**DISEÑO DE LA BASE DE DATOS**

**🏫 1. Institutos**

Cada usuario pertenece a un instituto.

| id (PK) | nombre | clave | direccion | telefono | logo\_url |

**👤 2. Usuarios**

Incluye alumnos, docentes, directivos, etc.

| id (PK) | nombre\_completo | correo | password | telefono (opcional) | rol (enum: alumno, psicologia, maestro\_planta, maestro\_contrato, sub-director, director) | estado (bit) | matricula | instituto\_id (FK) | verificar\_correo (datetime o null) | codigo\_verificacion (varchar) |

**📚 3. Materias**

| id (PK) | nombre | clave | creditos | tipo (obligatoria / optativa) | grado\_id (FK) |

**🎓 4. Grados**

Define cada grado, ejemplo: 1° primaria, 2° secundaria, 3° prepa, etc.

| id (PK) | nombre | nivel (primaria, secundaria, preparatoria) | tipo\_periodo (semestre/cuatrimestre) | orden (1,2,3...) |

orden sirve para saber cuál sigue después.

**🏫 5. Grupos**

Ejemplo: "1A", "2B", etc.

| id (PK) | nombre | grado\_id (FK) | ciclo\_escolar (ej: "2025-2026") |

**🕑 6. Horarios**

| id (PK) | grupo\_id (FK) | materia\_id (FK) | docente\_id (FK a usuarios.id) | dia\_semana | hora\_inicio | hora\_fin |

**🧾 7. Calificaciones**

| id (PK) | alumno\_id (FK a usuarios.id) | materia\_id (FK) | grupo\_id (FK) | periodo (cuatri o semestre actual) | calificacion | reprobado (bit) | nivelacion (calificacion final si tuvo) | aprobo\_periodo (bit) |

**🎓 8. Avance / graduación**

Tabla para saber a qué grado/cuatrimestre está el alumno y cambiarlo automáticamente cuando aprueba.

| id (PK) | alumno\_id (FK) | grado\_actual\_id (FK) | grupo\_actual\_id (FK) | status (en\_curso, graduado) | fecha\_cambio |

Cuando termine último grado y aprueba → status = graduado.

**🗃 9. Documentos**

| id (PK) | alumno\_id (FK) | nombre | url | tipo (constancia, boleta, etc.) | fecha\_subida |

**🔔 10. Notificaciones**

| id (PK) | usuario\_id (FK) | titulo | mensaje | leido (bit) | fecha\_envio |

**📜 11. Antecedentes escolares**

Historial del alumno antes de entrar.

| id (PK) | alumno\_id (FK) | instituto\_origen\_id (FK) | nivel (primaria, secundaria, etc.) | observaciones | logros | reportes | fecha\_inicio | fecha\_fin |

instituto\_origen\_id conecta con tabla institutos para saber de qué escuela viene.

**⚙️ 12. Logros y reportes**

Tabla para logros o reportes internos mientras está en el instituto actual.

| id (PK) | alumno\_id (FK) | instituto\_id (FK) | tipo (logro/reporte) | descripcion | fecha |

**🚀 13. Tabla para pasar de grado automáticamente**

Cuando termine el cuatri/semestre, se consulta quienes aprobo\_periodo=1 en calificaciones, y:

* Si no es el último grado → crea nuevo registro en avance con el grado siguiente.
* Si es el último → actualiza status=graduado.

SQL:

CREATE DATABASE IF NOT EXISTS school\_manager;

USE school\_manager;

-- Tabla de institutos

CREATE TABLE institutos (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(255) NOT NULL,

clave VARCHAR(100),

direccion TEXT,

telefono VARCHAR(20),

logo\_url VARCHAR(500)

);

-- Tabla de usuarios

CREATE TABLE usuarios (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre\_completo VARCHAR(255) NOT NULL,

correo VARCHAR(255) NOT NULL UNIQUE,

password VARCHAR(255) NOT NULL,

telefono VARCHAR(20),

rol ENUM('alumno', 'psicologia', 'maestro\_planta', 'maestro\_contrato', 'sub-director', 'director') NOT NULL,

estado BOOLEAN DEFAULT 1,

matricula VARCHAR(50),

instituto\_id INT,

verificar\_correo DATETIME NULL,

codigo\_verificacion VARCHAR(100) NULL,

FOREIGN KEY (instituto\_id) REFERENCES institutos(id)

);

-- Tabla de grados

CREATE TABLE grados (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(100) NOT NULL,

nivel ENUM('primaria', 'secundaria', 'preparatoria') NOT NULL,

tipo\_periodo ENUM('semestre', 'cuatrimestre') NOT NULL,

orden INT NOT NULL

);

-- Tabla de grupos

CREATE TABLE grupos (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(50) NOT NULL,

grado\_id INT NOT NULL,

ciclo\_escolar VARCHAR(20) NOT NULL,

FOREIGN KEY (grado\_id) REFERENCES grados(id)

);

-- Tabla de materias

CREATE TABLE materias (

id INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(255) NOT NULL,

clave VARCHAR(50),

creditos INT,

tipo ENUM('obligatoria', 'optativa'),

grado\_id INT NOT NULL,

FOREIGN KEY (grado\_id) REFERENCES grados(id)

);

-- Tabla de horarios

CREATE TABLE horarios (

id INT AUTO\_INCREMENT PRIMARY KEY,

grupo\_id INT NOT NULL,

materia\_id INT NOT NULL,

docente\_id INT NOT NULL,

dia\_semana ENUM('lunes', 'martes', 'miércoles', 'jueves', 'viernes', 'sábado'),

hora\_inicio TIME,

hora\_fin TIME,

FOREIGN KEY (grupo\_id) REFERENCES grupos(id),

FOREIGN KEY (materia\_id) REFERENCES materias(id),

FOREIGN KEY (docente\_id) REFERENCES usuarios(id)

);

-- Tabla de calificaciones

CREATE TABLE calificaciones (

id INT AUTO\_INCREMENT PRIMARY KEY,

alumno\_id INT NOT NULL,

materia\_id INT NOT NULL,

grupo\_id INT NOT NULL,

periodo VARCHAR(20),

calificacion DECIMAL(5,2),

reprobado BOOLEAN DEFAULT 0,

nivelacion DECIMAL(5,2) NULL,

aprobo\_periodo BOOLEAN DEFAULT 0,

FOREIGN KEY (alumno\_id) REFERENCES usuarios(id),

FOREIGN KEY (materia\_id) REFERENCES materias(id),

FOREIGN KEY (grupo\_id) REFERENCES grupos(id)

);

-- Tabla de avance

CREATE TABLE avance (

id INT AUTO\_INCREMENT PRIMARY KEY,

alumno\_id INT NOT NULL,

grado\_actual\_id INT NOT NULL,

grupo\_actual\_id INT NOT NULL,

status ENUM('en\_curso', 'graduado') DEFAULT 'en\_curso',

fecha\_cambio DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (alumno\_id) REFERENCES usuarios(id),

FOREIGN KEY (grado\_actual\_id) REFERENCES grados(id),

FOREIGN KEY (grupo\_actual\_id) REFERENCES grupos(id)

);

-- Tabla de documentos

CREATE TABLE documentos (

id INT AUTO\_INCREMENT PRIMARY KEY,

alumno\_id INT NOT NULL,

nombre VARCHAR(255),

url VARCHAR(500),

tipo ENUM('constancia', 'boleta', 'otros'),

fecha\_subida DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (alumno\_id) REFERENCES usuarios(id)

);

-- Tabla de notificaciones

CREATE TABLE notificaciones (

id INT AUTO\_INCREMENT PRIMARY KEY,

usuario\_id INT NOT NULL,

titulo VARCHAR(255),

mensaje TEXT,

leido BOOLEAN DEFAULT 0,

fecha\_envio DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (usuario\_id) REFERENCES usuarios(id)

);

-- Tabla de antecedentes escolares

CREATE TABLE antecedentes (

id INT AUTO\_INCREMENT PRIMARY KEY,

alumno\_id INT NOT NULL,

instituto\_origen\_id INT NOT NULL,

nivel ENUM('primaria', 'secundaria', 'preparatoria'),

observaciones TEXT,

logros TEXT,

reportes TEXT,

fecha\_inicio DATE,

fecha\_fin DATE,

FOREIGN KEY (alumno\_id) REFERENCES usuarios(id),

FOREIGN KEY (instituto\_origen\_id) REFERENCES institutos(id)

);

-- Tabla de logros y reportes internos

CREATE TABLE logros\_reportes (

id INT AUTO\_INCREMENT PRIMARY KEY,

alumno\_id INT NOT NULL,

instituto\_id INT NOT NULL,

tipo ENUM('logro', 'reporte'),

descripcion TEXT,

fecha DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (alumno\_id) REFERENCES usuarios(id),

FOREIGN KEY (instituto\_id) REFERENCES institutos(id)

);