

Institución Universitaria Pascual Bravo
Facultad de Ingeniería
Departamento de Sistemas Digitales

PROYECTO DE AULA (PA) - 2025-2 (TIA5)
DDL - Diccionario de Datos Físico
EQUIPO "5"

Programa : Tecnología en desarrollo de Software
Asignatura : BASE DE DATOS I Código ET-0057
Docente : JAIME E SOTO U
Tipo proyecto : Aprendizaje Basado en Proyecto (ABP)

PROYECTO

**Arquitectura de una base de datos para el servicio de Hospitalización del Sistema de Salud del
Departamento de Antioquia, Colombia**

Objetivo General del Proyecto

Los estudiantes de "Base de Datos I" deben diseñar, construir, poblar, consultar y validar una base de datos robusta, flexible y segura para almacenar y monitorear la información de hospitalización de la Red de Salud del Departamento de Antioquia, Colombia. La información de hospitalización debe estar integrada y alimentar a un Sistema de Historia Clínica Electrónica de cada paciente del Departamento de Antioquia. Para lograr esto, deben realizar las siguientes fases:

- **Tia 5: Implementación Base de Datos. Modelo Físico (DDL) (esta es la tarea a realizar)**
 - **Antes de construir el Diccionario de Datos Físico, los estudiantes deben corregir**
 - El Diagrama de Entidad-Relación
 - Revisar el proceso de normalización. Las tablas que surjan de la normalización deben estar presentes en el Diccionario de Datos Físico.
 - **Fase 4: Construir el Modelo Físico**
 - Inventario de tablas definitivo
 - Diccionario de Datos Físico. Las tablas deben tener los mismos nombres de las tablas resultantes de entidades y relaciones del proceso de normalización.
 - Scripts de la creación de todas las tablas. Deben incluir
 - Claves primarias y foráneas
 - Indices y Reglas (CONSTRAINTS: NOT NULL, PK, FK, UK, CHECK)
 - Deben estar colocadas den orden de creación. Primero las tablas independientes y después las tablas dependientas
 - Scripts de modificación de las tablas.

CONTEXTO

La Secretaría de Salud de la Gobernación de Antioquia tiene contemplada la licitación de un proyecto de envergadura para la atención en Salud en los servicios de Hospitalización de la Red De Atención del departamento. Inicialmente se requiere un sistema de información Web de Servicio de Atención e Información que se integrará en el futuro a un sistema de información de Historia Clínica Electrónica mucho más robusto. Para lograr este sistema de información, primero se debe construir una base de datos que cumpla con los requerimientos de la Secretaría de Salud para implementar un sistema de información de Hospitalización básico que se integre a un Sistema de Información de Historia Clínica Electrónica masivo. Adicionalmente, desde el punto de vista técnico, la base de datos debe cumplir con las propiedades ACID. Las propiedades ACID garantizan que una transacción tenga fiabilidad, integridad y robustez en un sistema de gestión de bases de datos (DBMS). ACID es un acrónimo que representa Atomicidad, Consistencia, Aislamiento y Durabilidad.

Institución Universitaria Pascual Bravo

Facultad de Ingeniería

Departamento de Sistemas Digitales

REQUERIMIENTOS

1.- REQUERIMIENTO GENERAL DE LA ACTIVIDAD

Creación el Diccionario de Datos FÍSICO de una Bases de Datos que almacene la información del servicio de Hospitalización como parte de la Historia Clínica Electrónica del Departamento de Antioquia, Colombia. En la actividad anterior, se realizó el proceso de Modelado Conceptual y Lógico. En esta fase, se debe implementar la base de datos en Modelo Lógico en una base de datos física a través de un Sistema de Gestión de Base de Datos (SGBD). El nombre de la base de datos es “*hce_antioquia*”

2.- REQUERIMIENTOS ESPECÍFICOS

- Corrección del Diagrama de Entidad-Relación de Chen
- Revisión del proceso de Normalización que culmine en el conjunto de tablas coherente con el Diagrama de Entidad-Relación
- Construcción del Diccionario de Datos Físico. Nota: La cantidad y nombres de las tablas del diccionario de datos físico debe ser coherente con el conjunto de tablas resultado del proceso de Normalización “revisado”. Cada tabla debe contener: PK, FK (si se requiere), UK, CHECK e INDEX.
- Construcción de los Scripts DDL de creación de las tablas de la Bases de Datos. Nota: La cantidad y nombres de las tablas del Script de creación debe ser coherente con el Diccionario de Datos Físico. Las tablas se deben crear en orden correcto de creación. Es decir, primero las tablas independientes y después las dependientes.
- Una vez culminada la tarea, se deben colocar todos los productos en el repositorio GIT que se viene utilizando durante el semestre. El GIT debe estar estructurado por Tareas y debe estar debidamente identificado como se ha solicitado desde el inicio del semestre.
- Anexo a esta plantilla de Informe, se le entrega una plantilla en Hoja de Cálculo para que coloque los resultados. En este informe cada estudiante miembro del grupo debe elaborar sus conclusiones individuales (mínimo de 300 palabras) sobre el impacto de esta tarea en su desarrollo académico y profesional; y cualquier reflexión que desee realizar sobre las competencias y saberes adquiridos.
- Video de Sustentación. En el video, se debe presentar con nombre e imagen cada miembro del grupo demostrando su participación en la tarea y evidenciando el código en ejecución.

3.- REQUERIMIENTOS DE DATOS

- La Secretaría hace entrega de 32 datos de uso obligatorio
- El equipo de diseño debe investigar y agregar 8 datos nuevos para integrarlos en la base de datos.

4.- REQUERIMIENTOS DE DISEÑO

- Diseño de una Arquitectura Conceptual
- Diseño de un Diccionario de Datos Genérico

5.- REQUERIMIENTOS DE HERRAMIENTAS (debe utilizar estas herramientas)

- [Draw.io](https://draw.io), Excel, PostgreSQL 15+, pgAdmin4, Python (opcional)

6.- REQUERIMIENTOS DE ENTREGA DE PRODUCTOS (las entregas deben subirse al repositorio GIT)

- Diccionario de Datos Físico (considerar el Diccionario de Datos Genérico como insumo)
- Creación de la base de datos “*hce_antioquia*” (DDL)
- Implementación de todas las reglas y restricciones

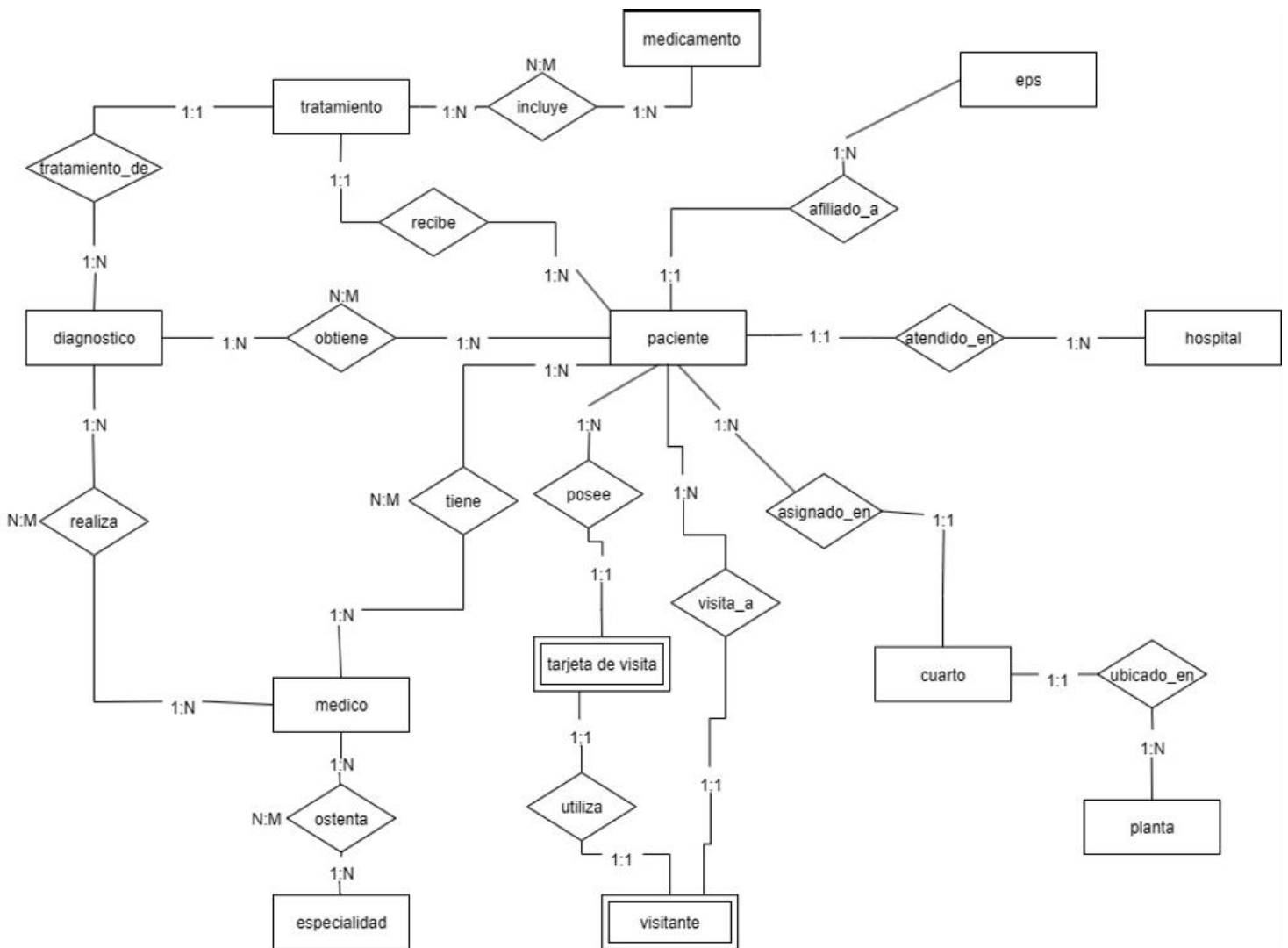
INFORME DE ENTREGA

Tarea 5 (TIA5): Diccionario de Datos FÍSICO BASES DE DATOS: “hce_antioquia”

Miembros del grupo

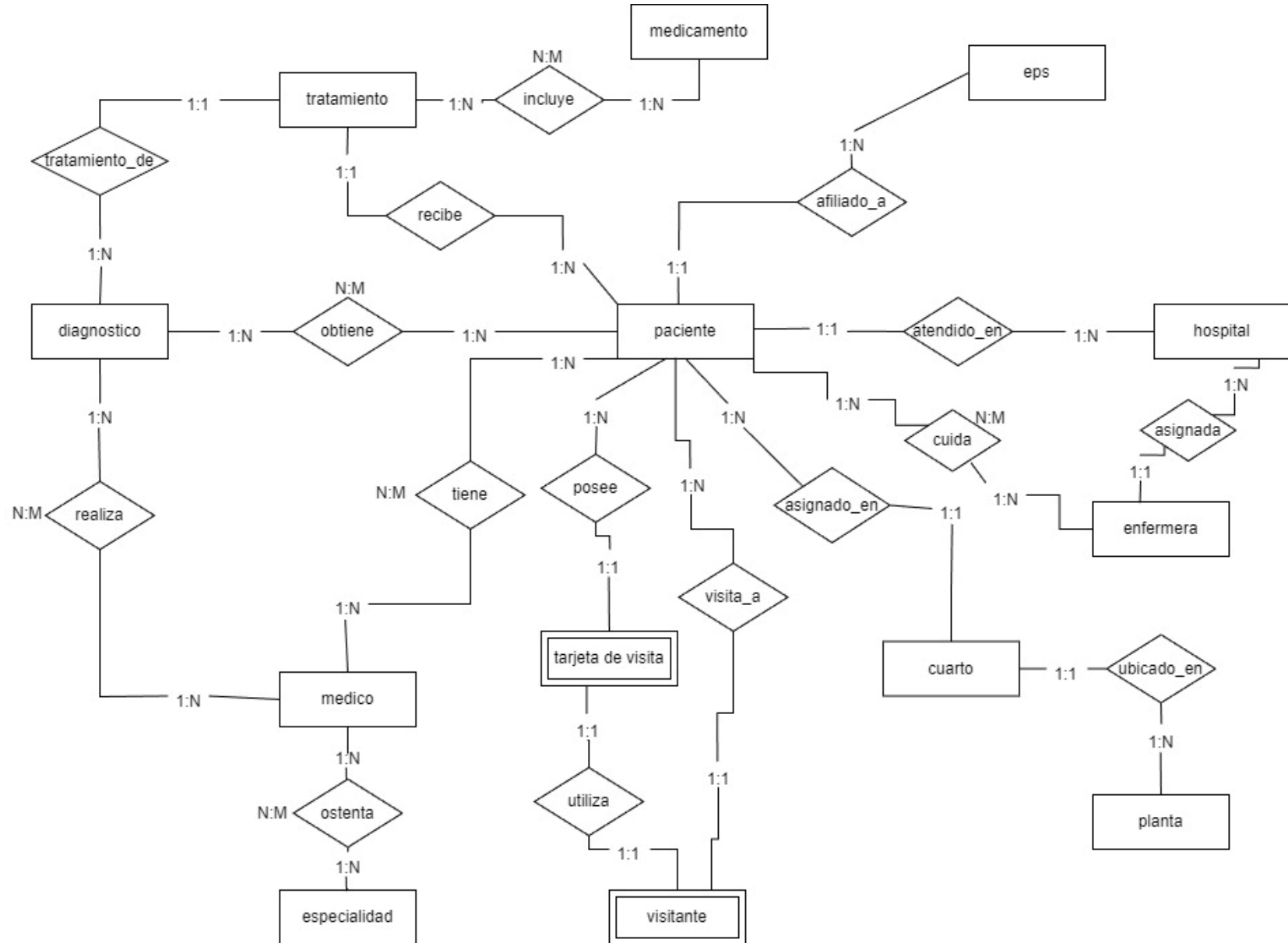
- Andres Santiago Quiroz Gómez
- Sebastián Sepúlveda Quintero
- Emanuel Ceballos Mesa

0.- Diagrama de Entidad-Relación de Chen (VIEJO CON ERROES)



1.- Diagrama de Entidad-Relación de Chen (corregido)

Institución Universitaria Pascual Bravo
Facultad de Ingeniería
Departamento de Sistemas Digitales



Hecho con Draw.io

Institución Universitaria Pascual Bravo
Facultad de Ingeniería
Departamento de Sistemas Digitales

2.- Revisión del proceso de Normalización anterior

Descripción de las correcciones del proceso de Normalización para llegar a 3FN		
#	Error o Falta en el resultado del Modelo Lógico	Solución (corrección)
1	datos multivalorados en la primera forma normal.	correccion de los atributos, separarlos en filas distintas haciendto que hayan mas de los 40 datos originales
2	19 tablas en la 2FN, cuando eran 13 entidades y 6 tablas tipo M:N = 19 tablas son.	eliminacion de la tabla sobrante y adiccion de las 3 tablas de muchos a muchos que faltaban
3	en 2FN habia una tabla que se deberia de haber añadido en la 3FN	eliminacion de la tabla relacional "visita" y se añadió en la 3FN
4	en 2FN habia una tabla que se deberia de haber añadido en la 3FN	eliminacion de la tabla relacional "paciente_tratamiento" porque se crea en la 3FN
5	en 2FN habia una tabla que se deberia de haber añadido en la 3FN	eliminacion de la tabla relacional "tratamiento_paciente" porque se cra en la 3FN
6	la tabla "estancia" en la 3FN era redundante y podría ser reemplazada por "asignacion"	se añadió los atributos de fechas de ingreso y salida del cuarto a la tabla "asignacion"
7	se elimina el campo de edad de "pacientes"	se puede calcular automaticamente con la fecha de nacimiento
8	con la nueva tabla de relacion M:N de medicos y pacientes se añade su id a hospitalizaciones	en la tabla "hospitalizaciones" se añade la FK de la relacion medico_paciente
9	clave de "medicamento" poco descriptiva	se modifica por medicida_id
10	en la tabla "visitante" faltaba otra PK para que sea entidad debil	se agrego los dos campos que conforman la PK y el campo "paciente_id" que se dirige a la tabla dependiente
11	en tabla "visita" quedaba la antigua PK de "visitante"	se elimino "visitante_id" y se puso el nuevo "numero_visitante"

Inventario de Tablas (Normalización)		
#	Nombre tabla	Tipo
1	paciente	E
2	eps	E
3	ciudad_pais	R
4	hospital	E
5	planta	E
6	cuarto	E
7	asignacion	R
8	tarjeta_visita	E
9	visitante	E
10	visita	R
11	medico	E
12	especialidad	E
13	medico_especialidad	R
14	medico_paciente	R
15	hospitalizaciones	R
16	diagnostico	E
17	enfermera	E
18	enfermera_paciente	R
19	diagnostico_paciente	R
20	diagnostico_medico	R
21	tratamiento	E
22	medicamento	E
23	tratamiento_paciente	R
24	tratamiento_medicamento	R

Institución Universitaria Pascual Bravo
Facultad de Ingeniería
Departamento de Sistemas Digitales

3.- Diccionario de Datos Físico

DICCIONARIO DE DATOS Inventario de Tablas			
#	Nombre Tabla	Descripción Tabla	Observaciones
1	paciente	Almacena la información personal y de contacto de los pacientes del sistema hospitalario	
2	eps	Catálogo de entidades prestadoras de salud	
3	ciudad_pais	Catálogo de ciudades y sus países correspondientes	
4	hospital	Catálogo de ciudades y sus países correspondientes	
5	planta	pisos del edificio del hospital	
6	cuarto	habitacion en numerada del hospital	
7	asignacion	asignacion del paciente a un cuarto y el motivo	
8	tarjeta_visita	tarjetas asignada a los pacientes	
9	visitante	personas que visitan al paciente	
10	visita	registro de quien visito al paciente	
11	medico	datos del medico	
12	especialidad	especialidades medicas	
13	medico_especialidad	relacion entre medico y especialidades	
14	medico_paciente	relacion entre el medico y el paciente	
15	atencion	registro de una atencion medica	
16	diagnostico	catalogo de diagnostico	
17	enfermera	persona que trabaja en enfermería	
18	enfermera_paciente	enfermeras y los pacientes que cuuidan	
19	diagnostico_paciente	aplicacion de diagnostico a paciente	
20	diagnostico_medico	medicos que participan en una atencion	
21	tratamiento	catalogo de planes de tratamientos	
22	medicamento	catalogo de medicamentos	
23	tratamiento_paciente	instancia del tratamiento aplicado	
24	tratamiento_medicamento	relacion tratamientos y medicamentos	

1	Tabla	paciente	Descripción Tabla	datos del paciente					Versión		2.0
#	Campo	Descripción Campo	Tipo Dato (SGBD)	Tamaño / Longitud	Nulo	PK	FK	UK	CHECK	Nota/Comentarios	
1	id_paciente	identificador interno del paciente	SERIAL		NO	S				Autoincremental	
2	nombre	nombre del paciente	VARCHAR	100	NO						
3	primer_apellido	primer apellido	VARCHAR	100	NO						
4	segundo_apellido	segundo apellido	VARCHAR	100	NO						
5	cedula	documento de identificación	VARCHAR	20	NO			S			
6	email	correo electronico del paciente	VARCHAR	100	NO			S			
7	fecha_nacimiento	fecha de nacimiento	DATE		NO				fecha_nacimiento <= current_date		
8	sexo	genero del paciente	VARCHAR	10	NO					M/F	
9	ciudad_id	lugar de nacimiento	INTEGER		NO		S			fk a ciudad	
10	eps_id	eps donde esta afiliado el paciente	INTEGER		NO		S			fk a eps	
11	hospital_id	hospital del lugar de atencion	INTEGER		NO		S			fk a hospital	
12	tarjetas_disponibles	tarjetas de visita asignadas	INTEGER		NO				(tarjetas_disponibles <= 4)	por defecto 4 tarjetas disponibles	
13	tarjetas_utilizadas	tarjetas de visita usadas por los visitantes	INTEGER		NO				(tarjetas_utilizadas >= 0)		
14	fecha_alta	fecha de salida del hospital del paciente	TIMESTAMP		SI					null si aun esta ingresado	
15	tel1	numero celular de contacto	VARCHAR	20	SI				CHECK (tel1 ~ "[0-9]+\$")	solo digitos	
16	tel2	segundo numero de contacto (opcional)	VARCHAR	20	SI				CHECK (tel2 ~ "[0-9]+\$")	solo digitos	

Institución Universitaria Pascual Bravo
Facultad de Ingeniería
Departamento de Sistemas Digitales

4.- Scripts DDL de creación de las Tablas

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer displays the database structure: Servers (1) > PostgreSQL 18 > Databases (4) > hce_antioquia > Tables (22). The main query editor is open to the 'hce_antioquia/postgres@PostgreSQL 18*' connection. The query history shows the following SQL script:

```
57 );
58 -----tablas dependientes-----
59 -----
60 -- 1. tabla paciente
61 create table paciente (
62     id_paciente serial primary key,
63     nombre varchar(100) not null,
64     primer_apellido varchar (100) not null,
65     segundo_apellido varchar (100),
66     cedula varchar (20) unique not null,
67     fecha_nacimiento date not null,
68     ciudad_id integer,
69     eps_id integer,
70     hospital_id integer,
71     constraint fk_ciudad_pais foreign key (ciudad_id)
72     references ciudad_pais(ciudad_id),
73
74     constraint fk_eps foreign key (eps_id)
75     references eps(eps_id),
76
77     constraint fk_hospital foreign key (hospital_id)
78     references hospital(hospital_id),
79 );
```

The Data Output pane at the bottom shows the table structure for 'paciente':

hospital_id	codigo_hospital	nombre_hospital	ciudad_id	direccion
[PK] integer	character varying (50)	character varying (200)	integer	character varying (250)

Total rows: 0 Query complete 00:00:00.681 CRLF Ln 9, Col 1

5.- Scripts DDL de “actualización o modificación” de las Tablas

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer displays the database structure: Servers > PostgreSQL 18 > Databases > hce_antioquia > Schemas > public > Tables > especialidad. The main query editor is open to the 'hce_antioquia/postgres@PostgreSQL 18*' connection. The query history shows the following SQL script:

```
42 -- Agregar email a la tabla paciente
43 ALTER TABLE paciente
44 ADD COLUMN email VARCHAR(100) UNIQUE NOT NULL;
45
46 -- Agregar fecha de emisión a la tabla tarjeta_visita
47 ALTER TABLE tarjeta_visita
48 ADD COLUMN fecha_emision DATE DEFAULT CURRENT_DATE;
49
50 --Agregar observaciones a la tabla visita
51 ALTER TABLE visita
52 ADD COLUMN observaciones TEXT;
53
54 -- Agregar dirección a la tabla hospital
55 ALTER TABLE hospital
56 ADD COLUMN direccion VARCHAR(250);
57
58 -- Agregar estado (activo/inactivo) a la tabla medico
59 ALTER TABLE medico
60 ADD COLUMN estado BOOLEAN DEFAULT TRUE;
61 --
62
63 --modificar nombre de campos
64 ALTER TABLE visitante
65 RENAME COLUMN nombre TO nombre_completo;
```

The Data Output pane at the bottom shows the table structure for 'especialidad':

medico_id	codigo_profesional	nombre	especialidad_id	telefono	estado
[PK] integer	character varying (20)	character varying (150)	integer	character varying (15)	boolean

Servers > PostgreSQL 18 > Databases > hce_antioquia > Schemas > public > Tables > especialidad CRLF Ln 54, Col 41

Institución Universitaria Pascual Bravo
Facultad de Ingeniería
Departamento de Sistemas Digitales

6.- Conclusiones individuales

- **Andres Quiroz Gómez**

El desarrollo de este proyecto de base de datos hospitalaria ha representado un proceso de aprendizaje que abarcó desde la conceptualización hasta la implementación física en PostgreSQL. Partiendo del diagrama entidad-relación de Chen y atravesando el proceso de normalización hasta la Tercera Forma Normal (3FN), logré comprender la importancia fundamental de diseñar bases de datos bien estructuradas que eviten redundancias y garanticen la integridad de los datos.

Durante el proceso de normalización, enfrenté desafíos significativos que fueron señalados por el profesor, lo cual me permitió identificar dependencias funcionales incorrectas y tablas que no cumplían completamente con los criterios de 3FN. Resolver estos errores fue una experiencia valiosa que reforzó mi comprensión sobre cómo cada forma normal elimina diferentes tipos de anomalías: la 1FN eliminando grupos repetitivos, y datos multivalorados, la 2FN removiendo dependencias parciales, y teniendo el numero de tablas consistente con las entidades y relaciones M:N y la 3FN eliminando dependencias transitivas, y las tablas que salen de esta forma, y junto a la correcta creación de las tablas de relaciones Esta corrección y mejora del modelo inicial me enseñó que el diseño de bases de datos es un proceso iterativo que requiere revisión constante y atención al detalle.

La transición del modelo conceptual al modelo físico en PostgreSQL me permitió profundizar en aspectos técnicos fundamentales. Aprendí a seleccionar tipos de datos apropiados (SERIAL, VARCHAR, INTEGER, DATE, TEXT, BOOLEAN, TIMESTAMP) considerando tanto las necesidades funcionales como la eficiencia del almacenamiento. La implementación de constraints como PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL y CHECK me mostró cómo garantizar la integridad referencial y la validez de los datos a nivel de base de datos, lo cual es crucial en sistemas hospitalarios.

En mi desarrollo académico, este proyecto consolidó conocimientos teóricos previamente fragmentados. La teoría de normalización cobró sentido práctico al aplicarla a un caso real con múltiples entidades y relaciones complejas. Comprendí que el diseño de bases de datos no es solo un ejercicio técnico, sino una disciplina que requiere pensamiento analítico, comprensión del dominio del negocio (en este caso, el sistema hospitalario), y capacidad para anticipar cómo se utilizará y consultará la información.

Profesionalmente, este trabajo me ha proporcionado habilidades directamente aplicables al mercado laboral. La demanda de profesionales capaces de diseñar, implementar y optimizar bases de datos es constante en la industria tecnológica.

- **Sebastian Sepulveda Quintero**

La elaboración del diagrama de Chen, el proceso de normalización y la construcción de los scripts DDL y DML representaron un ejercicio integral que permitió comprender de manera profunda cómo se estructura, organiza e implementa una base de datos desde sus fundamentos conceptuales hasta su funcionamiento real. A través del diagrama de Chen pude visualizar claramente las entidades, atributos y relaciones del caso de estudio, lo cual me ayudó a traducir un problema del mundo real en un modelo lógico entendible, preciso y bien estructurado. Reflexioné sobre la importancia de este paso inicial, ya que sin un buen diseño conceptual, la base

Institución Universitaria Pascual Bravo
Facultad de Ingeniería
Departamento de Sistemas Digitales

de datos puede presentar inconsistencias, duplicidad de información o problemas de integridad que afectan directamente el desempeño del sistema.

El proceso de normalización reforzó mi capacidad para analizar dependencias, identificar redundancias y mejorar la calidad de los datos. Comprendí que las formas normales no son simplemente reglas teóricas, sino herramientas prácticas que garantizan una base de datos más eficiente, coherente y fácil de mantener. Esta experiencia incrementó mi dominio técnico y fortaleció mi criterio profesional, pues ahora entiendo mejor por qué los sistemas reales deben diseñarse de manera rigurosa desde la estructura más básica.

Por otra parte, la creación de los scripts DDL y DML me permitió experimentar la transición del diseño conceptual al ambiente técnico. Definir tablas, llaves primarias, foráneas y restricciones implicó un alto nivel de atención, ya que cualquier error en la estructura afecta el funcionamiento general. Asimismo, los DML evidenciaron cómo se manipulan los datos de manera segura y ordenada, reforzando la importancia del control sobre la información.

El trabajo en equipo fue una parte fundamental del proceso. Intercambiar ideas, discutir modelos y corregir errores permitió generar una base de datos más sólida y con una visión compartida del caso de estudio. Esta colaboración no solo mejoró el resultado final, sino que también fortaleció habilidades sociales y profesionales esenciales para el ejercicio laboral. En conjunto, este proyecto no solo mejoró mis conocimientos técnicos, sino que también generó aprendizajes significativos que serán de gran utilidad en mi formación académica y en mi futuro profesional como desarrollador o administrador de sistemas de información.

- **Emanuel Ceballos Mesa**

El desarrollo completo del modelo de datos —desde el diagrama de Chen hasta la normalización y la ejecución de los scripts DDL y DML— representó una experiencia integral que me permitió comprender la importancia de cada etapa en la construcción sólida de una base de datos. El proceso comenzó con el diseño conceptual mediante el diagrama de entidad–relación, el cual facilitó visualizar los elementos clave del caso de estudio, los vínculos existentes entre ellos y la lógica interna del sistema. Pude evidenciar que antes de escribir una sola línea de código es esencial tener claridad estructural, pues un buen modelo conceptual evita inconsistencias, ambigüedades y errores futuros al momento de implementar la base de datos.

Posteriormente, la normalización se convirtió en una herramienta fundamental para refinar el diseño. A través de las distintas formas normales comprendí cómo se eliminan redundancias, cómo se mantienen las dependencias funcionales correctas y cómo se garantiza la integridad de los datos. Este proceso no solo fortaleció mis habilidades analíticas, sino que también me enseñó a valorar la importancia de la eficiencia y la coherencia en los sistemas de información. La normalización fue, sin duda, un puente entre la teoría académica y la práctica real en el diseño de bases de datos.

La implementación mediante scripts DDL y DML representó el componente técnico del proyecto. Al crear tablas, llaves primarias, llaves foráneas y restricciones, entendí cómo se materializa el diseño conceptual en un entorno de base de datos real. Además, trabajar con inserciones y consultas DML me permitió validar el funcionamiento

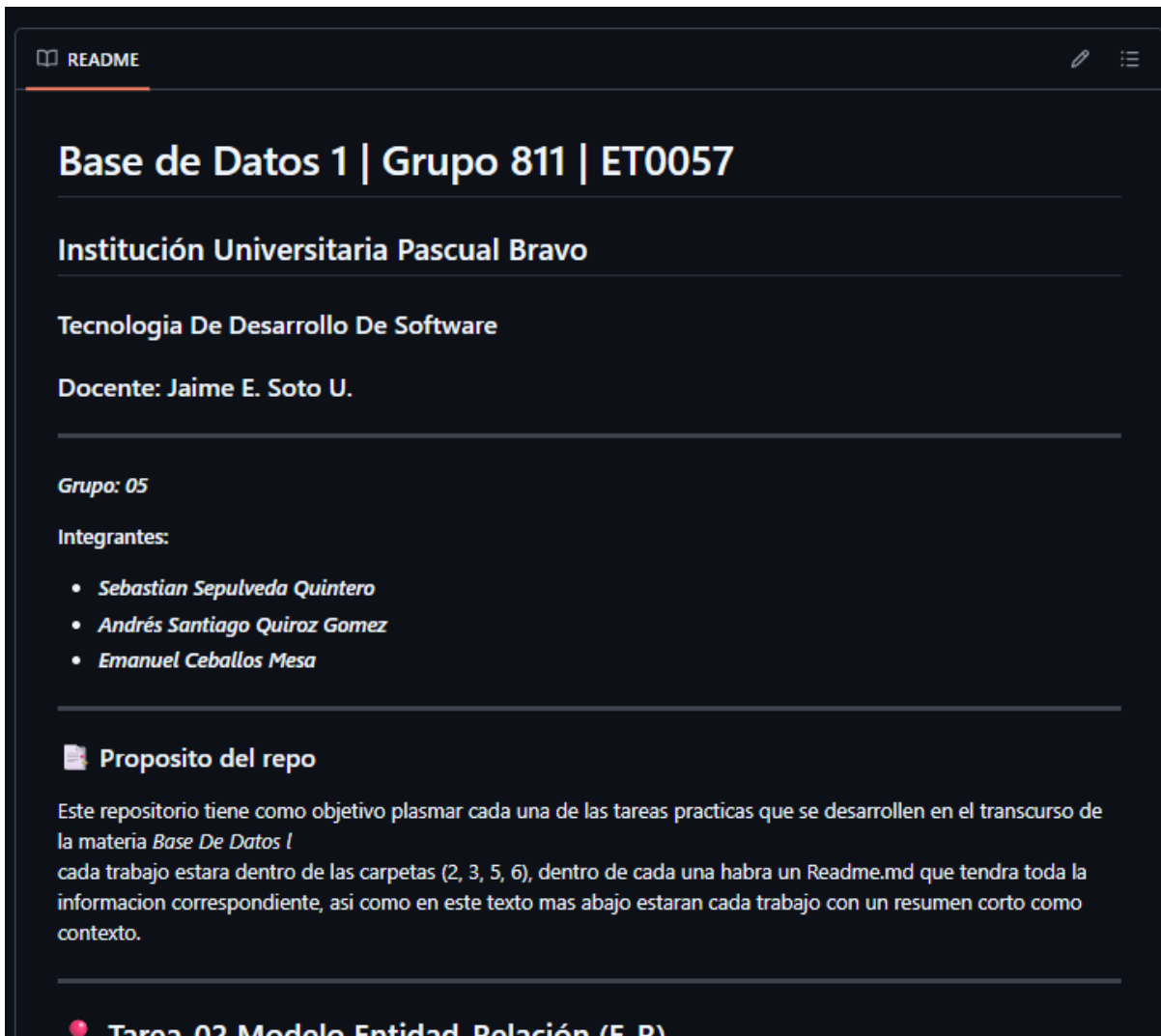
Institución Universitaria Pascual Bravo
Facultad de Ingeniería
Departamento de Sistemas Digitales

del modelo y comprobar que las reglas establecidas se cumplían adecuadamente. En este punto, pude reconocer lo relevante que es dominar los aspectos técnicos para construir sistemas robustos y seguros.

Finalmente, este proyecto fortaleció mi capacidad de trabajo en equipo, ya que analizar un caso de estudio de forma conjunta ayudó a identificar errores, mejorar ideas y construir una solución más completa. A nivel personal, esta experiencia amplió mi comprensión sobre la importancia del modelamiento de datos en el ámbito profesional, especialmente en áreas como desarrollo de software, análisis de información y administración de sistemas. Sin duda, los aprendizajes adquiridos en este trabajo tendrán un impacto positivo en mi formación académica y en mi futuro desempeño laboral.

7.- Informe

8.- Repositorio



9.- Video de Sustentación

Institución Universitaria Pascual Bravo

Facultad de Ingeniería

Departamento de Sistemas Digitales

RÚBRICA (TIA5)

Tarea Diccionario de Datos Físico (Creación y Modificación)

Las instrucciones y criterios de cada ítem se encuentran en el ítem en letras azules itálicas

#	Ítems Tarea	Peso	Cal
1	Diagrama ER de Chen (corregido)	5	
2	Revisión Normalización. Nota: debe llenar el formato con las correcciones	5	
3	Diccionario de Datos Físico completo. Nota: Diccionario de Datos es coherente con las entidades y relaciones que resultaron en tablas. Deben aparecer todas las tablas resultantes del proceso de normalización	20	
4	Scripts DDL de creación de la tablas. Nota: Todas las tablas del diccionario de datos con los mismos nombres y en orden de creación. Ejecución de los Scripts DDL de creación de la base de datos física sin errores. Utilizar la pestaña “Hoja de Cálculo	20	
5	Scripts DDL de “modificación” de las tablas. Nota: Primero se crean las tablas y después se ejecutan las instrucciones de modificación. Ejecución de los Scripts DDL de modificación de la base de datos física sin errores.	10	
6	Conclusiones individuales. Nota: Explicar el impacto que ha tenido en su desarrollo académico y profesional. Cada estudiante debe estar bien identificado en sus conclusiones (300 palabras mínimo)	5	
7	Informe de resultados. Nota: Informe con la calidad de presentación requerida (informe y hoja de cálculo).	5	
8	Repositorio GIT. Nota: Debe estar bien identificado y con la estructura en carpetas solicitada desde el inicio del semestre.	5	
9	Video de Sustentación. Nota: ES OBLIGATORIO. Se evalúa la calidad del Video. Cada participante se presenta adecuadamente con su nombre y con su rostro. Cada uno debe explicar su participación y mostrar código en ejecución en el SGBD. Si no se presenta el Video, la tarea se evaluará sobre 3 puntos como máxima nota.	25	
	NOTA = $xx/100 =$	Total	100

Nota: Eliminar todas las instrucciones en azul. Solo resultados