# Informe Integrador - Base de Datos para Clínica Médica

Materia: Bases de Datos

Profesor: Nicolás Quiros

Integrantes: Flores Leandro, Aciar Agustin, Vera Santiago, Ramirez Rodrigo, Franco Camila.

## 1. Diseño conceptual y lógico

### Tablas principales

#### Tabla: **paciente**

* Clave primaria: id\_paciente
* Claves foráneas: id\_obra\_social → obra\_social.id\_obra\_social
* Restricciones: dni y email son UNIQUE, nombre, apellido, dni y fecha\_nac son NOT NULL
* Forma normal: 3FN (todos los atributos dependen solo de la clave primaria y no entre ellos)
* Tipo: Dimensión (describe entidades, no hechos)

#### Tabla: **obra\_social**

* Clave primaria: id\_obra\_social
* Restricciones: nombre es UNIQUE y NOT NULL
* Forma normal: 3FN
* Tipo: Dimensión

#### Tabla: **medico**

* Clave primaria: id\_medico
* Clave foránea: id\_especialidad → especialidad.id\_especialidad
* Restricciones: matricula es UNIQUE, todos los campos son NOT NULL
* Forma normal: 3FN
* Tipo: Dimensión

#### 

#### Tabla: **especialidad**

* Clave primaria: id\_especialidad
* Restricciones: nombre es UNIQUE y NOT NULL
* Forma normal: 3FN
* Tipo: Dimensión

#### Tabla: **turno**

* Clave primaria: id\_turno
* Claves foráneas: id\_paciente, id\_medico, id\_secretario, id\_estado, id\_sede
* Restricciones: fecha, hora, estado son NOT NULL, estado con CHECK si no se normaliza
* Forma normal: 3FN
* Tipo: Hecho (representa acciones medibles)

#### Tabla: **estado**

* Clave primaria: id\_estado
* Restricciones: nombre es UNIQUE y NOT NULL
* Tipo: Dimensión (catálogo para evitar errores y normalizar)

#### Tabla: **secretario**

* Clave primaria: id\_secretario
* Clave foránea: id\_sede
* Tipo: Dimensión

#### Tabla: **sede**

* Clave primaria: id\_sede
* Tipo: Dimensión

#### Tabla: **enfermedades\_cronicas**

* Clave primaria: id\_enfermedad
* Tipo: Dimensión

#### Tabla: **pacientes\_enfermedades\_cronicas**

* Clave primaria compuesta: (id\_paciente, id\_enfermedad)
* Tipo: Relación N:M

#### Tabla: **evoluciones\_clinicas**

* Clave primaria: id\_evolucion
* Tipo: Hecho (seguimiento médico)

## 2. Modelo de hechos y dimensiones

* Hecho central: **fact\_turno**
* Dimensiones: **dim\_paciente**, **dim\_medico**, **dim\_especialidad**, **dim\_sede**, **dim\_fecha**, **dim\_estado**
* Relaciones: 1:N desde cada dimensión a **fact\_turno**
* Tipo de esquema: Estrella (dimensiones desnormalizadas)

## 3. SQL de creación de dos tablas

CREATE TABLE paciente (

id\_paciente INT PRIMARY KEY AUTO\_INCREMENT,

nombre VARCHAR(50) NOT NULL,

apellido VARCHAR(50) NOT NULL,

dni VARCHAR(20) UNIQUE NOT NULL,

email VARCHAR(100) UNIQUE NOT NULL,

fecha\_nac DATE NOT NULL,

id\_obra\_social INT REFERENCES obra\_social(id\_obra\_social)

);

CREATE TABLE turno (

id\_turno INT PRIMARY KEY AUTO\_INCREMENT,

id\_paciente INT REFERENCES paciente(id\_paciente),

id\_medico INT REFERENCES medico(id\_medico),

id\_secretario INT REFERENCES secretario(id\_secretario),

id\_estado INT REFERENCES estado(id\_estado),

id\_sede INT REFERENCES sede(id\_sede),

fecha DATE NOT NULL,

hora TIME NOT NULL,

motivo TEXT

);

## 

## 4. Vistas

### Vista para médicos

CREATE VIEW vw\_medico\_pacientes AS

SELECT \* FROM paciente;

### Vista de evoluciones Clínicas para médicos

CREATE VIEW evoluciones\_para\_medicos AS

SELECT

ec.id\_evolucion,

p.dni,

p.historia\_clinica,

p.apellido,

p.nombre,

p.fecha\_nacimiento

ec.fecha AS fecha\_evolucion,

ec.hora,

ec.motivo\_consulta,

ec.diagnostico,

ec.tratamiento,

ec.notas\_adicionales

FROM

evolucion\_clinica AS ec

LEFT JOIN

paciente AS p

ON

ec.id\_paciente = p.id\_paciente;

LEFT JOIN

obra social AS os

ON

os.id\_obra\_social = ec.id\_obra\_social

### Vista evoluciones clínicas para secretarios

CREATE VIEW evoluciones\_para\_administrativos AS

SELECT

ec.id\_evolucion

p.dni,

p.apellido,

p.nombre,

p.fecha\_nacimiento,

p.telefono,

p.mail,

os.nombre AS obra\_social,

p.numero\_afiliado

ec.fecha AS fecha\_evolucion,

ec.hora,

ec.motivo\_consulta,

ec.diagnostico

FROM

evolucion\_clinica AS ec

LEFT JOIN

paciente AS p

ON

ec.id\_paciente = p.id\_paciente;

LEFT JOIN

obra social AS os

ON

os.id\_obra\_social = ec.id\_obra\_social;

### Vista para secretarios

CREATE VIEW vw\_secretario\_pacientes AS

SELECT

id\_paciente,

nombre,

apellido,

dni,

email,

fecha\_nac,

id\_obra\_social

FROM

paciente;

## Justificación

Las vistas permiten controlar el acceso a la información sensible (como tratamiento, patologias y notas adicionales medicas) dependiendo del rol del usuario, evitando errores o accesos indebidos a las tablas base.