DOCUMENTACIÓN DE BASE DE DATOS

# Sistema MEDISYNC

Estructura completa de la base de datos SQLite  
Generado el: 12/08/2025 17:22  
Base de datos: medisync.db  
Motor: SQLite 3

# RESUMEN DE LA BASE DE DATOS

La base de datos MEDISYNC utiliza SQLite como motor de base de datos relacional. Está diseñada para gestionar de manera integral todos los aspectos de una clínica médica, incluyendo usuarios, pacientes, doctores, citas médicas, historiales clínicos y facturación.  
  
CARACTERÍSTICAS PRINCIPALES:  
• Motor: SQLite 3 (base de datos embebida)  
• Archivo: medisync.db  
• Ubicación: /database/medisync.db  
• Codificación: UTF-8  
• Integridad referencial: Habilitada  
• Transacciones ACID: Soportadas  
  
DISEÑO DE LA BASE DE DATOS:  
• Arquitectura relacional normalizada  
• Claves primarias auto-incrementales  
• Relaciones con claves foráneas  
• Índices para optimización de consultas  
• Timestamps automáticos  
• Validaciones a nivel de base de datos

# DIAGRAMA ENTIDAD-RELACIÓN CONCEPTUAL

ENTIDADES PRINCIPALES Y SUS RELACIONES:  
  
📊 USUARIOS (Entidad padre)  
 ├── PACIENTES (Especialización)  
 ├── DOCTORES (Especialización)   
 ├── ADMINISTRADORES (Especialización)  
 └── SECRETARIAS (Especialización)  
  
🏥 CITAS MÉDICAS  
 ├── Pertenece a → PACIENTE (1:N)  
 ├── Asignada a → DOCTOR (1:N)  
 └── Genera → FACTURA (1:1)  
  
📋 HISTORIALES MÉDICOS  
 ├── Pertenece a → PACIENTE (1:N)  
 └── Creado por → DOCTOR (1:N)  
  
💰 FACTURAS  
 ├── Emitida a → PACIENTE (1:N)  
 ├── Generada por → DOCTOR (1:N)  
 └── Relacionada con → CITA (1:1)  
  
🛡️ SEGUROS MÉDICOS  
 └── Asignado a → PACIENTES (1:N)  
  
RELACIONES CLAVE:  
• Un PACIENTE puede tener múltiples CITAS  
• Un DOCTOR puede atender múltiples CITAS  
• Una CITA genera una FACTURA  
• Un PACIENTE tiene múltiples HISTORIALES MÉDICOS  
• Un DOCTOR crea múltiples HISTORIALES MÉDICOS  
• Un PACIENTE puede tener un SEGURO MÉDICO

# ÍNDICE DE TABLAS

|  |  |  |  |
| --- | --- | --- | --- |
| Tabla | Propósito | Columnas | Registros |
| usuarios | Gestión de todos los usuarios del sistema | 13 | 24 |
| sqlite\_sequence | Tabla del sistema | 2 | 13 |
| doctores | Información específica de doctores | 10 | 7 |
| pacientes | Información específica de pacientes | 11 | 13 |
| secretarias | Tabla del sistema | 5 | 2 |
| citas | Gestión de citas médicas | 12 | 163 |
| historiales\_medicos | Historiales clínicos (legacy) | 12 | 46 |
| facturas | Sistema de facturación | 20 | 71 |
| pagos | Tabla del sistema | 8 | 16 |
| configuraciones | Tabla del sistema | 6 | 0 |
| auditoria | Tabla del sistema | 9 | 0 |
| horarios\_medicos | Tabla del sistema | 7 | 11 |
| seguros\_medicos | Catálogo de seguros médicos | 6 | 5 |
| tarifas\_doctores | Tabla del sistema | 11 | 7 |
| seguros | Tabla del sistema | 4 | 6 |
| historial\_medico | Historiales clínicos mejorados | 15 | 36 |
| billing\_config | Tabla del sistema | 4 | 0 |
| billing\_services | Tabla del sistema | 6 | 5 |
| servicios\_medicos | Tabla del sistema | 8 | 16 |
| facturas\_detalle | Tabla del sistema | 6 | 165 |

# DOCUMENTACIÓN DETALLADA DE TABLAS

## 🗃️ TABLA: USUARIOS

|  |  |
| --- | --- |
| Nombre de la tabla | usuarios |
| Número de columnas | 13 |
| Número de registros | 24 |
| Claves foráneas | 0 |

PROPÓSITO: Tabla principal que almacena información básica de todos los usuarios del sistema (administradores, doctores, secretarias y pacientes). Actúa como tabla padre para especialización por tipo de usuario.

### Estructura de Columnas - usuarios

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Columna | Tipo | Nulo | Defecto | PK | Descripción |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| id | INTEGER | SÍ |  | SÍ | Identificador único del usuario |
| nombre | VARCHAR(100) | NO |  |  | Nombre(s) del usuario |
| apellido | VARCHAR(100) | NO |  |  | Apellido(s) del usuario |
| email | VARCHAR(150) | NO |  |  | Correo electrónico único para autenticación |
| telefono | VARCHAR(20) | SÍ |  |  | Número de teléfono de contacto |
| direccion | TEXT | SÍ |  |  | Dirección física del usuario |
| fecha\_nacimiento | DATE | SÍ |  |  | Fecha de nacimiento del usuario |
| tipo\_usuario | VARCHAR(20) | NO |  |  | Rol: admin, doctor, secretaria, paciente |
| password\_hash | VARCHAR(255) | NO |  |  | Contraseña encriptada con SHA256 |
| activo | BOOLEAN | SÍ | 1 |  | Estado del usuario (activo/inactivo) |
| fecha\_creacion | TIMESTAMP | SÍ | CURRENT\_TIMESTAMP |  | Timestamp de creación del registro |
| fecha\_actualizacion | TIMESTAMP | SÍ | CURRENT\_TIMESTAMP |  | Timestamp de última actualización |
| ultimo\_acceso | TEXT | SÍ |  |  | Campo del sistema |

#### Definición SQL - usuarios

CREATE TABLE usuarios (  
id INTEGER PRIMARY KEY AUTOINCREMENT,  
 nombre TEXT NOT NULL,  
 apellido TEXT NOT NULL,  
 email TEXT UNIQUE NOT NULL,  
 telefono TEXT,  
 direccion TEXT,  
 fecha\_nacimiento DATE,  
 tipo\_usuario TEXT NOT NULL,  
 password\_hash TEXT NOT NULL,  
 activo BOOLEAN DEFAULT 1,  
 fecha\_creacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,  
 fecha\_actualizacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP  
);

## 🗃️ TABLA: PACIENTES

|  |  |
| --- | --- |
| Nombre de la tabla | pacientes |
| Número de columnas | 11 |
| Número de registros | 13 |
| Claves foráneas | 2 |

PROPÓSITO: Extiende la tabla usuarios con información específica de pacientes: expediente médico, tipo de sangre, alergias, contactos de emergencia y seguro médico.

### Estructura de Columnas - pacientes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Columna | Tipo | Nulo | Defecto | PK | Descripción |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| id | INTEGER | SÍ |  | SÍ | FK a usuarios.id - Identificador del paciente |
| numero\_expediente | VARCHAR(50) | SÍ |  |  | Número único de expediente médico |
| tipo\_sangre | VARCHAR(5) | SÍ |  |  | Grupo sanguíneo del paciente |
| alergias | TEXT | SÍ |  |  | Alergias conocidas del paciente |
| contacto\_emergencia | VARCHAR(100) | SÍ |  |  | Nombre del contacto de emergencia |
| telefono\_emergencia | VARCHAR(20) | SÍ |  |  | Teléfono del contacto de emergencia |
| seguro\_medico | VARCHAR(100) | SÍ |  |  | Nombre del seguro médico |
| seguro\_medico\_id | INTEGER | SÍ |  |  | FK a seguros\_medicos.id |
| tiene\_seguro | BOOLEAN | SÍ | 0 |  | Indica si tiene cobertura de seguro |
| numero\_seguro | TEXT | SÍ |  |  | Campo del sistema |
| porcentaje\_cobertura | REAL | SÍ | 0.0 |  | Campo del sistema |

#### Claves Foráneas - pacientes

|  |  |  |  |
| --- | --- | --- | --- |
| Columna Local | Tabla Referenciada | Columna Referenciada | Acción |
|  |  |  |  |
|  |  |  |  |
| seguro\_medico\_id | seguros\_medicos | id | ON UPDATE NO ACTION, ON DELETE NO ACTION |
| id | usuarios | id | ON UPDATE NO ACTION, ON DELETE CASCADE |

#### Definición SQL - pacientes

CREATE TABLE pacientes (  
id INTEGER PRIMARY KEY,  
 numero\_expediente TEXT UNIQUE,  
 tipo\_sangre TEXT,  
 alergias TEXT,  
 contacto\_emergencia TEXT,  
 telefono\_emergencia TEXT,  
 seguro\_medico TEXT,  
 seguro\_medico\_id INTEGER DEFAULT 4,  
 tiene\_seguro BOOLEAN DEFAULT 0,  
 FOREIGN KEY (id) REFERENCES usuarios(id)  
);

## 🗃️ TABLA: DOCTORES

|  |  |
| --- | --- |
| Nombre de la tabla | doctores |
| Número de columnas | 10 |
| Número de registros | 7 |
| Claves foráneas | 1 |

PROPÓSITO: Extiende la tabla usuarios con información específica de doctores: especialidad, cédula profesional, tarifas de consulta y horarios de atención.

### Estructura de Columnas - doctores

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Columna | Tipo | Nulo | Defecto | PK | Descripción |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| id | INTEGER | SÍ |  | SÍ | FK a usuarios.id - Identificador del doctor |
| especialidad | VARCHAR(100) | NO |  |  | Especialidad médica del doctor |
| cedula\_profesional | VARCHAR(50) | NO |  |  | Número único de cédula profesional |
| horario\_inicio | TIME | SÍ | '08:00' |  | Hora de inicio de atención |
| horario\_fin | TIME | SÍ | '18:00' |  | Hora de fin de atención |
| consultorio | VARCHAR(50) | SÍ |  |  | Campo del sistema |
| tarifa | DECIMAL(10,2) | SÍ | 500.00 |  | Campo del sistema |
| acepta\_seguros | BOOLEAN | SÍ | 1 |  | Indica si acepta seguros médicos |
| tarifa\_consulta\_base | REAL | SÍ | 1000.0 |  | Campo del sistema |
| tarifa\_consulta | DECIMAL(10,2) | SÍ | 500.00 |  | Tarifa por consulta médica |

#### Claves Foráneas - doctores

|  |  |  |  |
| --- | --- | --- | --- |
| Columna Local | Tabla Referenciada | Columna Referenciada | Acción |
|  |  |  |  |
| id | usuarios | id | ON UPDATE NO ACTION, ON DELETE CASCADE |

#### Definición SQL - doctores

CREATE TABLE doctores (  
id INTEGER PRIMARY KEY,  
 especialidad TEXT,  
 cedula\_profesional TEXT UNIQUE,  
 acepta\_seguros BOOLEAN DEFAULT 1,  
 tarifa\_consulta DECIMAL(10,2) DEFAULT 500.00,  
 horario\_inicio TIME DEFAULT '08:00',  
 horario\_fin TIME DEFAULT '17:00',  
 FOREIGN KEY (id) REFERENCES usuarios(id)  
);

## 🗃️ TABLA: CITAS

|  |  |
| --- | --- |
| Nombre de la tabla | citas |
| Número de columnas | 12 |
| Número de registros | 163 |
| Claves foráneas | 2 |

PROPÓSITO: Gestiona las citas médicas del sistema, relacionando pacientes con doctores en fechas y horarios específicos. Incluye motivo, estado y observaciones.

### Estructura de Columnas - citas

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Columna | Tipo | Nulo | Defecto | PK | Descripción |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| id | INTEGER | SÍ |  | SÍ | Identificador único de la cita |
| paciente\_id | INTEGER | SÍ |  |  | FK a usuarios.id - Paciente de la cita |
| doctor\_id | INTEGER | SÍ |  |  | FK a usuarios.id - Doctor asignado |
| fecha\_hora | TIMESTAMP | NO |  |  | Fecha y hora programada de la cita |
| motivo | TEXT | NO |  |  | Motivo o razón de la consulta |
| estado | VARCHAR(20) | SÍ | 'pendiente' |  | Estado: pendiente, confirmada, en\_curso, completada, cancelada |
| notas | TEXT | SÍ |  |  | Observaciones adicionales de la cita |
| duracion\_minutos | INTEGER | SÍ | 60 |  | Duración estimada en minutos |
| fecha\_creacion | TIMESTAMP | SÍ | CURRENT\_TIMESTAMP |  | Timestamp de creación |
| fecha\_actualizacion | TIMESTAMP | SÍ | CURRENT\_TIMESTAMP |  | Timestamp de actualización |
| tarifa\_consulta | REAL | SÍ | 1000.0 |  | Tarifa aplicada a esta cita |
| seguro\_aplicable | BOOLEAN | SÍ | 0 |  | Indica si aplica descuento de seguro |

#### Claves Foráneas - citas

|  |  |  |  |
| --- | --- | --- | --- |
| Columna Local | Tabla Referenciada | Columna Referenciada | Acción |
|  |  |  |  |
|  |  |  |  |
| doctor\_id | usuarios | id | ON UPDATE NO ACTION, ON DELETE CASCADE |
| paciente\_id | usuarios | id | ON UPDATE NO ACTION, ON DELETE CASCADE |

#### Definición SQL - citas

CREATE TABLE citas (  
id INTEGER PRIMARY KEY AUTOINCREMENT,  
 paciente\_id INTEGER NOT NULL,  
 doctor\_id INTEGER NOT NULL,  
 fecha\_hora TIMESTAMP,  
 motivo TEXT,  
 estado VARCHAR(20) DEFAULT 'programada',  
 notas TEXT,  
 duracion\_minutos INTEGER DEFAULT 30,  
 fecha\_creacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,  
 fecha\_actualizacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,  
 tarifa\_consulta REAL,  
 seguro\_aplicable BOOLEAN DEFAULT 0,  
 FOREIGN KEY (paciente\_id) REFERENCES usuarios(id),  
 FOREIGN KEY (doctor\_id) REFERENCES usuarios(id)  
);

## 🗃️ TABLA: HISTORIAL\_MEDICO

|  |  |
| --- | --- |
| Nombre de la tabla | historial\_medico |
| Número de columnas | 15 |
| Número de registros | 36 |
| Claves foráneas | 2 |

PROPÓSITO: Versión mejorada de los historiales médicos que almacena consultas, diagnósticos, tratamientos y medicamentos de cada paciente.

### Estructura de Columnas - historial\_medico

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Columna | Tipo | Nulo | Defecto | PK | Descripción |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| id | INTEGER | SÍ |  | SÍ | Identificador único del historial |
| paciente\_id | INTEGER | NO |  |  | FK a usuarios.id - Paciente del historial |
| doctor\_id | INTEGER | NO |  |  | FK a usuarios.id - Doctor que atendió |
| fecha\_consulta | DATE | NO |  |  | Fecha de la consulta médica |
| tipo\_consulta | TEXT | SÍ | 'Consulta General' |  | Tipo de consulta realizada |
| motivo\_consulta | TEXT | SÍ |  |  | Motivo principal de la consulta |
| sintomas | TEXT | SÍ |  |  | Síntomas presentados por el paciente |
| diagnostico | TEXT | SÍ |  |  | Diagnóstico médico establecido |
| tratamiento | TEXT | SÍ |  |  | Tratamiento prescrito |
| medicamentos | TEXT | SÍ |  |  | Medicamentos recetados |
| observaciones | TEXT | SÍ |  |  | Observaciones adicionales |
| proxima\_cita | DATE | SÍ |  |  | Fecha sugerida para próxima cita |
| estado | TEXT | SÍ | 'Completada' |  | Estado del historial |
| fecha\_creacion | TIMESTAMP | SÍ | CURRENT\_TIMESTAMP |  | Timestamp de creación |
| fecha\_modificacion | TIMESTAMP | SÍ | CURRENT\_TIMESTAMP |  | Timestamp de modificación |

#### Claves Foráneas - historial\_medico

|  |  |  |  |
| --- | --- | --- | --- |
| Columna Local | Tabla Referenciada | Columna Referenciada | Acción |
|  |  |  |  |
|  |  |  |  |
| doctor\_id | usuarios | id | ON UPDATE NO ACTION, ON DELETE NO ACTION |
| paciente\_id | usuarios | id | ON UPDATE NO ACTION, ON DELETE NO ACTION |

#### Definición SQL - historial\_medico

CREATE TABLE historial\_medico (  
id INTEGER PRIMARY KEY AUTOINCREMENT,  
 paciente\_id INTEGER NOT NULL,  
 doctor\_id INTEGER NOT NULL,  
 fecha\_consulta DATE NOT NULL,  
 tipo\_consulta TEXT DEFAULT 'Consulta General',  
 motivo\_consulta TEXT,  
 sintomas TEXT,  
 diagnostico TEXT,  
 tratamiento TEXT,  
 medicamentos TEXT,  
 observaciones TEXT,  
 proxima\_cita DATE,  
 estado TEXT DEFAULT 'Completada',  
 fecha\_creacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,  
 fecha\_modificacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,  
 FOREIGN KEY (paciente\_id) REFERENCES usuarios(id),  
 FOREIGN KEY (doctor\_id) REFERENCES usuarios(id)  
);

## 🗃️ TABLA: HISTORIALES\_MEDICOS

|  |  |
| --- | --- |
| Nombre de la tabla | historiales\_medicos |
| Número de columnas | 12 |
| Número de registros | 46 |
| Claves foráneas | 3 |

PROPÓSITO: Versión legacy de historiales médicos. Mantiene compatibilidad con versiones anteriores del sistema.

### Estructura de Columnas - historiales\_medicos

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Columna | Tipo | Nulo | Defecto | PK | Descripción |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| id | INTEGER | SÍ |  | SÍ | Campo del sistema |
| paciente\_id | INTEGER | SÍ |  |  | Campo del sistema |
| doctor\_id | INTEGER | SÍ |  |  | Campo del sistema |
| cita\_id | INTEGER | SÍ |  |  | Campo del sistema |
| fecha\_consulta | DATE | NO |  |  | Campo del sistema |
| diagnostico | TEXT | NO |  |  | Campo del sistema |
| tratamiento | TEXT | SÍ |  |  | Campo del sistema |
| medicamentos | TEXT | SÍ |  |  | Campo del sistema |
| observaciones | TEXT | SÍ |  |  | Campo del sistema |
| adjuntos | TEXT | SÍ |  |  | Campo del sistema |
| estado | VARCHAR(20) | SÍ | 'activo' |  | Campo del sistema |
| fecha\_creacion | TIMESTAMP | SÍ | CURRENT\_TIMESTAMP |  | Campo del sistema |

#### Claves Foráneas - historiales\_medicos

|  |  |  |  |
| --- | --- | --- | --- |
| Columna Local | Tabla Referenciada | Columna Referenciada | Acción |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| cita\_id | citas | id | ON UPDATE NO ACTION, ON DELETE SET NULL |
| doctor\_id | usuarios | id | ON UPDATE NO ACTION, ON DELETE CASCADE |
| paciente\_id | usuarios | id | ON UPDATE NO ACTION, ON DELETE CASCADE |

#### Definición SQL - historiales\_medicos

CREATE TABLE historiales\_medicos (  
id INTEGER PRIMARY KEY AUTOINCREMENT,  
 paciente\_id INTEGER NOT NULL,  
 doctor\_id INTEGER NOT NULL,  
 fecha\_consulta DATE NOT NULL,  
 diagnostico TEXT NOT NULL,  
 tratamiento TEXT,  
 medicamentos TEXT,  
 observaciones TEXT,  
 estado TEXT DEFAULT 'activo',  
 fecha\_creacion TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,  
 FOREIGN KEY (paciente\_id) REFERENCES usuarios(id),  
 FOREIGN KEY (doctor\_id) REFERENCES usuarios(id)  
);

## 🗃️ TABLA: FACTURAS

|  |  |
| --- | --- |
| Nombre de la tabla | facturas |
| Número de columnas | 20 |
| Número de registros | 71 |
| Claves foráneas | 3 |

PROPÓSITO: Sistema de facturación que gestiona cobros por consultas y servicios médicos. Relaciona pacientes, doctores y métodos de pago.

### Estructura de Columnas - facturas

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Columna | Tipo | Nulo | Defecto | PK | Descripción |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| id | INTEGER | SÍ |  | SÍ | Identificador único de la factura |
| paciente\_id | INTEGER | SÍ |  |  | FK a usuarios.id - Paciente facturado |
| cita\_id | INTEGER | SÍ |  |  | Campo del sistema |
| numero\_factura | VARCHAR(50) | NO |  |  | Número único de factura |
| concepto | TEXT | NO |  |  | Descripción del servicio facturado |
| monto | DECIMAL(10,2) | NO |  |  | Monto total de la factura |
| estado | VARCHAR(20) | SÍ | 'pendiente' |  | Estado: pendiente, pagada, vencida |
| fecha\_creacion | DATE | SÍ | CURRENT\_DATE |  | Fecha de emisión de la factura |
| fecha\_vencimiento | DATE | NO |  |  | Fecha límite de pago |
| fecha\_pago | TIMESTAMP | SÍ |  |  | Fecha en que se realizó el pago |
| metodo\_pago | VARCHAR(50) | SÍ |  |  | Método utilizado para el pago |
| referencia\_pago | VARCHAR(100) | SÍ |  |  | Campo del sistema |
| notas | TEXT | SÍ |  |  | Campo del sistema |
| seguro\_aplicado | TEXT | SÍ |  |  | Campo del sistema |
| descuento\_seguro | REAL | SÍ | 0.0 |  | Campo del sistema |
| monto\_original | REAL | SÍ |  |  | Campo del sistema |
| monto\_descuento | REAL | SÍ | 0.0 |  | Campo del sistema |
| moneda | TEXT | SÍ | "RD$" |  | Campo del sistema |
| doctor\_id | INTEGER | SÍ |  |  | FK a usuarios.id - Doctor que prestó el servicio |
| tipo\_consulta | TEXT | SÍ | "consulta\_general" |  | Campo del sistema |

#### Claves Foráneas - facturas

|  |  |  |  |
| --- | --- | --- | --- |
| Columna Local | Tabla Referenciada | Columna Referenciada | Acción |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
| doctor\_id | usuarios | id | ON UPDATE NO ACTION, ON DELETE NO ACTION |
| cita\_id | citas | id | ON UPDATE NO ACTION, ON DELETE SET NULL |
| paciente\_id | usuarios | id | ON UPDATE NO ACTION, ON DELETE CASCADE |

#### Definición SQL - facturas

CREATE TABLE facturas (  
id INTEGER PRIMARY KEY AUTOINCREMENT,  
 numero\_factura TEXT UNIQUE NOT NULL,  
 paciente\_id INTEGER NOT NULL,  
 doctor\_id INTEGER,  
 concepto TEXT NOT NULL,  
 monto DECIMAL(10,2) NOT NULL,  
 estado TEXT DEFAULT 'pendiente',  
 fecha\_creacion DATE NOT NULL,  
 fecha\_vencimiento DATE NOT NULL,  
 fecha\_pago DATETIME,  
 metodo\_pago TEXT,  
 FOREIGN KEY (paciente\_id) REFERENCES usuarios(id),  
 FOREIGN KEY (doctor\_id) REFERENCES usuarios(id)  
);

## 🗃️ TABLA: SEGUROS\_MEDICOS

|  |  |
| --- | --- |
| Nombre de la tabla | seguros\_medicos |
| Número de columnas | 6 |
| Número de registros | 5 |
| Claves foráneas | 0 |

PROPÓSITO: Catálogo de seguros médicos disponibles con porcentajes de descuento y descripciones.

### Estructura de Columnas - seguros\_medicos

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Columna | Tipo | Nulo | Defecto | PK | Descripción |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| id | INTEGER | SÍ |  | SÍ | Identificador único del seguro |
| nombre | TEXT | NO |  |  | Nombre del seguro médico |
| descuento\_porcentaje | REAL | NO |  |  | Porcentaje de descuento aplicable |
| descripcion | TEXT | SÍ |  |  | Descripción del plan de seguro |
| activo | BOOLEAN | SÍ | 1 |  | Estado del seguro (activo/inactivo) |
| fecha\_creacion | DATETIME | SÍ | CURRENT\_TIMESTAMP |  | Campo del sistema |

#### Definición SQL - seguros\_medicos

CREATE TABLE seguros\_medicos (  
id INTEGER PRIMARY KEY AUTOINCREMENT,  
 nombre TEXT UNIQUE NOT NULL,  
 descuento\_porcentaje DECIMAL(5,2) DEFAULT 0,  
 descripcion TEXT,  
 activo BOOLEAN DEFAULT 1  
);

# RELACIONES Y RESTRICCIONES

RELACIONES PRINCIPALES:  
  
1. USUARIOS → PACIENTES (1:1)  
 - usuarios.id = pacientes.id  
 - Un usuario de tipo 'paciente' tiene un registro en pacientes  
  
2. USUARIOS → DOCTORES (1:1)  
 - usuarios.id = doctores.id  
 - Un usuario de tipo 'doctor' tiene un registro en doctores  
  
3. USUARIOS → CITAS (1:N como paciente)  
 - usuarios.id = citas.paciente\_id  
 - Un paciente puede tener múltiples citas  
  
4. USUARIOS → CITAS (1:N como doctor)  
 - usuarios.id = citas.doctor\_id  
 - Un doctor puede atender múltiples citas  
  
5. USUARIOS → HISTORIAL\_MEDICO (1:N como paciente)  
 - usuarios.id = historial\_medico.paciente\_id  
 - Un paciente tiene múltiples historiales  
  
6. USUARIOS → HISTORIAL\_MEDICO (1:N como doctor)  
 - usuarios.id = historial\_medico.doctor\_id  
 - Un doctor crea múltiples historiales  
  
7. USUARIOS → FACTURAS (1:N como paciente)  
 - usuarios.id = facturas.paciente\_id  
 - Un paciente puede tener múltiples facturas  
  
8. SEGUROS\_MEDICOS → PACIENTES (1:N)  
 - seguros\_medicos.id = pacientes.seguro\_medico\_id  
 - Un seguro puede cubrir múltiples pacientes  
  
RESTRICCIONES DE INTEGRIDAD:  
• Todos los emails en usuarios deben ser únicos  
• Los números de expediente en pacientes deben ser únicos  
• Las cédulas profesionales en doctores deben ser únicas  
• Los números de factura deben ser únicos  
• No se pueden eliminar usuarios referenciados en otras tablas  
• Las fechas de citas no pueden ser en el pasado (validación de aplicación)  
• Los montos de facturas deben ser positivos

# ÍNDICES Y OPTIMIZACIÓN

ÍNDICES AUTOMÁTICOS:  
• PRIMARY KEY automáticamente indexadas  
• UNIQUE constraints automáticamente indexadas  
• emails únicos en tabla usuarios  
• números de expediente únicos en pacientes  
• cédulas profesionales únicas en doctores  
• números de factura únicos  
  
ÍNDICES RECOMENDADOS PARA OPTIMIZACIÓN:  
• usuarios.tipo\_usuario (para filtros por rol)  
• citas.fecha\_hora (para búsquedas por fecha)  
• citas.estado (para filtros por estado)  
• facturas.estado (para facturas pendientes)  
• historial\_medico.fecha\_consulta (para historiales cronológicos)  
  
CONSULTAS OPTIMIZADAS:  
• Búsqueda de usuarios por email (índice único)  
• Citas por doctor y fecha (claves foráneas)  
• Historiales por paciente (clave foránea)  
• Facturas pendientes por paciente (índice compuesto recomendado)  
  
ESTRATEGIAS DE MANTENIMIENTO:  
• VACUUM periódico para desfragmentación  
• ANALYZE para actualizar estadísticas  
• Backup incremental diario  
• Verificación de integridad referencial

# SEGURIDAD Y RESPALDOS

MEDIDAS DE SEGURIDAD:  
  
AUTENTICACIÓN:  
• Contraseñas encriptadas con SHA256  
• Validación de credenciales en cada sesión  
• Control de acceso basado en roles (RBAC)  
• Sesiones con timeout automático  
  
INTEGRIDAD DE DATOS:  
• Transacciones ACID garantizadas  
• Validaciones a nivel de aplicación  
• Restricciones de claves foráneas  
• Verificación de tipos de datos  
  
RESPALDOS:  
• Ubicación: /database/  
• Formato: SQLite .db  
• Frecuencia recomendada: Diaria  
• Retención: 30 días mínimo  
• Verificación de integridad post-backup  
  
CONSIDERACIONES DE PRIVACIDAD:  
• Datos médicos sensibles protegidos  
• Acceso limitado por rol de usuario  
• Logs de auditoría recomendados  
• Cumplimiento con regulaciones de salud  
  
RECOMENDACIONES DE SEGURIDAD:  
• Implementar cifrado de archivo de BD  
• Auditoría de accesos y modificaciones  
• Backup en ubicación segura  
• Rotación regular de contraseñas  
• Validación de entrada estricta

# CONCLUSIONES Y RECOMENDACIONES

FORTALEZAS DEL DISEÑO:  
✅ Arquitectura relacional normalizada  
✅ Integridad referencial bien definida  
✅ Separación clara de responsabilidades  
✅ Escalabilidad para clínicas pequeñas-medianas  
✅ Estructura flexible para extensiones futuras  
  
ÁREAS DE MEJORA:  
🔶 Migrar a PostgreSQL para mayor robustez  
🔶 Implementar índices adicionales para optimización  
🔶 Agregar triggers para auditoría automática  
🔶 Implementar particionamiento por fecha  
🔶 Cifrado a nivel de base de datos  
  
RECOMENDACIONES TÉCNICAS:  
📋 Monitoreo de performance de consultas  
📋 Implementar versionado de esquema  
📋 Documentar procedimientos de migración  
📋 Establecer políticas de retención de datos  
📋 Configurar alertas de espacio en disco  
  
MANTENIMIENTO REQUERIDO:  
🔧 VACUUM semanal para optimización  
🔧 Backup diario automatizado  
🔧 Verificación de integridad mensual  
🔧 Análisis de crecimiento de datos  
🔧 Revisión de performance trimestral