



U

P

Tarea

Modelado lógico del caso "Hospital"

Axel David Pérez Hernández, Guadalupe Joan Anaya Cárdenas, Alejandro Yahir
Mérida Cruz, Ángel Yahir Martínez Gómez, Alexis Yahir Villar Escorcía

Universidad Politécnica De Tulancingo, Calle Ingenierías #100, Huapalcalco,
Tulancingo

Base de Datos

Nombre del profesor: Víctor Hugo Fernández Cruz

Tulancingo Hidalgo 11 de abril de 2024



índice

| | |
|--|----|
| Resumen | 4 |
| Introducción | 5 |
| Caso | 6 |
| Lista de Entidades..... | 6 |
| Lista de atributos | 6 |
| Relaciones..... | 8 |
| Grafo binario..... | 9 |
| Grafos sin redundancia | 9 |
| Matriz de relación | 9 |
| Primera Propuesta de Diagrama MER | 10 |
| Diagrama MER Según Análisis Lógico | 11 |
| Diccionario de datos..... | 12 |
| Esquema de transición..... | 13 |
| Modelo ER..... | 14 |
| Descripción de 3 reglas formales | 14 |
| Primera forma normal | 14 |
| Segunda forma normal..... | 15 |
| Tercera forma normal | 15 |
| Código | 15 |
| Datos de registro | 22 |

| | |
|--|----|
| Joins | 25 |
| 1. Obtener el nombre del paciente y la fecha de su próxima cita médica: | 25 |
| 2. Mostrar el nombre del médico y la especialidad de cada cita médica: | 25 |
| 3. Listar el historial médico de un paciente junto con su nombre: | 26 |
| 4. Mostrar los tratamientos activos de un paciente y el nombre del médico responsable: | 26 |
| Conclusiones | 27 |
| Referencias de las ilustraciones | 28 |

Resumen

Este proyecto tiene como objetivo desarrollar una base de datos relacional para gestionar un sistema de gestión de pacientes para un hospital. Se necesita almacenar información detallada sobre los pacientes, citas médicas, resultados de pruebas, medicamentos recetados, tratamientos recibidos, médicos y personal de atención médica. El diseño de la base de datos seguirá los principios de modelado conceptual, incluyendo un Modelo de Entidad-Relación (MER), diccionario de datos, diagrama de transición y un Modelo de Entidad-Relación (ER). Además, se explicarán las tres formas normales y se proporcionarán ejemplos de datos de registro y consultas de prueba.

Introducción

El presente proyecto se centra en el diseño y desarrollo de una base de datos relacional para un hospital, con el objetivo de gestionar un sistema integral de gestión de pacientes. Este sistema abarca desde la información básica de los pacientes hasta los registros detallados de su historial médico, tratamientos recibidos, medicamentos recetados y citas médicas programadas. Además, se incluye un seguimiento de los médicos y personal de atención médica, junto con detalles sobre sus especialidades, horarios de trabajo y notas de progreso de los pacientes.

Para lograr este objetivo, se seguirá un proceso estructurado que incluye la creación de un Modelo de Entidad-Relación (MER), un diccionario de datos, un diagrama de transición y un Modelo de Entidad-Relación (ER). Además, se abordarán conceptos clave de normalización de bases de datos, como la primera, segunda y tercera forma normal, para garantizar la integridad y eficiencia de la base de datos.

A lo largo de este proyecto, se explorarán diversos aspectos del diseño y desarrollo de bases de datos relacionales, con el objetivo final de proporcionar una solución efectiva y escalable para la gestión de pacientes en un entorno hospitalario.

Caso

Se necesita desarrollar una base de datos relacional para gestionar un sistema de gestión de pacientes para un hospital. La base de datos debe incluir información detallada sobre los pacientes, como nombres, números de seguro médico, historiales médicos, medicamentos recetados y tratamientos recibidos. Además, se requiere un seguimiento de las citas médicas programadas, así como los resultados de pruebas y procedimientos médicos. También se necesita un registro de los médicos y personal de atención médica, incluyendo especialidades, horarios de trabajo y notas de progreso de los pacientes.

Lista de Entidades

- Paciente
- Médico
- Personal Médico
- Cita Médica
- Historial Médico
- Medicamento
- Tratamiento

Lista de atributos

Paciente:

- ID_Paciente (PK)

- Nombre
- Número de Seguro Médico
- Fecha de Nacimiento
- Género
- Teléfono

Médico:

- ID_Médico (PK)
- Nombre
- Especialidad
- Teléfono

Personal Médico:

- ID_Personal (PK)
- Nombre
- Cargo
- Teléfono

Cita Médica:

- ID_Cita (PK)
- Fecha y Hora
- ID_Paciente (FK)
- ID_Médico (FK)

Historial Médico:

- ID_Historial (PK)
- ID_Paciente (FK)

- Descripción
- Fecha

Medicamento:

- ID_Medicamento (PK)
- Nombre
- Descripción

Tratamiento:

- ID_Tratamiento (PK)
- ID_Paciente (FK)
- ID_Médico (FK)
- Descripción
- Fecha de Inicio
- Fecha de Fin

Relaciones

Paciente - Cita Médica (1:N)

Paciente - Historial Médico (1:N)

Paciente - Tratamiento (1:N)

Médico - Cita Médica (1:N)

Médico - Tratamiento (1:N)

Paciente - Medicamento(M:N)

Medico – Personal Medico(1:1)

Grafo binario

Paciente -> Cita Médica

Paciente -> Historial Médico

Paciente -> Tratamiento

Médico -> Cita Médica

Médico -> Tratamiento

Grafos sin redundancia

Paciente -> Cita Médica, Historial Médico, Tratamiento

Médico -> Cita Médica, Tratamiento

Matriz de relación

| | Cita Medica | Historial Medico | Tratamiento |
|----------|-------------|------------------|-------------|
| Paciente | 1 | 1 | 1 |
| Medico | 1 | 0 | 1 |

Primera Propuesta de Diagrama MER

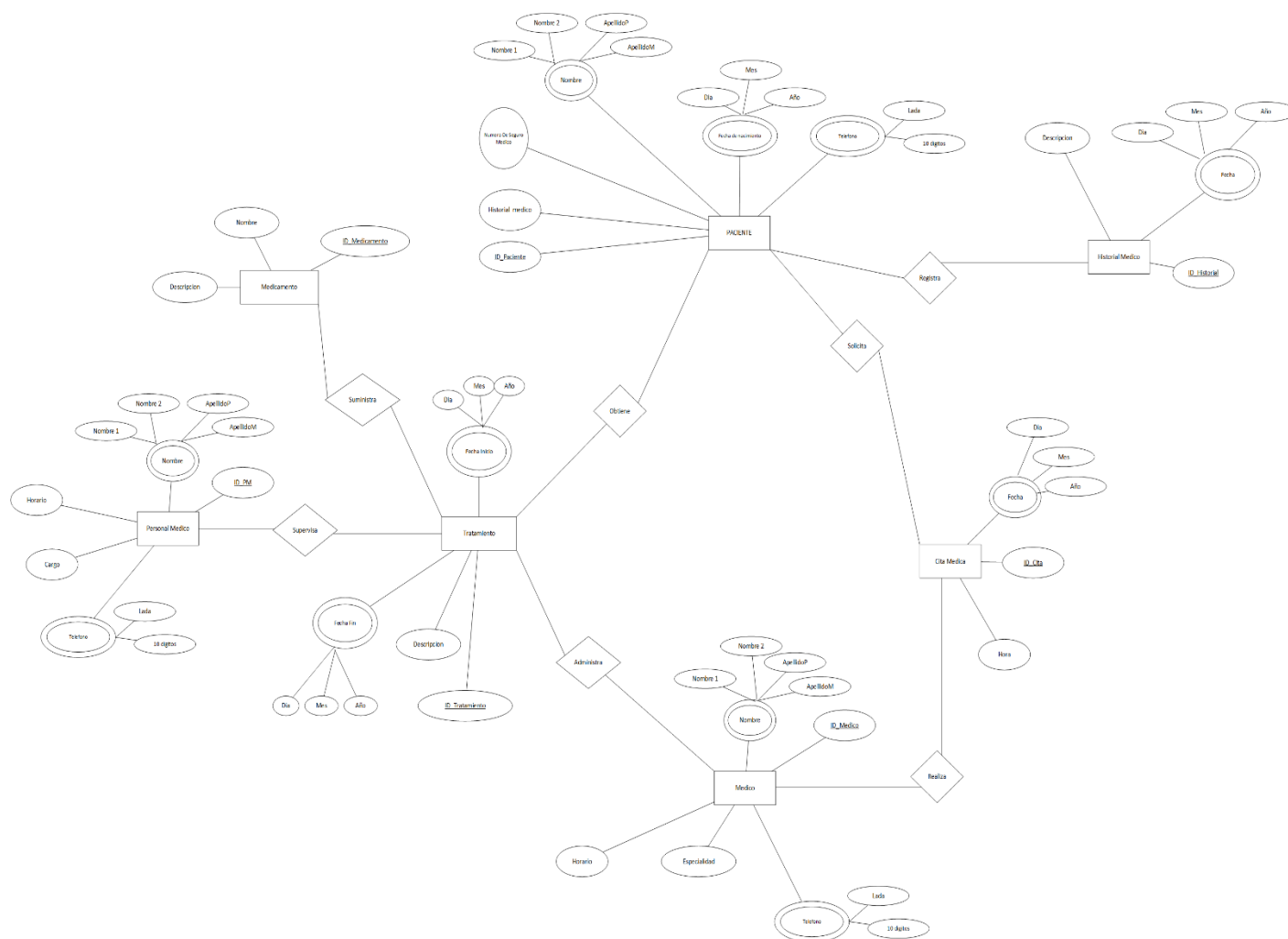


Ilustración 1 Primer Propuesta Diagrama MER del caso Hospital

Diagrama MER Según Análisis Lógico

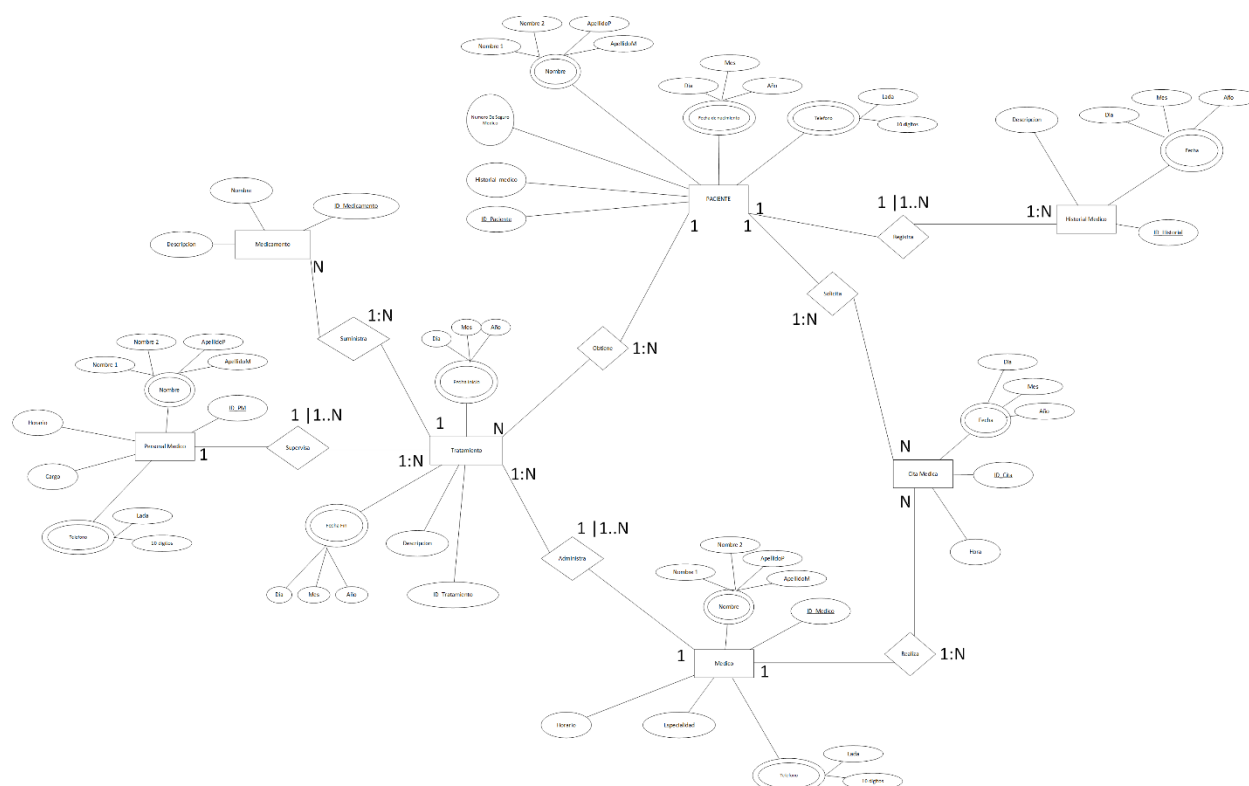


Ilustración 2 Diagrama MER del caso Hospital

Diccionario de datos

| Inicio del formato | | | | | | | | | | |
|---|-------------------------|--|---|--|-------------------|---|--|---|---|---|
| HOSPITAL | | | Preparado por: | | Grupo ISC53 | | | Fecha de preparación: | | 16/04/2024 |
| 1. Identificación del archivo de datos | | | | | | | | 2. Reglas de validación | | |
| Descripción del Archivo | Nombre del Campo | Descripción del Campo | Llave Primaria | Llave Foránea | Campo obligatorio | Dominio o lista de valores | Tipo de datos | Longitud del campo | Regla de validación (en lenguaje natural) | Observaciones |
| | | | [Indicar si el campo hace parte de la llave primaria] | [Si el campo hace parte de una llave foránea, indique el archivo y el campo de referencia] | [SI/NO] | [Liste los valores válidos, patrón o rango para el campo] | [Numérico, Cadena, Carácter, Entero, otro] | [Según el tipo de dato indicar la longitud del campo] | [Enumere claramente cada regla que se debe aplicar a la Variable] | [Establecer observaciones acerca de la especificación del registro] |
| Se muestran las tablas y atributos que usaremos en la base de datos para el caso de el hospital | ID_Paciente | Identificador único del paciente. | SI | | si | numeros | INT(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | Nombre | Nombre completo del paciente. | | | si | letras | VARCHAR(30) | 30 | Campo requerido, NO DEBE SER NULO. | |
| | Número_de_Seguro_Médico | Identificación personal para servicios de salud. | | | no | letras | VARCHAR(20) | 20 | Campo opcional, PUEDE SER NULO. | |
| | Fecha_de_Nacimiento | Fecha del naimiento del paciente. | | | si | fecha | DATETIME | 20 | Campo requerido, NO DEBE SER NULO. | |
| | Género | Designación de sexo | | | si | letras | VARCHAR(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | Dirección | Ubicación física de una residencia. | | | si | letras | VARCHAR(200) | 200 | Campo requerido, NO DEBE SER NULO. | |
| | Teléfono | Número de contacto telefónico | | | si | numeros | INT(20) | 20 | Campo requerido, NO DEBE SER NULO. | |
| | ID_Cita | Identificador único de la cita médica. | SI | | si | numeros | INT(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | Fecha_y_Hora: | Datos del tiempo de programación. | | | si | fecha | DATETIME | 20 | Campo no requerido, PUEDE SER NULO. | |
| | ID_Paciente | Identificador del paciente y su cita. | | SI TABLA CITA_MEDICA | si | numeros | INT(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | ID_Médico | Identificador del médico y su cita. | | SI TABLA CITA_MEDICA | si | numeros | INT(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | ID_Historial | Identificador único del historial. | SI | | si | numeros | INT(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | ID_Paciente | Identificador del paciente y historial. | | SI TABLA | si | numeros | INT(10) | 10 | Campo no requerido, PUEDE SER NULO. | |
| | Descripción | Explicación breve del historial. | | | no | texto | TEXT | 100 | Campo opcional, PUEDE SER NULO. | |
| | Fecha | Datos del tiempo de programación. | | | si | fecha | DATETIME | 20 | Campo requerido, NO DEBE SER NULO. | |
| | ID_Médico | Identificador único del médico. | SI | | si | numeros | INT(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | Nombre | Nombre completo del médico. | | | si | letras | VARCHAR(30) | 30 | Campo requerido, NO DEBE SER NULO. | |
| | Especialidad | Área específica de conocimiento | | | si | letras | VARCHAR(20) | 20 | Campo requerido, NO DEBE SER NULO. | |
| | Horario_de_Trabajo | Cantidad de horas empleadas. | | | si | letras | VARCHAR(20) | 20 | Campo requerido, NO DEBE SER NULO. | |
| | Teléfono | Número de contacto telefónico | | | si | numeros | VARCHAR(20) | 20 | Campo requerido, NO DEBE SER NULO. | |
| | ID_Tratamiento | Identificador único del tratamiento. | SI | | si | numeros | INT(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | ID_Paciente | Identificador del paciente-tratamiento. | | SI TABLA TRATAM | si | numeros | INT(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | ID_Médico | Identificador del médico-tratamiento. | | SI TABLA TRATAM | si | numeros | INT(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | Descripción | Explicación breve del tratamiento. | | | si | texto | TEXT | 100 | Campo requerido, NO DEBE SER NULO. | |
| | Fecha_de_inicio | Datos del tiempo de arranque | | | si | fecha | DATETIME | 20 | Campo requerido, NO DEBE SER NULO. | |
| | Fecha_deFin | Datos del tiempo de conclusión. | | | no | fecha | DATETIME | 20 | Campo opcional, PUEDE SER NULO. | |
| | ID_Medicamento | Identificador único del medicamento. | SI | | si | numero | INT(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | Nombre | Nombre del medicamento. | | | si | letras | VARCHAR(30) | 30 | Campo requerido, NO DEBE SER NULO. | |
| | Descripción | Explicación breve del medicamento. | | | si | texto | TEXT | 100 | Campo requerido, NO DEBE SER NULO. | |
| | ID_Personal | Identificador único del personal médico. | SI | | si | numeros | INT(10) | 10 | Campo requerido, NO DEBE SER NULO. | |
| | Nombre | Nombre completo del personal médico. | | | si | letras | VARCHAR(30) | 30 | Campo requerido, NO DEBE SER NULO. | |
| | Cargo | Posición o responsabilidad ocupada. | | | si | letras | VARCHAR(20) | 20 | Campo requerido, NO DEBE SER NULO. | |
| | Horario_de_Trabajo | Cantidad de horas empleadas. | | | si | letras | VARCAR(20) | 20 | Campo requerido, NO DEBE SER NULO. | |
| | Teléfono | Número de contacto telefónico | | | si | numeros | INT(20) | 20 | Campo requerido, NO DEBE SER NULO. | |

Ilustración 3 Diccionario de Datos del caso Hospital

| | |
|--|------------------------|
| | Tabla HISTORIAL_MEDICO |
| | Tabla MÉDICO |
| | Tabla TRATAMIENTO |
| | Tabla MEDICAMENTO |

Esquema de transición

Se deben crear las siguientes tablas:

PACIENTE(ID_Paciente, Nombre, Numero_Seguro_Medico, Fecha_Nacimiento, Genero, Direccion, Telefono) PACIENTE.ID_Paciente es una clave primaria.

CITA_MEDICA(ID_Cita, Fecha_Hora, ID_Paciente, ID_Medico) CITA_MEDICA.ID_Cita es una clave primaria, CITA_MEDICA.ID_Paciente es una clave foránea de PACIENTE, CITA_MEDICA.ID_Medico es una clave foránea de MEDICO.

HISTORIAL_MEDICO(ID_Historial, ID_Paciente, Descripcion, Fecha) HISTORIAL_MEDICO.ID_Historial es una clave primaria, HISTORIAL_MEDICO.ID_Paciente es una clave foránea de PACIENTE.

MEDICO(ID_Medico, Nombre, Especialidad, Horario_Trabajo, Telefono) MEDICO.ID_Medico es una clave primaria.

TRATAMIENTO(ID_Tratamiento, ID_Paciente, ID_Medico, Descripcion, Fecha_Inicio, Fecha_Fin) TRATAMIENTO.ID_Tratamiento es una clave primaria, TRATAMIENTO.ID_Paciente es una clave foránea de PACIENTE, TRATAMIENTO.ID_Medico es una clave foránea de MEDICO.

MEDICAMENTO(ID_Medicamento, Nombre, Descripcion) MEDICAMENTO.ID_Medicamento es una clave primaria.

PERSONAL_MEDICO(ID_Personal, Nombre, Cargo, Horario_Trabajo, Telefono) PERSONAL_MEDICO.ID_Personal es una clave primaria.

Modelo ER

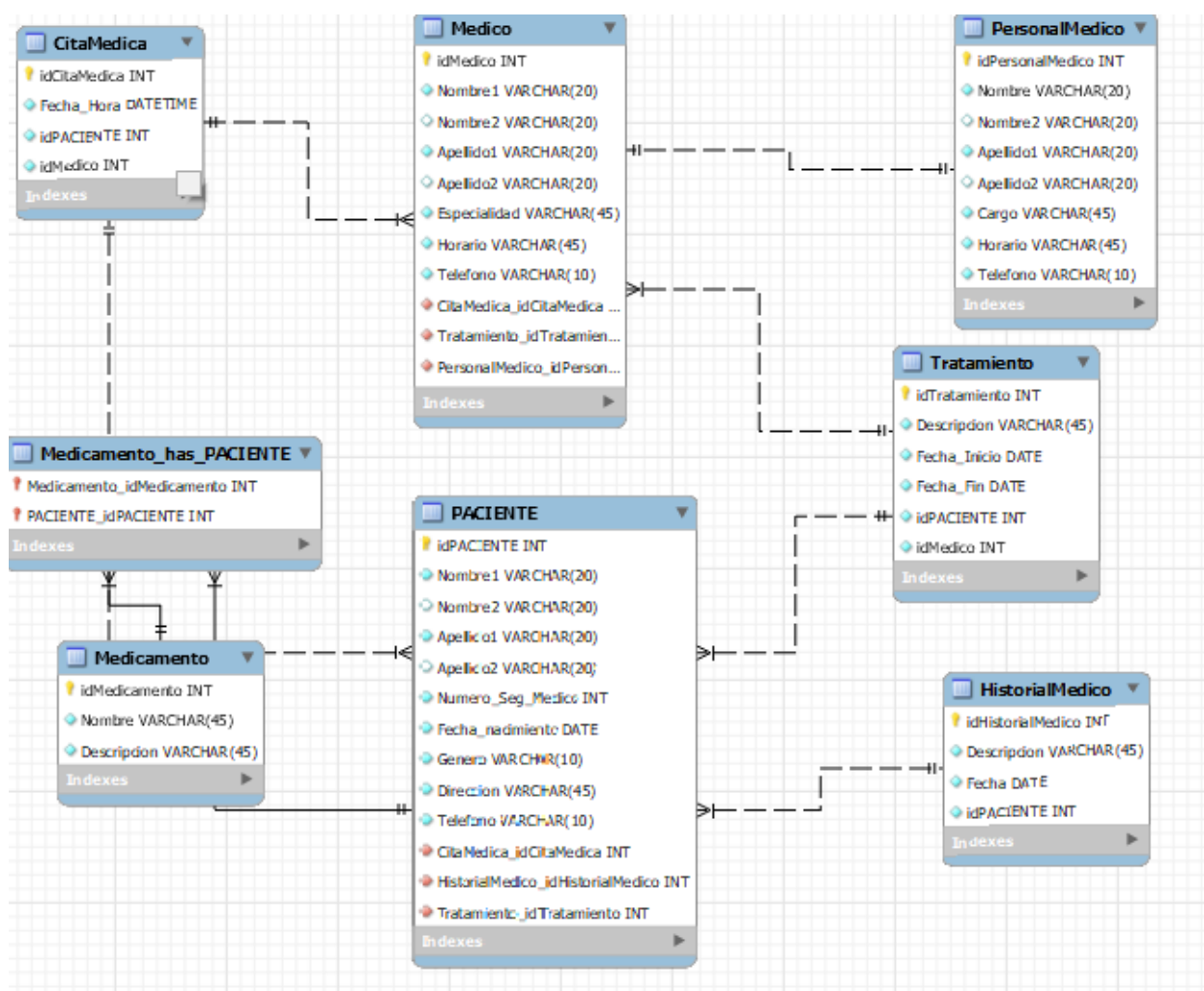


Ilustración 4 Diagrama ER del caso Hospital

Descripción de 3 reglas formales

Primera forma normal

En la primera forma normal, cada atributo en cada tabla debe contener un solo valor, es decir, los atributos no deben ser multivaluados ni repetidos.

Para el caso del hospital, todos los atributos en las tablas PACIENTE, CITA MÉDICA, HISTORIAL MÉDICO, MÉDICO, TRATAMIENTO, MEDICAMENTO y PERSONAL MÉDICO

cumplen con esta regla, ya que cada atributo tiene un solo valor y no es multivaluado ni repetido.

Segunda forma normal

En la segunda forma normal, además de cumplir con la 1FN, cada atributo no clave debe depender completamente de la clave primaria de la tabla y no de una parte de ella.

Para el caso del hospital, todas las tablas ya están en 2FN, ya que cada atributo no clave depende completamente de la clave primaria correspondiente. Por ejemplo, en la tabla PACIENTE, los atributos como Nombre, Número de Seguro Médico, Fecha de Nacimiento, etc., dependen completamente del ID_Paciente, que es la clave primaria.

Tercera forma normal

En la tercera forma normal, además de cumplir con la 2FN, no debe haber dependencias transitivas, es decir, los atributos no clave no deben depender de otros atributos no clave.

Para el caso del hospital, ninguna tabla tiene dependencias transitivas, por lo que todas cumplen con la 3FN. Por ejemplo, en la tabla TRATAMIENTO, la Descripción del tratamiento depende directamente del ID_Tratamiento (clave primaria) y no de otros atributos no clave.

Código

```
CREATE TABLE IF NOT EXISTS `mydb`.`citamedica` (

  `idCitaMedica` INT NOT NULL,

  `Fecha_Hora` DATETIME NOT NULL,

  `idPACIENTE` INT NOT NULL,
```

`idMedico` INT NOT NULL,

PRIMARY KEY (`idCitaMedica`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

CREATE TABLE IF NOT EXISTS `mydb`.`historialmedico` (

`idHistorialMedico` INT NOT NULL,

`Descripcion` VARCHAR(45) NOT NULL,

`Fecha` DATE NOT NULL,

`idPACIENTE` INT NOT NULL,

PRIMARY KEY (`idHistorialMedico`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

CREATE TABLE IF NOT EXISTS `mydb`.`medicamento` (

`idMedicamento` INT NOT NULL,

`Nombre` VARCHAR(45) NOT NULL,

`Descripcion` VARCHAR(45) NOT NULL,

PRIMARY KEY (`idMedicamento`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

CREATE TABLE IF NOT EXISTS `mydb`.`tratamiento` (

 `idTratamiento` INT NOT NULL,

 `Descripcion` VARCHAR(45) NOT NULL,

 `Fecha_Inicio` DATE NOT NULL,

 `Fecha_Fin` DATE NOT NULL,

 `idPACIENTE` INT NOT NULL,

 `idMedico` INT NOT NULL,

 PRIMARY KEY (`idTratamiento`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

CREATE TABLE IF NOT EXISTS `mydb`.`paciente` (

 `idPACIENTE` INT NOT NULL,

 `Nombre` VARCHAR(70) NOT NULL,

 `Numero_Seg_Medico` INT NOT NULL,

 `Fecha_nacimiento` DATE NOT NULL,

```

`Genero` VARCHAR(10) NOT NULL,

`Direccion` VARCHAR(45) NOT NULL,

`Telefono` VARCHAR(10) NOT NULL,

`CitaMedica_idCitaMedica` INT NOT NULL,

`HistorialMedico_idHistorialMedico` INT NOT NULL,

`Tratamiento_idTratamiento` INT NOT NULL,

PRIMARY KEY (`idPACIENTE`),

INDEX `fk_PACIENTE_CitaMedica_idx` (`CitaMedica_idCitaMedica` ASC) VISIBLE,

INDEX `fk_PACIENTE_HistorialMedico1_idx` (`HistorialMedico_idHistorialMedico` ASC)
VISIBLE,

INDEX `fk_PACIENTE_Tratamiento1_idx` (`Tratamiento_idTratamiento` ASC) VISIBLE,

CONSTRAINT `fk_PACIENTE_CitaMedica`

FOREIGN KEY (`CitaMedica_idCitaMedica`)

REFERENCES `mydb`.`citamedica` (`idCitaMedica`),

CONSTRAINT `fk_PACIENTE_HistorialMedico1`

FOREIGN KEY (`HistorialMedico_idHistorialMedico`)

REFERENCES `mydb`.`historialmedico` (`idHistorialMedico`),

CONSTRAINT `fk_PACIENTE_Tratamiento1`

FOREIGN KEY (`Tratamiento_idTratamiento`)

REFERENCES `mydb`.`tratamiento` (`idTratamiento`))

```

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

```
CREATE TABLE IF NOT EXISTS `mydb`.`medicamento_has_paciente` (

  `Medicamento_idMedicamento` INT NOT NULL,

  `PACIENTE_idPACIENTE` INT NOT NULL,

  PRIMARY KEY (`Medicamento_idMedicamento`, `PACIENTE_idPACIENTE`),

  INDEX `fk_Medicamento_has_PACIENTE_PACIENTE1_idx` (`PACIENTE_idPACIENTE`
ASC) VISIBLE,

  INDEX `fk_Medicamento_has_PACIENTE_Medicamento1_idx`
(`Medicamento_idMedicamento` ASC) VISIBLE,

  CONSTRAINT `fk_Medicamento_has_PACIENTE_Medicamento1`

  FOREIGN KEY (`Medicamento_idMedicamento`)

  REFERENCES `mydb`.`medicamento` (`idMedicamento`),

  CONSTRAINT `fk_Medicamento_has_PACIENTE_PACIENTE1`

  FOREIGN KEY (`PACIENTE_idPACIENTE`)

  REFERENCES `mydb`.`paciente` (`idPACIENTE`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`personalmedico` (
```

```
  `idPersonalMedico` INT NOT NULL,
```

```
  `Nombre` VARCHAR(70) NOT NULL,
```

```
  `Cargo` VARCHAR(45) NOT NULL,
```

```
  `Horario` VARCHAR(45) NOT NULL,
```

```
  `Telefono` VARCHAR(10) NOT NULL,
```

```
  PRIMARY KEY (`idPersonalMedico`))
```

```
ENGINE = InnoDB
```

```
DEFAULT CHARACTER SET = utf8mb3;
```

```
CREATE TABLE IF NOT EXISTS `mydb`.`medico` (
```

```
  `idMedico` INT NOT NULL,
```

```
  `Nombre` VARCHAR(70) NOT NULL,
```

```
  `Especialidad` VARCHAR(45) NOT NULL,
```

```
  `Horario` VARCHAR(45) NOT NULL,
```

```
  `Telefono` VARCHAR(10) NOT NULL,
```

```
  `CitaMedica_idCitaMedica` INT NOT NULL,
```

```
  `Tratamiento_idTratamiento` INT NOT NULL,
```

```
  `PersonalMedico_idPersonalMedico` INT NOT NULL,
```

```
  PRIMARY KEY (`idMedico`),
```

```

INDEX `fk_Medico_CitaMedica1_idx` (`CitaMedica_idCitaMedica` ASC) VISIBLE,

INDEX `fk_Medico_Tratamiento1_idx` (`Tratamiento_idTratamiento` ASC) VISIBLE,

INDEX `fk_Medico_PersonalMedico1_idx` (`PersonalMedico_idPersonalMedico` ASC)
VISIBLE,

CONSTRAINT `fk_Medico_CitaMedica1`

FOREIGN KEY (`CitaMedica_idCitaMedica`)

REFERENCES `mydb`.`citamedica` (`idCitaMedica`),

CONSTRAINT `fk_Medico_PersonalMedico1`

FOREIGN KEY (`PersonalMedico_idPersonalMedico`)

REFERENCES `mydb`.`personalmedico` (`idPersonalMedico`),

CONSTRAINT `fk_Medico_Tratamiento1`

FOREIGN KEY (`Tratamiento_idTratamiento`)

REFERENCES `mydb`.`tratamiento` (`idTratamiento`))

ENGINE = InnoDB

DEFAULT CHARACTER SET = utf8mb3;

```

Datos de registro

```
mysql> use mydb;
Database changed
mysql> show tables;
+-----+
| Tables_in_mydb |
+-----+
| citamedica      |
| historialmedico |
| medicamento     |
| medicamento_has_paciente |
| medico          |
| paciente        |
| personalmedico  |
| tratamiento     |
+-----+
8 rows in set (0.00 sec)
```

Ilustración 5 Tablas de la Base de datos Hospital

```
mysql> describe citamedica;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| idCitaMedica | int       | NO   | PRI | NULL    |       |
| Fecha_Hora   | datetime  | NO   |     | NULL    |       |
| idPACIENTE   | int       | NO   |     | NULL    |       |
| idMedico     | int       | NO   |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)
```

Ilustración 6 Tabla citamedica correspondiente a la base de datos de Hospital

```
mysql> describe historialmedico;
```

| Field | Type | Null | Key | Default | Extra |
|-------------------|-------------|------|-----|---------|-------|
| idHistorialMedico | int | NO | PRI | NULL | |
| Descripcion | varchar(45) | NO | | NULL | |
| Fecha | date | NO | | NULL | |
| idPACIENTE | int | NO | | NULL | |

4 rows in set (0.00 sec)

Ilustración 7 Tabla historialmedico correspondiente a la base de datos de Hospital

```
mysql> describe medicamento;
```

| Field | Type | Null | Key | Default | Extra |
|---------------|-------------|------|-----|---------|-------|
| idMedicamento | int | NO | PRI | NULL | |
| Nombre | varchar(45) | NO | | NULL | |
| Descripcion | varchar(45) | NO | | NULL | |

3 rows in set (0.00 sec)

Ilustración 8 Tabla medicamento correspondiente a la base de datos de Hospital

```
mysql> describe medicamento_has_paciente;
```

| Field | Type | Null | Key | Default | Extra |
|---------------------------|------|------|-----|---------|-------|
| Medicamento_idMedicamento | int | NO | PRI | NULL | |
| PACIENTE_idPACIENTE | int | NO | PRI | NULL | |

2 rows in set (0.00 sec)

Ilustración 9 Tabla medicamento_has_paciente correspondiente a la base de datos de Hospital

```
mysql> describe medico;
```

| Field | Type | Null | Key | Default | Extra |
|---------------------------------|-------------|------|-----|---------|-------|
| idMedico | int | NO | PRI | NULL | |
| Nombre | varchar(70) | NO | | NULL | |
| Especialidad | varchar(45) | NO | | NULL | |
| Horario | varchar(45) | NO | | NULL | |
| Telefono | varchar(10) | NO | | NULL | |
| CitaMedica_idCitaMedica | int | NO | MUL | NULL | |
| Tratamiento_idTratamiento | int | NO | MUL | NULL | |
| PersonalMedico_idPersonalMedico | int | NO | MUL | NULL | |

8 rows in set (0.00 sec)

Ilustración 10 Tabla medico correspondiente a la base de datos de Hospital

```
mysql> describe paciente;
```

| Field | Type | Null | Key | Default | Extra |
|-----------------------------------|-------------|------|-----|---------|-------|
| idPACIENTE | int | NO | PRI | NULL | |
| Nombre | varchar(70) | NO | | NULL | |
| Numero_Seg_Medico | int | NO | | NULL | |
| Fecha_nacimiento | date | NO | | NULL | |
| Genero | varchar(10) | NO | | NULL | |
| Direccion | varchar(45) | NO | | NULL | |
| Telefono | varchar(10) | NO | | NULL | |
| CitaMedica_idCitaMedica | int | NO | MUL | NULL | |
| HistorialMedico_idHistorialMedico | int | NO | MUL | NULL | |
| Tratamiento_idTratamiento | int | NO | MUL | NULL | |

10 rows in set (0.00 sec)

Ilustración 11 Tabla paciente correspondiente a la base de datos de Hospital

```
mysql> describe personalmedico;
```

| Field | Type | Null | Key | Default | Extra |
|------------------|-------------|------|-----|---------|-------|
| idPersonalMedico | int | NO | PRI | NULL | |
| Nombre | varchar(70) | NO | | NULL | |
| Cargo | varchar(45) | NO | | NULL | |
| Horario | varchar(45) | NO | | NULL | |
| Telefono | varchar(10) | NO | | NULL | |

5 rows in set (0.00 sec)

Ilustración 12 Tabla personalmedico correspondiente a la base de datos de Hospital


```
mysql> describe tratamiento;
```

| Field | Type | Null | Key | Default | Extra |
|---------------|-------------|------|-----|---------|-------|
| idTratamiento | int | NO | PRI | NULL | |
| Descripcion | varchar(45) | NO | | NULL | |
| Fecha_Inicio | date | NO | | NULL | |
| Fecha_Fin | date | NO | | NULL | |
| idPACIENTE | int | NO | | NULL | |
| idMedico | int | NO | | NULL | |

```
6 rows in set (0.00 sec)
```

Ilustración 13 Tabla tratamiento correspondiente a la base de datos de Hospital

Joins

1. **Obtener el nombre del paciente y la fecha de su próxima cita médica:**

```
SELECT p.Nombre, cm.Fecha_y_Hora
```

```
FROM Paciente p
```

```
INNER JOIN Cita_Médica cm ON p.ID_Paciente = cm.ID_Paciente
```

2. **Mostrar el nombre del médico y la especialidad de cada cita médica:**

```
SELECT m.Nombre AS Nombre_Médico, m.Especialidad, cm.Fecha_y_Hora
```

```
FROM Médico m
```

```
INNER JOIN Cita_Médica cm ON m.ID_Médico = cm.ID_Médico
```

```
JOIN TELEFONO T ON N.codigo_de_socio
```

3. Listar el historial médico de un paciente junto con su nombre:

```
SELECT p.Nombre, hm.Descripción, hm.Fecha  
  
FROM Paciente p  
  
INNER JOIN Historial_Médico hm ON p.ID_Paciente = hm.ID_Paciente  
  
WHERE p.Nombre = 'Nombre del paciente'
```

4. Mostrar los tratamientos activos de un paciente y el nombre del médico responsable:

```
SELECT p.Nombre AS Nombre_Paciente, t.Descripción AS Tratamiento, m.Nombre AS  
Nombre_Médico  
  
FROM Paciente p  
  
INNER JOIN Tratamiento t ON p.ID_Paciente = t.ID_Paciente  
  
INNER JOIN Médico m ON t.ID_Médico = m.ID_Médico  
  
WHERE t.Fecha_de_Fin IS NULL
```

Conclusiones

En conclusión, el diseño e implementación de una base de datos relacional para un sistema de gestión de pacientes en un entorno hospitalario es un proceso complejo pero fundamental para garantizar la eficiencia, integridad y calidad del cuidado médico proporcionado. Desde el modelado conceptual hasta la implementación física, cada etapa del proceso requiere una cuidadosa planificación y consideración de los requisitos del sistema y las necesidades del usuario. Al aplicar principios de normalización, diseño óptimo de tablas y relaciones adecuadas entre entidades, se puede crear una base de datos robusta y escalable que satisfaga las demandas del entorno hospitalario y facilite la toma de decisiones clínicas informadas. Además, la capacidad de adaptación y mantenimiento continuo de la base de datos es crucial para garantizar su utilidad a largo plazo en un entorno dinámico y en constante evolución. En última instancia, una base de datos bien diseñada y gestionada puede contribuir significativamente a la mejora de la atención médica y la satisfacción del paciente en un hospital.

Referencias de las ilustraciones

| | |
|--|----|
| Ilustración 1 Primer Propuesta Diagrama MER del caso Hospital..... | 10 |
| Ilustración 2 Diccionario de Datos del caso Hospital | 12 |
| Ilustración 3 Diagrama ER del caso Hospital | 14 |
| Ilustración 4 Tablas de la Base de datos Hospital | 22 |
| Ilustración 5 Tabla citamedica correspondiente a la base de datos de Hospital | 22 |
| Ilustración 6 Tabla historialmedico correspondiente a la base de datos de Hospital | 23 |
| Ilustración 7 Tabla medicamento correspondiente a la base de datos de Hospital..... | 23 |
| Ilustración 8 Tabla medicamento_has_paciente correspondiente a la base de datos de Hospital | 23 |
| Ilustración 9 Tabla medico correspondiente a la base de datos de Hospital..... | 24 |
| Ilustración 10 Tabla paciente correspondiente a la base de datos de Hospital..... | 24 |
| Ilustración 11 Tabla personalmedico correspondiente a la base de datos de Hospital | 24 |
| Ilustración 12 Tabla tratamiento correspondiente a la base de datos de Hospital | 25 |

LINK DE REPOSITORIO GITHUB

<https://github.com/Los-Yucks/Proyecto-Final.git>