**Documentación de la Base de Datos para el Sistema de Gestión Académica de CEVAM**

**1. Introducción**

Este documento detalla el diseño de la base de datos para el sistema de gestión académica de CEVAM. El propósito principal de esta aplicación es centralizar y optimizar la gestión académica y administrativa de estudiantes y profesores, enfocándose en el seguimiento del progreso en libros y unidades. La solución busca resolver la desorganización de la información y la ineficiencia en el seguimiento del avance académico, mejorando la eficiencia interna y la experiencia educativa general para estudiantes y profesores.

**2. Objetivos del Sistema**

* **Centralización de la Información**: Consolidar datos de estudiantes, profesores, libros y progreso en un único lugar.
* **Optimización del Seguimiento Académico**: Permitir a profesores marcar unidades completadas y a estudiantes visualizar su progreso.
* **Mejora de la Eficiencia Operativa**: Reducir la carga administrativa del personal y el tiempo dedicado a buscar y organizar información.
* **Personalización de la Experiencia Educativa**: Ofrecer un seguimiento más claro y personalizado a los estudiantes.
* **Escalabilidad**: Proporcionar una base sólida para manejar un creciente número de estudiantes y profesores en el futuro.

**3. Diagrama Entidad-Relación (DER) / Diagrama de Base de Datos**

erDiagram

ROLES {

INT id\_rol PK

VARCHAR nombre\_rol UNIQUE

}

USUARIOS {

INT id\_usuario PK

VARCHAR nombre

VARCHAR apellido

VARCHAR correo UNIQUE

VARCHAR contrasena\_hash

VARCHAR cedula\_identidad UNIQUE

VARCHAR numero\_celular UNIQUE

INT id\_rol FK

TIMESTAMP fecha\_creacion

TIMESTAMP fecha\_actualizacion

}

LIBROS {

INT id\_libro PK

VARCHAR nombre\_libro

VARCHAR nivel

TEXT descripcion

TIMESTAMP fecha\_creacion

TIMESTAMP fecha\_actualizacion

}

UNIDADES {

INT id\_unidad PK

INT id\_libro FK

VARCHAR nombre\_unidad

INT numero\_unidad

TEXT descripcion

}

TIPOS\_CONTENIDO {

INT id\_tipo\_contenido PK

VARCHAR nombre\_tipo UNIQUE

}

TIPOS\_MATERIAL {

INT id\_tipo\_material PK

VARCHAR nombre\_material UNIQUE

}

CONTENIDOS\_UNIDAD {

INT id\_contenido\_unidad PK

INT id\_unidad FK

INT id\_tipo\_contenido FK

INT id\_tipo\_material FK

VARCHAR titulo

TEXT descripcion

VARCHAR url\_recurso

TEXT contenido\_texto

TIMESTAMP fecha\_creacion

TIMESTAMP fecha\_actualizacion

}

ESTUDIANTES\_LIBROS {

INT id\_estudiante\_libro PK

INT id\_usuario\_estudiante FK

INT id\_libro FK

TIMESTAMP fecha\_asignacion

}

PROGRESO\_UNIDADES {

INT id\_progreso PK

INT id\_usuario\_estudiante FK

INT id\_unidad FK

DATE fecha\_completado

TEXT comentarios\_profesor

}

PROFESORES\_ESTUDIANTES {

INT id\_profesor\_estudiante PK

INT id\_usuario\_profesor FK

INT id\_usuario\_estudiante FK

TIMESTAMP fecha\_asignacion

}

ROLES ||--o{ USUARIOS : tiene

USUARIOS ||--o{ ESTUDIANTES\_LIBROS : asignado\_a

USUARIOS ||--o{ PROGRESO\_UNIDADES : registra

USUARIOS ||--o{ PROFESORES\_ESTUDIANTES : enseña\_a

LIBROS ||--o{ UNIDADES : contiene

LIBROS ||--o{ ESTUDIANTES\_LIBROS : asignado\_a

UNIDADES ||--o{ PROGRESO\_UNIDADES : tiene\_progreso

UNIDADES ||--o{ CONTENIDOS\_UNIDAD : tiene\_contenido

TIPOS\_CONTENIDO ||--o{ CONTENIDOS\_UNIDAD : clasifica\_contenido

TIPOS\_MATERIAL ||--o{ CONTENIDOS\_UNIDAD : clasifica\_material

USUARIOS }o--|| PROFESORES\_ESTUDIANTES : gestiona

### 4. Descripción Detallada de las Tablas y sus Relaciones

A continuación, se describe cada tabla, sus campos y las relaciones que establece con otras tablas en la base de datos:

#### 4.1. Roles

* **Propósito**: Almacena los distintos tipos de usuarios que interactuarán con la aplicación, permitiendo una gestión de permisos y funcionalidades basada en roles.
* **Campos**:
  + id\_rol (INT, PK, AUTO\_INCREMENT): Identificador único del rol.
  + nombre\_rol (VARCHAR(50), NOT NULL, UNIQUE): Nombre del rol (ej., 'Administrador', 'Profesor', 'Estudiante').
* **Relaciones**:
  + **Uno a Muchos** con Usuarios: Un rol puede ser asignado a múltiples usuarios.

#### 4.2. Usuarios

* **Propósito**: Contiene la información fundamental de todos los usuarios del sistema (estudiantes, profesores y administradores).
* **Campos**:
  + id\_usuario (INT, PK, AUTO\_INCREMENT): Identificador único del usuario.
  + nombre (VARCHAR(100), NOT NULL): Nombre del usuario.
  + apellido (VARCHAR(100)): Apellido del usuario.
  + correo (VARCHAR(100), NOT NULL, UNIQUE): Correo electrónico del usuario, utilizado para el inicio de sesión y debe ser único.
  + contrasena\_hash (VARCHAR(255), NOT NULL): Hash seguro de la contraseña del usuario.
  + cedula\_identidad (VARCHAR(20), UNIQUE): Número de cédula de identidad del usuario, debe ser único.
  + numero\_celular (VARCHAR(20), UNIQUE): Número de teléfono celular del usuario, debe ser único.
  + id\_rol (INT, FK): Clave foránea que referencia a la tabla Roles, indicando el rol del usuario.
  + fecha\_creacion (TIMESTAMP): Fecha y hora en que se creó el registro del usuario.
  + fecha\_actualizacion (TIMESTAMP): Última fecha y hora en que se actualizó el registro del usuario.
* **Relaciones**:
  + **Muchos a Uno** con Roles: Cada usuario pertenece a un único rol.
  + **Uno a Muchos** con Estudiantes\_Libros: Un usuario (estudiante) puede tener varios libros asignados.
  + **Uno a Muchos** con Progreso\_Unidades: Un usuario (estudiante) registra progreso en varias unidades.
  + **Uno a Muchos** con Profesores\_Estudiantes: Un usuario puede ser profesor de varios estudiantes o un estudiante puede ser asignado a un profesor.

#### 4.3. Libros

* **Propósito**: Almacena los detalles de los libros de inglés utilizados en CEVAM.
* **Campos**:
  + id\_libro (INT, PK, AUTO\_INCREMENT): Identificador único del libro.
  + nombre\_libro (VARCHAR(255), NOT NULL): Título del libro.
  + nivel (VARCHAR(50), NOT NULL): Nivel del libro (ej., 'Adolescentes', 'Adultos').
  + descripcion (TEXT): Descripción opcional del libro.
  + fecha\_creacion (TIMESTAMP): Fecha y hora de creación del registro del libro.
  + fecha\_actualizacion (TIMESTAMP): Última fecha y hora de actualización del registro del libro.
* **Relaciones**:
  + **Uno a Muchos** con Unidades: Un libro contiene múltiples unidades.
  + **Uno a Muchos** con Estudiantes\_Libros: Un libro puede ser asignado a múltiples estudiantes.

#### 4.4. Unidades

* **Propósito**: Detalla cada una de las unidades que componen los libros.
* **Campos**:
  + id\_unidad (INT, PK, AUTO\_INCREMENT): Identificador único de la unidad.
  + id\_libro (INT, FK): Clave foránea que referencia a la tabla Libros, indicando a qué libro pertenece la unidad.
  + nombre\_unidad (VARCHAR(255), NOT NULL): Nombre o título de la unidad.
  + numero\_unidad (INT, NOT NULL): Número secuencial de la unidad dentro de su libro.
  + descripcion (TEXT): Descripción opcional de la unidad.
  + **Restricción UNIQUE**: (id\_libro, numero\_unidad) asegura que no haya dos unidades con el mismo número dentro del mismo libro.
* **Relaciones**:
  + **Muchos a Uno** con Libros: Varias unidades pertenecen a un mismo libro.
  + **Uno a Muchos** con Progreso\_Unidades: Una unidad puede tener el progreso registrado por múltiples estudiantes.
  + **Uno a Muchos** con Contenidos\_Unidad: Una unidad puede tener múltiples recursos o contenidos asociados.

#### 4.5. Tipos\_Contenido

* **Propósito**: Cataloga las grandes categorías de contenido pedagógico, como se muestra en la estructura de contenidos proporcionada (Grammar, Listening, Speaking, Reading, Writing).
* **Campos**:
  + id\_tipo\_contenido (INT, PK, AUTO\_INCREMENT): Identificador único del tipo de contenido.
  + nombre\_tipo (VARCHAR(100), NOT NULL, UNIQUE): Nombre de la categoría de contenido (ej., 'Grammar').
* **Relaciones**:
  + **Uno a Muchos** con Contenidos\_Unidad: Un tipo de contenido puede estar asociado a múltiples recursos de unidades.

#### 4.6. Tipos\_Material

* **Propósito**: Cataloga los tipos específicos de materiales educativos dentro de cada categoría de contenido (ej., Explanations, Videos, Exercises, Practice tests, Samples).
* **Campos**:
  + id\_tipo\_material (INT, PK, AUTO\_INCREMENT): Identificador único del tipo de material.
  + nombre\_material (VARCHAR(100), NOT NULL, UNIQUE): Nombre del tipo de material (ej., 'Explanations', 'Videos').
* **Relaciones**:
  + **Uno a Muchos** con Contenidos\_Unidad: Un tipo de material puede ser utilizado en múltiples recursos de unidades.

#### 4.7. Contenidos\_Unidad

* **Propósito**: Relaciona las unidades con los recursos educativos específicos (videos, audios, textos, ejercicios) que las componen, permitiendo una gestión detallada de los materiales por unidad. Esta tabla responde a la necesidad de gestionar contenidos como "Recursos Favoritos" o que los profesores puedan "subir" o "editar" contenidos.
* **Campos**:
  + id\_contenido\_unidad (INT, PK, AUTO\_INCREMENT): Identificador único del contenido de la unidad.
  + id\_unidad (INT, FK): Clave foránea que referencia a la tabla Unidades, indicando a qué unidad pertenece este contenido.
  + id\_tipo\_contenido (INT, FK): Clave foránea que referencia a la tabla Tipos\_Contenido, categorizando el contenido (ej., Grammar).
  + id\_tipo\_material (INT, FK): Clave foránea que referencia a la tabla Tipos\_Material, especificando el tipo de material (ej., Videos).
  + titulo (VARCHAR(255), NOT NULL): Título descriptivo del recurso.
  + descripcion (TEXT): Descripción del recurso.
  + url\_recurso (VARCHAR(255)): URL si el recurso es un archivo externo (video, audio, PDF).
  + contenido\_texto (TEXT): El contenido del recurso si es texto directo (ej., una explicación escrita, un ejercicio de texto).
  + fecha\_creacion (TIMESTAMP): Fecha y hora de creación del recurso.
  + fecha\_actualizacion (TIMESTAMP): Última fecha y hora de actualización del recurso.
  + **Restricción UNIQUE**: (id\_unidad, id\_tipo\_contenido, id\_tipo\_material, titulo) asegura que un recurso específico no se duplique en una unidad.
* **Relaciones**:
  + **Muchos a Uno** con Unidades: Varios contenidos pueden pertenecer a una misma unidad.
  + **Muchos a Uno** con Tipos\_Contenido: Un contenido es de un tipo de contenido específico.
  + **Muchos a Uno** con Tipos\_Material: Un contenido es de un tipo de material específico.

#### 4.8. Estudiantes\_Libros

* **Propósito**: Registra qué libros están asignados a cada estudiante, permitiendo a los estudiantes visualizar los libros que están cursando y sus unidades.
* **Campos**:
  + id\_estudiante\_libro (INT, PK, AUTO\_INCREMENT): Identificador único de la asignación.
  + id\_usuario\_estudiante (INT, FK): Clave foránea que referencia al id\_usuario de la tabla Usuarios (solo para usuarios con rol de Estudiante).
  + id\_libro (INT, FK): Clave foránea que referencia al id\_libro de la tabla Libros.
  + fecha\_asignacion (TIMESTAMP): Fecha en que se asignó el libro al estudiante.
  + **Restricción UNIQUE**: (id\_usuario\_estudiante, id\_libro) impide que un estudiante tenga el mismo libro asignado dos veces.
* **Relaciones**:
  + **Muchos a Uno** con Usuarios (rol Estudiante): Un estudiante puede tener múltiples libros asignados.
  + **Muchos a Uno** con Libros: Un libro puede ser asignado a múltiples estudiantes.

#### 4.9. Progreso\_Unidades

* **Propósito**: Almacena el registro del progreso de cada estudiante en unidades específicas, indicando cuándo se completó una unidad y permitiendo notas del profesor. Esta es una funcionalidad esencial para el seguimiento académico.
* **Campos**:
  + id\_progreso (INT, PK, AUTO\_INCREMENT): Identificador único del registro de progreso.
  + id\_usuario\_estudiante (INT, FK): Clave foránea que referencia al id\_usuario de la tabla Usuarios (solo para usuarios con rol de Estudiante).
  + id\_unidad (INT, FK): Clave foránea que referencia al id\_unidad de la tabla Unidades.
  + fecha\_completado (DATE, NOT NULL): Fecha en que la unidad fue marcada como completada.
  + comentarios\_profesor (TEXT): Notas opcionales del profesor sobre el desempeño del estudiante en esa unidad.
  + **Restricción UNIQUE**: (id\_usuario\_estudiante, id\_unidad) asegura que un estudiante solo pueda marcar una unidad como completada una vez.
* **Relaciones**:
  + **Muchos a Uno** con Usuarios (rol Estudiante): Un estudiante puede tener progreso en múltiples unidades.
  + **Muchos a Uno** con Unidades: Una unidad puede tener progreso registrado por múltiples estudiantes.

#### 4.10. Profesores\_Estudiantes

* **Propósito**: Relaciona a los profesores con los estudiantes que tienen a su cargo. Esto es fundamental para que un profesor pueda ver el progreso de "todos sus estudiantes en un dashboard".
* **Campos**:
  + id\_profesor\_estudiante (INT, PK, AUTO\_INCREMENT): Identificador único de la asignación profesor-estudiante.
  + id\_usuario\_profesor (INT, FK): Clave foránea que referencia al id\_usuario de la tabla Usuarios (solo para usuarios con rol de Profesor).
  + id\_usuario\_estudiante (INT, FK): Clave foránea que referencia al id\_usuario de la tabla Usuarios (solo para usuarios con rol de Estudiante).
  + fecha\_asignacion (TIMESTAMP): Fecha en que se estableció la relación profesor-estudiante.
  + **Restricción UNIQUE**: (id\_usuario\_profesor, id\_usuario\_estudiante) asegura que un profesor solo sea asignado una vez a un estudiante.
* **Relaciones**:
  + **Muchos a Uno** con Usuarios (rol Profesor): Un profesor puede gestionar a múltiples estudiantes.
  + **Muchos a Uno** con Usuarios (rol Estudiante): Un estudiante puede ser gestionado por un profesor.

### 5. Consideraciones Adicionales

* **Seguridad**: Se contempla el almacenamiento de contrasena\_hash en lugar de contraseñas en texto plano, lo cual es una buena práctica de seguridad.
* **Escalabilidad**: El diseño modular con tablas separadas para roles, libros, unidades, tipos de contenido y materiales permite que la base de datos crezca y se adapte a futuras funcionalidades sin requerir cambios estructurales mayores.
* **Rendimiento**: Se han incluido índices (CREATE INDEX) en campos clave (id\_rol, id\_libro, id\_usuario\_estudiante, id\_unidad, id\_tipo\_contenido, id\_tipo\_material) para optimizar el rendimiento de las consultas, especialmente aquellas que implican búsquedas y uniones frecuentes.
* **Flexibilidad de Contenidos**: La inclusión de las tablas Tipos\_Contenido, Tipos\_Material y Contenidos\_Unidad permite una estructura flexible para categorizar y asociar diversos recursos educativos a las unidades, preparando la aplicación para futuras funcionalidades como "Recursos Favoritos" y la edición de contenidos por parte de los profesores.
* **Restricciones de Unicidad**: Se han aplicado restricciones UNIQUE a correo, cedula\_identidad y numero\_celular en la tabla Usuarios para garantizar que estos identificadores clave sean únicos para cada persona en el sistema.

Esta estructura de base de datos proporciona una base robusta y escalable para el sistema de gestión académica de CEVAM, cubriendo todas las funcionalidades prioritarias y sentando las bases para futuras expansiones.