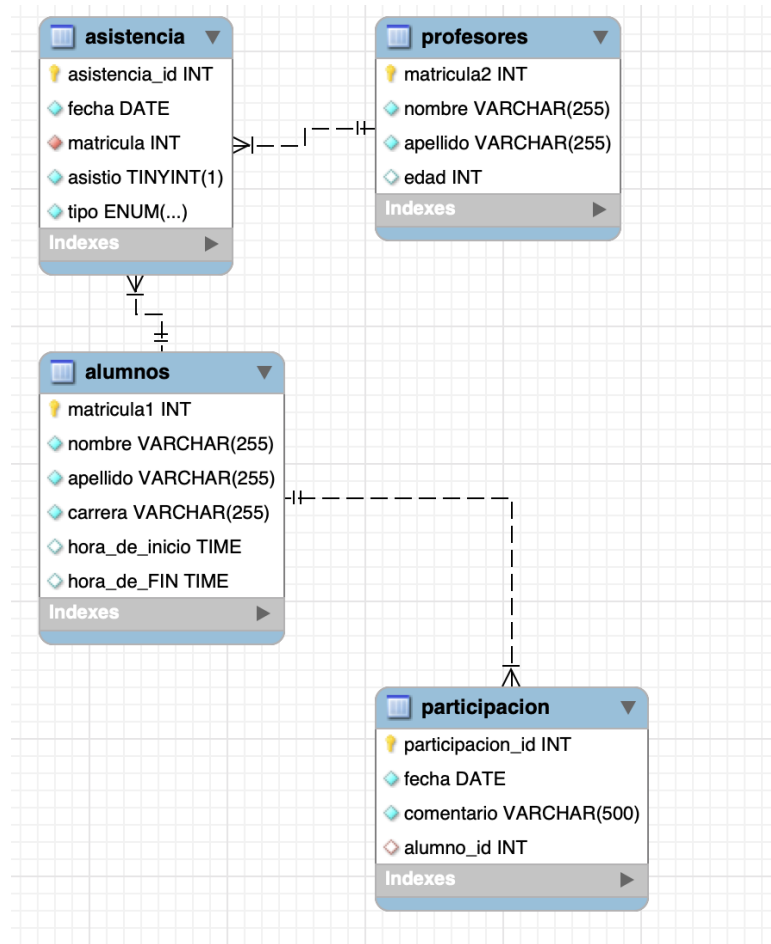
- id (clave única): tipo entero
- matricula: tipo caracter
- tipo: profesor o alumno
- asistio: tipo booleano

- *Participación*

Registra las participaciones de alumnos. Los atributos incluyen:

- id (clave única): de tipo entero
- fecha: tipo fecha
- detalle: tipo de texto
- alumno: tipo de texto

A continuación, se mostrará un esquema de cómo es que está estructurada nuestra base de datos con tablas relacionales:



- Relación entre alumnos y participacion:
 - La tabla participacion tiene una columna alumno_id que es una clave foránea que hace referencia a la columna matricula1 de la tabla alumnos.
 - Esto significa que cada registro en la tabla participacion está asociado con un alumno específico de la tabla alumnos.
 - Esta relación permite rastrear qué alumno hizo una participación específica y asegurarse de que cada participación esté asociada a un alumno existente.
- Relación entre alumnos y asistencia:
 - La tabla asistencia tiene una columna matricula que, cuando se refiere a un alumno, es una clave foránea que apunta a la columna matricula1 de la tabla alumnos.

- Esta relación permite rastrear qué alumnos asistieron en una fecha específica y asegurarse de que cada registro de asistencia de un alumno esté asociado con un alumno existente en la tabla alumnos.
- Relación entre profesores y asistencia:
 - La tabla asistencia también utiliza la columna matricula para hacer referencia a los profesores. En este caso, cuando se refiere a un profesor, es una clave foránea que apunta a la columna matricula2 de la tabla profesores.
 - Al igual que con los alumnos, esta relación permite rastrear qué profesores asistieron en una fecha específica y asegurarse de que cada registro de asistencia de un profesor esté asociado con un profesor existente en la tabla profesores.

Scripts de configuración:

Los scripts utilizados para poder desarrollar una base de datos y acceder a ella son los siguientes:

SQL DataSet creation:

Para configurar la base de datos lo primero que se realizó fue un script de SQL en donde se crean las tablas necesarias para el proyecto (las ya descritas en el punto Modelo de Almacenamiento).

```
-- Tablas principales
CREATE TABLE alumnos (
    matricula1 INT PRIMARY KEY ,
    nombre VARCHAR(255) NOT NULL,
    apellido VARCHAR(255) NOT NULL,
    carrera VARCHAR(255) NOT NULL,
    hora_de_inicio TIME,
    hora_de_FIN TIME
);

CREATE TABLE profesores (
    matricula2 INT PRIMARY KEY ,
    nombre VARCHAR(255) NOT NULL,
    apellido VARCHAR(255) NOT NULL,
    edad INT
);

-- Tablas relacionales
CREATE TABLE participacion (
```

```

    participacion_id INT PRIMARY KEY AUTO_INCREMENT,
    fecha DATE NOT NULL,
    comentario VARCHAR(500) NOT NULL,
    alumno_id INT,
    FOREIGN KEY (alumno_id) REFERENCES alumnos(matricula1)
);

CREATE TABLE asistencia (
    asistencia_id INT PRIMARY KEY AUTO_INCREMENT,
    fecha DATE NOT NULL,
    matricula INT NOT NULL,
    asistio BOOLEAN NOT NULL,
    tipo ENUM('alumno', 'profesor') NOT NULL,
    FOREIGN KEY (matricula) REFERENCES alumnos(matricula1) ON DELETE CASCADE
ON UPDATE CASCADE,
    FOREIGN KEY (matricula) REFERENCES profesores(matricula2) ON DELETE
CASCADE ON UPDATE CASCADE
);

```

Entorno de python:

Para la actualización de la base de datos utilizamos un script de python en donde integramos queries, gracias a la librería de *mysql connector*. En seguida presentamos la configuración del entorno para poder compilar el script:

```

appnope==0.1.3
asttokens==2.4.0
attrs==23.1.0
backcall==0.2.0
beautifulsoup4==4.12.2
bleach==6.0.0
blinker==1.6.2
click==8.1.7
comm==0.1.4
contourpy==1.1.1
cyclr==0.11.0
debugpy==1.8.0
decorator==5.1.1
defusedxml==0.7.1
distlib==0.3.7
executing==1.2.0
fastjsonschema==2.18.0
filelock==3.12.4
Flask==2.3.3
fonttools==4.42.1
ipykernel==6.25.2

```

ipython==8.15.0
itsdangerous==2.1.2
jedi==0.19.0
Jinja2==3.1.2
wcwidth==0.2.6
webencodings==0.5.1
Werkzeug==2.3.7

Separación de datos de entrenamiento y prueba:

El esquema k-fold cross validation no aplica para esta actividad debido a que no estamos manejando datos etiquetados para el entrenamiento y prueba de un algoritmo de machine learning. Este sería requisito para una actividad de machine learning, no concebimos el cómo implementarlo y justificarlo en nuestra base de datos.

Big Data? :

Aunque no existe un “threshold” que nos diga a partir de cuantos datos podemos considerar algo como Big Data, se suelen considerar tres características principales, conocidas como las 3 Vs.

Volumen de datos: Se refiere a la cantidad cruda de datos que manejas y aquí se puede determinar si aplica el termino de Big Data utilizando como criterio el si el análisis de estos datos con métodos convencionales de procesamiento de datos. En nuestro caso no requerimos métodos complicados de procesamiento, debido a que sólo se toma asistencia una vez por día y la manera en la que se cuentan las asistencias por clase es meramente un contador de asistencias en la base de datos.

Velocidad: Esto se refiere a la velocidad a la que los datos se generan y procesan, por ejemplo en redes sociales estos datos se generan cada segundo y hay que interpretarlos en seguida gracias a las funciones como #trending y los algoritmos de popularidad. No obstante, en nuestro caso, la velocidad a la que se procesan los datos no es un aspecto fundamental del proyecto, más bien lo que nos interesa es al final de la clase saber la asistencia de los alumnos y sus participaciones al final de la clase. lo que nos interesa más bien es la fidelidad de los datos.

Variedad: Un aspecto importante de los datos para considerarlos big data, es la variedad y los diferentes tipos de datos que pueden existir dentro de ellos, así como la veracidad, fidelidad y consistencia de los mismos, en nuestro caso como nosotros nos encargaremos de recompilar los datos con un mismo sistema replicable, no esperamos datos que no sean parecidos a lo que ya tenemos, mas bien nuestro sample es pequeño al tratarse de un salón de clase, con tipos de datos booleanos y enteros (Asistencias y Participaciones).

En conclusión, a pesar de que en nuestro caso de estudio, donde recopilaremos datos de un salon de clase de 30 personas no es considerado para nuestro poder de procesamiento y algoritmos como Big Data, este término es subjetivo y tal vez con unos cuantos salones,

diferentes tipos y características de datos, así como diferentes escuelas para nosotros podría ser Big Data, pero incluso así para una empresa grande no lo sería.