

---

---

# Base Universidad

Proyecto final Coderhouse

Matías Alberto Gonzalez Awe  
Comisión 57185  
SQL

---

---

# Introducción

En el entorno académico actual, la gestión eficiente y efectiva de la información es crucial para el buen funcionamiento de cualquier institución educativa. Las universidades manejan una vasta cantidad de datos relacionados con estudiantes, docentes, materias, aulas, departamentos y calificaciones. La complejidad y el volumen de esta información requieren un sistema robusto que facilite el almacenamiento, recuperación y análisis de los datos. Este proyecto se centra en el desarrollo de una base de datos para una universidad, diseñada para optimizar la gestión de estas entidades y sus relaciones. Utilizando MySQL, se ha creado una estructura que permite la interrelación de los diferentes componentes del sistema educativo universitario, asegurando integridad y consistencia de los datos.

---

---

# Objetivo

El objetivo principal de este proyecto es diseñar y desarrollar una base de datos relacional en MySQL que permita a la universidad gestionar de manera eficiente la información relacionada con estudiantes, docentes, materias, aulas, departamentos e inscripciones. La base de datos debe facilitar la realización de tareas administrativas, como la inscripción de estudiantes en materias, la asignación de docentes a diferentes departamentos y aulas, y la gestión de calificaciones. Además, se busca implementar mecanismos que aseguren la integridad referencial y permitan realizar transacciones de manera segura, incluyendo puntos de guardado y recuperación de datos. Con esto, se pretende mejorar la capacidad de toma de decisiones y la eficiencia operativa de la institución.

---

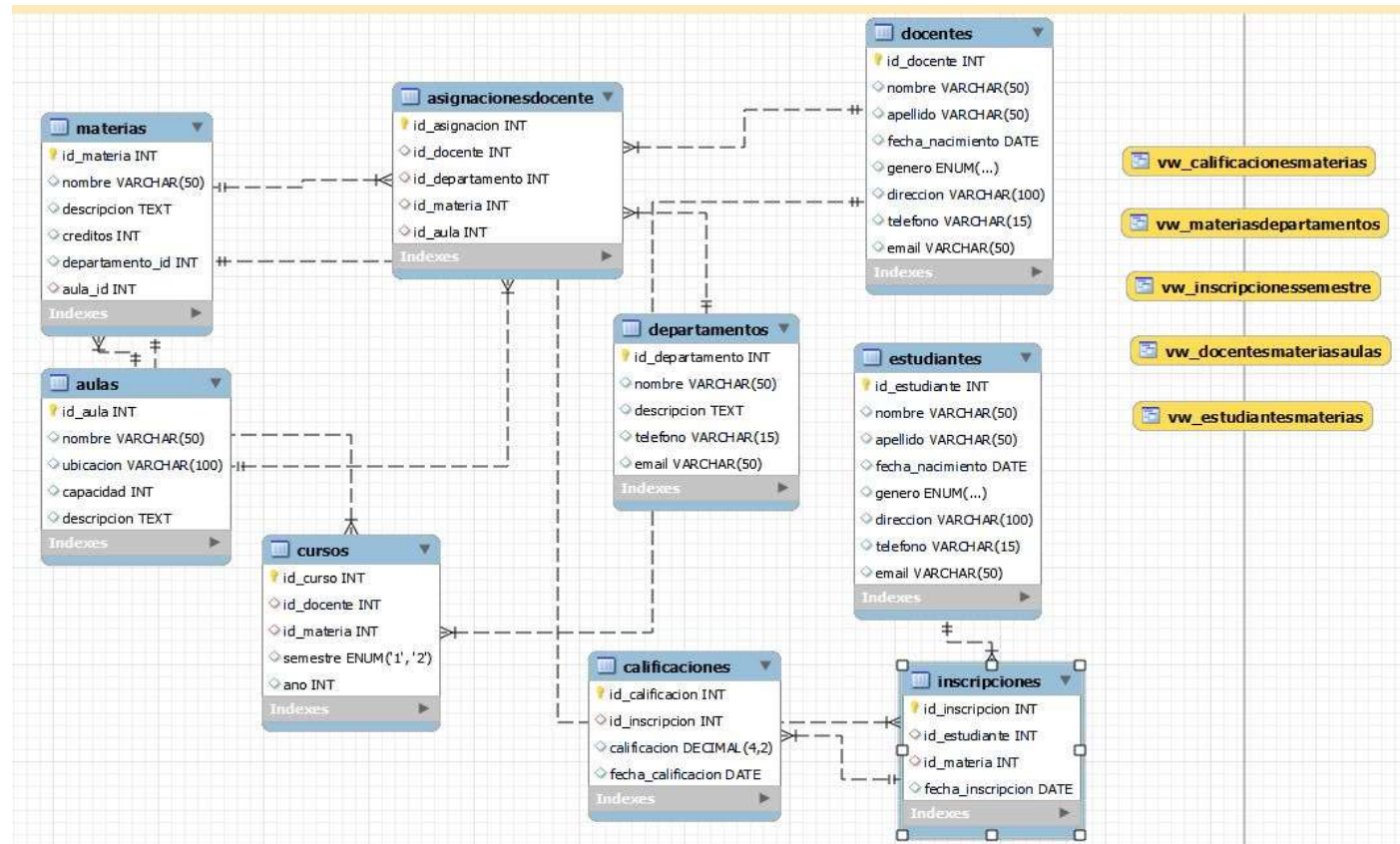
---

# Situación Problemática

La gestión de la información en una universidad presenta múltiples desafíos. La falta de un sistema centralizado y bien estructurado puede llevar a problemas como duplicación de datos, pérdida de información, errores en la inscripción de estudiantes y asignación de docentes, y dificultades en la generación de reportes y análisis de datos. Además, la integridad y consistencia de los datos se ven comprometidas cuando no se cuenta con controles adecuados para manejar transacciones y referencias entre tablas. En este contexto, la ausencia de un sistema eficiente puede afectar negativamente la operatividad y calidad del servicio educativo. Por lo tanto, surge la necesidad de desarrollar una base de datos que aborde estos problemas, proporcionando una solución que permita la administración eficaz de la información universitaria, asegurando la precisión y disponibilidad de los datos en todo momento.

---

# Diagrama entidad relación



---

# Tablas

Estudiantes: Almacena la información personal de los estudiantes, incluyendo su nombre, apellido, fecha de nacimiento, género, dirección, teléfono y correo electrónico. Variables y tipo de dato:

- id\_estudiante (Numérico)
  - nombre (Texto)
  - apellido (Texto)
  - fecha\_nacimiento (Fecha)
  - genero (Enumeración)
  - direccion (Texto)
  - telefono (Texto)
  - email (Texto)
-

---

# Tablas

Docentes: Almacena la información personal de los docentes, similar a la tabla de estudiantes, pero también incluye el departamento al que pertenecen. Variables y tipo de dato:

- id\_docente (Numérico)
  - nombre (Texto)
  - apellido (Texto)
  - fecha\_nacimiento (Fecha)
  - genero (Enumeración)
  - direccion (Texto)
  - telefono (Texto)
  - email (Texto)
-

---

# Tablas

Materias: Contiene información sobre las materias ofrecidas, incluyendo nombre, descripción, número de créditos y el departamento que las imparte. Variables y tipo de dato:

- id\_materia (Numérico)
  - nombre (Texto)
  - descripcion (Texto)
  - creditos (Numérico)
  - departamento\_id (Numérico)
  - aula\_id (Numérico)
-



---

# Tablas

Inscripciones: Relaciona a los estudiantes con las materias en las que están inscritos y registra la fecha de inscripción.

Variables y tipo de dato:

- id\_inscripcion (Numérico)
  - id\_estudiante (Numérico)
  - id\_materia (Numérico)
  - fecha\_inscripcion (Fecha)
-

---

# Tablas

Calificaciones: Registra las calificaciones obtenidas por los estudiantes en las materias en las que están inscritos, así como la fecha de la calificación. Variables y tipo de dato:

- id\_calificacion (Numérico)
  - id\_inscripcion (Numérico)
  - calificacion (Numérico)
  - fecha\_calificacion (Fecha)
-

---

# Tablas

Departamentos: Contiene información sobre los departamentos académicos, incluyendo nombre, descripción, teléfono y correo electrónico. Variables y tipo de dato:

- id\_departamento (Numérico)
  - nombre (Texto)
  - descripcion (Texto)
  - telefono (Texto)
  - email (Texto)
-

---

# Tablas

Cursos: Relaciona a los docentes con las materias que imparten, y almacena el semestre y año en que se imparten.

Variables y tipo de dato:

- id\_curso (Numérico)
  - id\_docente (Numérico)
  - id\_materia (Numérico)
  - semestre (Enumeración)
  - ano (Numérico)
-

---

# Tablas

Aulas: Almacena información sobre las aulas de la universidad, incluyendo nombre, ubicación, capacidad y descripción. Variables y tipo de dato:

- id\_aula (Numérico)
  - nombre (Texto)
  - ubicacion (Texto)
  - capacidad (Numérico)
  - descripcion (Texto)
-

---

# Tablas

AsignacionesDocente: Permite que un docente pueda estar en más de un departamento, impartir más de una materia y utilizar más de un aula. Esta tabla intermedia contendrá referencias a los docentes, departamentos, materias y aulas.

- id\_asignacion (Numérico)
  - id\_docente (Numérico)
  - id\_departamento (Numérico)
  - id\_materia (Numérico)
  - id\_aula (Numérico)
-

---

# Vistas

## 1- VW\_EstudiantesMaterias

Vista de Estudiantes con sus Inscripciones y Materias: Esta vista muestra información de los estudiantes junto con las materias en las que están inscritos.

## 2- VW\_DocentesMateriasAulas

Vista de Docentes con sus Materias y Aulas: Esta vista muestra información de los docentes junto con las materias que imparten y las aulas donde las imparten.

## 3- VW\_CalificacionesMaterias

Vista de Calificaciones por Materia: Esta vista muestra las calificaciones de los estudiantes organizadas por materia.

## 4- VW\_MateriasDepartamentos

Vista de Materias por Departamento: Esta vista muestra las materias organizadas por departamento.

## 5- VW\_InscripcionesSemestre

Vista de Inscripciones por Semestre: Esta vista muestra las inscripciones de estudiantes en materias organizadas por semestre.

---

---

# Funciones

- obtener\_promedio\_estudiante

Obtener el promedio de calificaciones de un estudiante

- contar\_estudiantes\_inscriptos

Contar el número de estudiantes inscriptos en una materia y semestre

---



---

## Store Procedure

- SP\_ordenar\_estudiantes

Permite ordenar la tabla Estudiantes según un campo y un orden especificados como parámetros

- SP\_gestionar\_registros

Permite gestionar registros en la tabla “Estudiantes”, tanto la inserción de nuevos registros como la eliminación de registros específicos

---

---

# Triggers

- TRG\_before\_insert\_aulas y TRG\_after\_update\_aulas:

Estos triggers aseguran que cualquier inserción o actualización en la tabla Aulas se registre en la tabla log\_Audit\_Aulas, proporcionando un historial detallado de las operaciones realizadas.

- TRG\_before\_insert\_docentes y TRG\_after\_update\_docentes:

Estos triggers registran automáticamente las operaciones de inserción y actualización en la tabla Docentes en la tabla log\_Audit\_Docentes, proporcionando un historial detallado de las operaciones realizadas, incluyendo el usuario, la fecha y la hora de cada operación.

---

---

## **USUARIOS Y PERMISOS (DCL)**

Se generaron los usuarios que van a interactuar con la base de datos de la institución, junto con sus respectivos permisos, para lectura, escritura y modificación según corresponda.

Usuario 1 con Permisos de Solo Lectura y Usuario 2 con Permisos de Lectura, Inserción y Modificación

---

---

## Herramientas utilizadas y Backup

- MySQL Workbench
  - El backup contiene el respaldo de los datos del proyecto, ante cualquier inconveniente es posible recuperar los mismos.
-