

Alejandro Parada - 202313816

Daniel Vargas - 202123892

Tomas Velasquez - 202311016

Proyecto - Entrega 1

## Modelo UML:

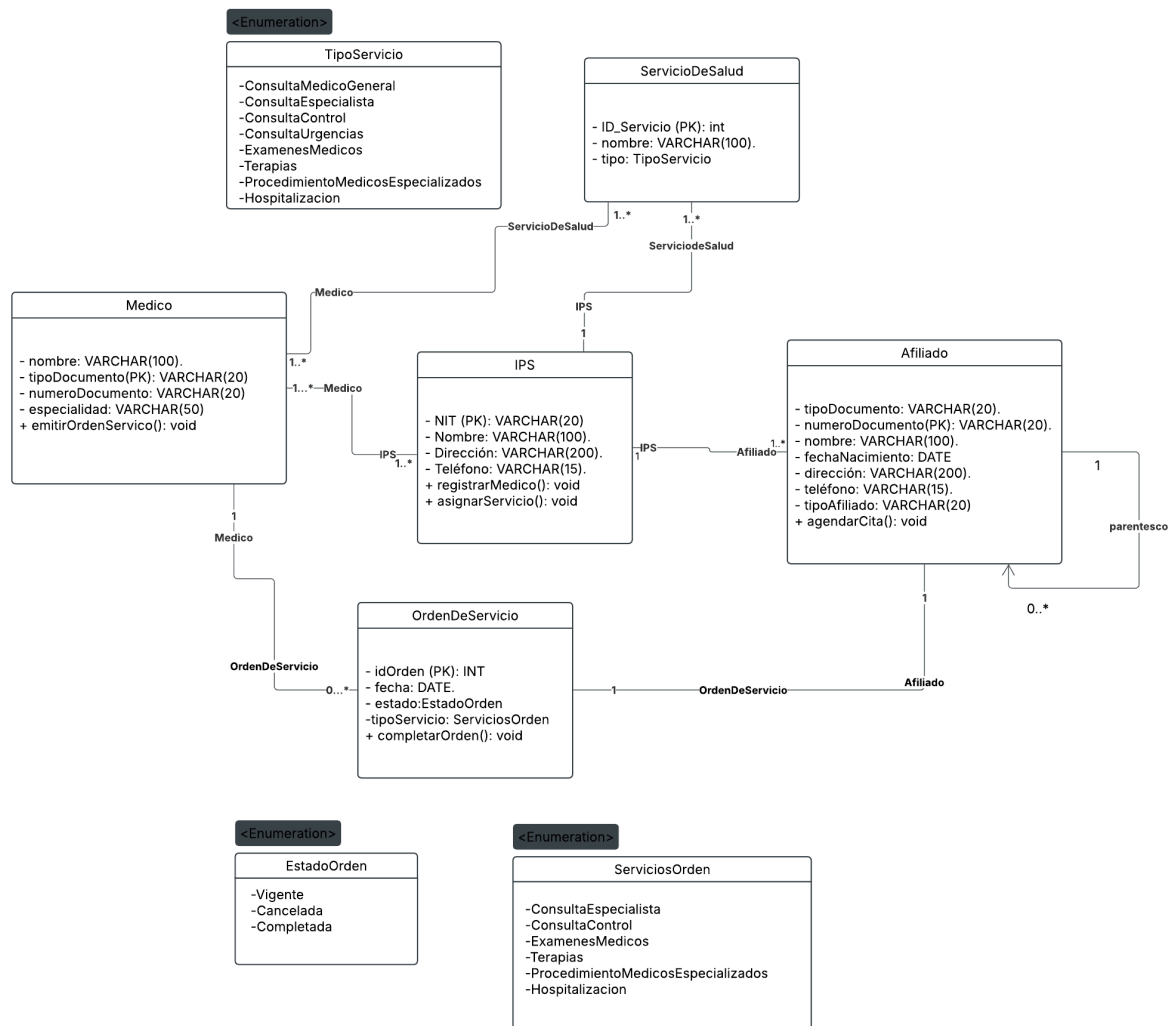
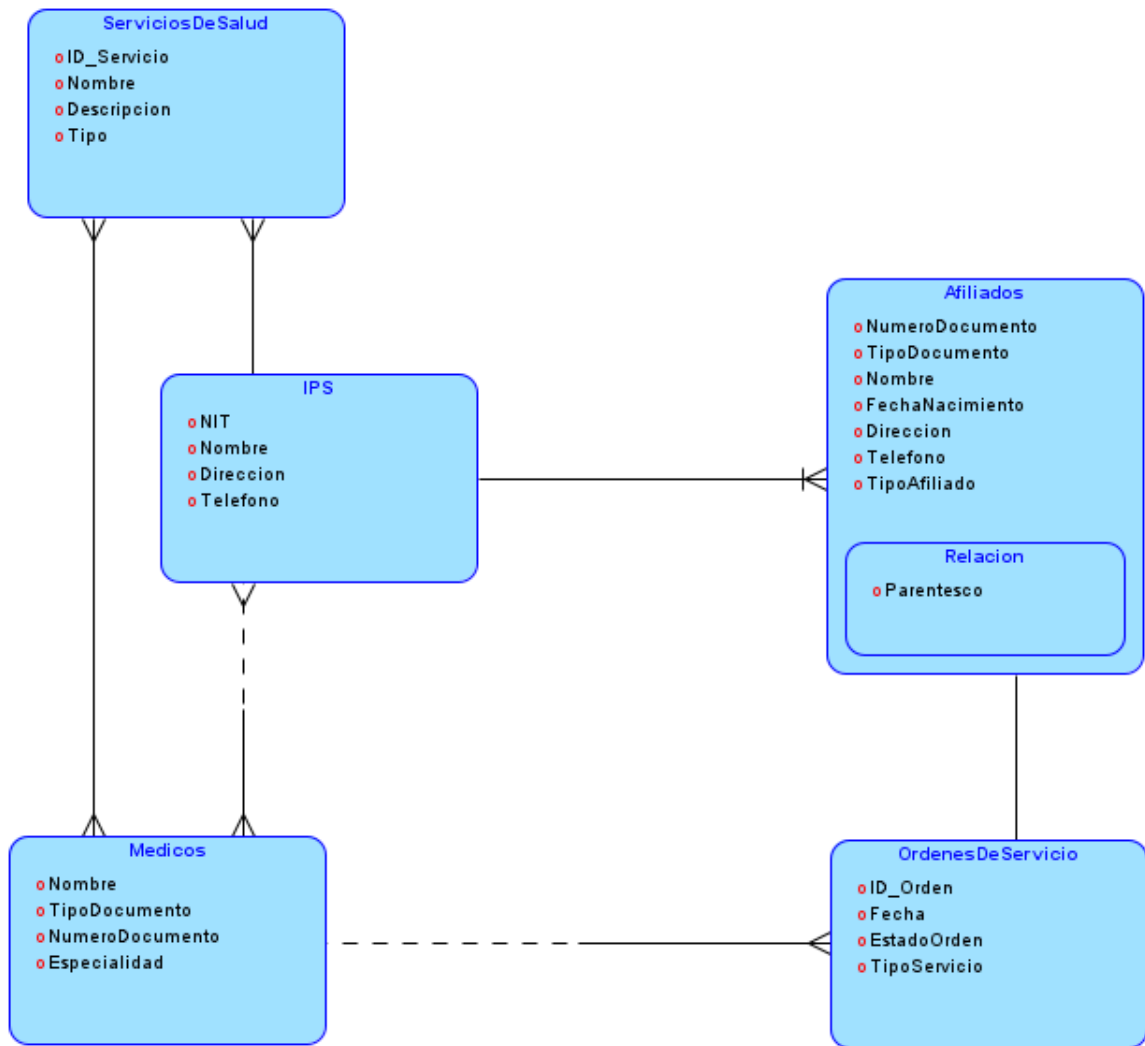
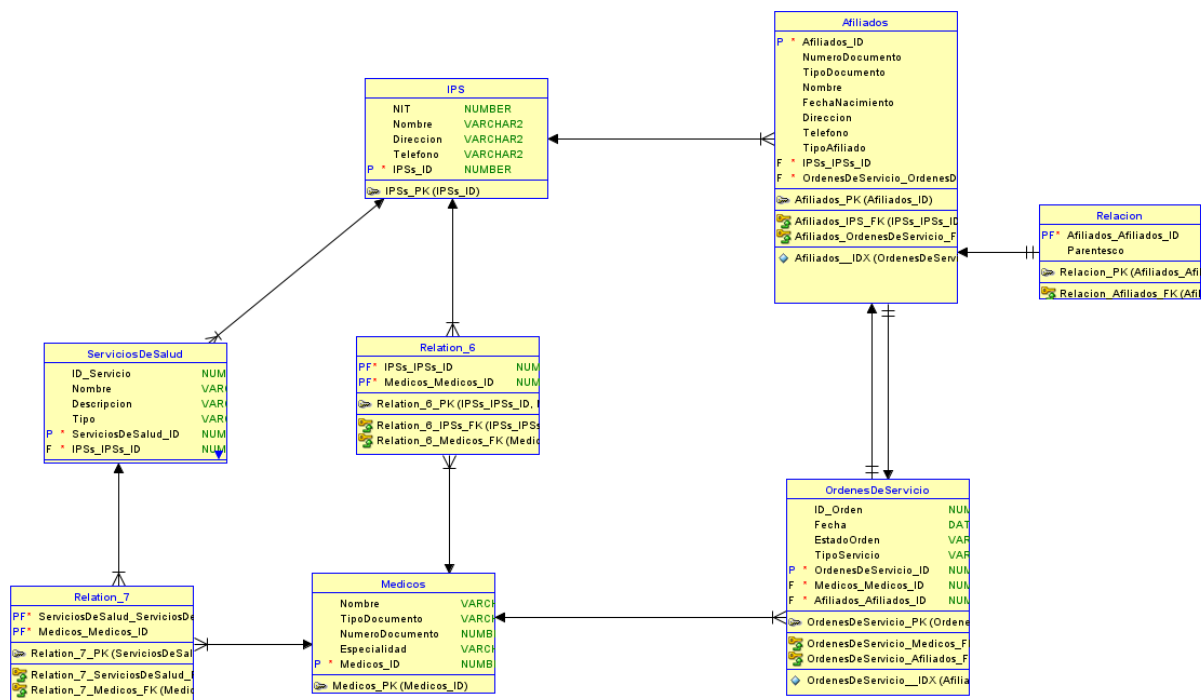


Diagrama E-R :





## Descripción:

El modelaje diseñado para la aplicación de EPSAndes contempla varias entidades que reflejan la lógica del sistema, y para poder hacer el paso formal del UML y el ER al modelo relacional debemos seguir el algoritmo de de Peter Chen donde se debe realizar cierto paso a paso.

En el apartado de modelaje E-R a modo de primer paso se va a realizar la conversión de clases básicas, lo que conlleva a que cada clase independiente es candidata para ser una relación por ende vamos a convertirla en plural, y modelar sus atributos propios lo que implica determinar el tipo de atributo, verificar que es un atributo simple y consolidar las restricciones sobre estos. Agregado a lo ya mencionado es primordial para el algoritmo identificar las llaves candidatas y seleccionar la primary key (PK) de cada una de estas tablas. Próximo a ello se identifica cual es el tipo de relación en el nuestro se priorizo la 1 a 1, muchos a 1, muchos a muchos y auto loop, y para cada uno se tomó la decisión correspondiente, de asignar a una columna, agregar al multivariable, crear una nueva tabla o agregar el atributo propio respectivamente. En general realizamos paso 1, 2, 3 y 7 ya que evitamos al máximo herencias y los pasos de composición o adición fueron obviados, y después de este paso a paso logramos conseguir la modificación de UML a relacional que nos otorga el modelo de Chen.

Realizando el análisis se determinó que las IPS son el eje central, ya que representan las instituciones encargadas de proveer los servicios médicos a los afiliados, además nos permite reunir todos los conceptos ya que las EPS poseen IPS sin embargo para evitar el patrón fachada no se coloca EPS. Cada una está identificada por un NIT único y posee atributos como el nombre, la dirección y el teléfono de contacto. Una IPS puede ofrecer múltiples Servicios de Salud. Esto establece una relación de muchos a muchos con los servicios de Salud.

Los Servicios de Salud se identifican con un ID único y contienen información como su nombre y tipo de servicio.

Los Afiliados son los usuarios del sistema que acceden a los servicios de salud. Cada afiliado posee un número de identificación único, tipo de documento, nombre, fecha de nacimiento, dirección, teléfono y tipo de afiliado. Además, se ha implementado una relación de parentesco entre la misma entidad.

Las Órdenes de Servicio son documentos que se emiten por un médico hacia un afiliado. Cada orden se identifica con un ID único y tiene asociados atributos como la fecha, el estado de la orden y el tipo de servicio solicitado. Una orden está siempre vinculada a un único afiliado.

Los médicos prestan servicios dentro de las IPS. Cada médico se identifica con su número de documento y tiene atributos como el nombre, el tipo de documento y la especialidad. Un médico debe estar vinculado a una o más IPS. Además, los médicos prescriben Órdenes de Servicio, lo que genera una relación de uno a muchos entre Médico y Orden de Servicio.

## **Normalización:**

### **Primera Forma Normal (1NF)**

La **Primera Forma Normal (1NF)** se refiere a la atomicidad de los datos y a la existencia de una llave primaria que identifique de manera única cada fila en una tabla. En este modelo, se ha logrado cumplir con esta norma al asegurar que todos los atributos sean atómicos y que cada tabla tenga una PK bien definida.

Por ejemplo, en la tabla IPSs, la columna NIT actúa como llave primaria, lo que garantiza que cada fila sea única. Además, los números de teléfono, que podrían haber sido almacenados como una lista en una sola columna, se han normalizado en una tabla separada llamada Telefonos\_IPS. En esta tabla, cada número de teléfono se almacena en una fila individual, asociado al NIT de la IPS correspondiente. Esto asegura que no haya columnas con múltiples valores, cumpliendo así con la regla de atomicidad. De manera similar, en la tabla Afiliados, los números de teléfono se han normalizado en la tabla Telefonos\_Afiliados, donde cada número de teléfono está asociado al numeroDocumento del afiliado. Este enfoque garantiza que los datos sean atómicos y que no haya redundancias innecesarias o listas en los atributos.

### **Segunda Forma Normal (2NF)**

La **Segunda Forma Normal (2NF)** exige que todos los atributos no primos dependan completamente de la llave primaria, evitando dependencias parciales. En nuestro modelo esto se cumple ya que se busca evitar al máximo las dependencias parciales usando en su mayoría llaves diferenciadas como PK lo que lleve a cumplir la segunda forma normal.

Por ejemplo, en la tabla Médicos, los atributos nombre, tipoDocumento y especialidad dependen completamente de la llave primaria numeroDocumento. En la tabla Afiliados, los atributos nombre, fechaNacimiento, direccion y telefonos dependen completamente de la llave primaria numeroDocumento. Esto asegura que no haya dependencias parciales, cumpliendo así con la 2NF, en todos los casos se busca un PK único, como en las otras tablas con un id.

Además, en las tablas Teléfonos IPS y Telefonos\_Afiliados, las llaves primarias son compuestas (NIT\_tel, telefono y numDocumento, telefono, respectivamente), pero los atributos no primos dependen completamente de estas llaves compuestas.

### **Tercera Forma Normal (3NF)**

La **Tercera Forma Normal (3NF)** va un paso más allá y exige que no haya dependencias transitivas entre atributos no llave. Es decir, ningún atributo no clave debe depender de otro atributo no clave, sino únicamente de la clave primaria. En este modelo, se ha logrado cumplir con esta norma al asegurar que todos los atributos no primos dependan directamente de la llave primaria.

Por ejemplo, en la tabla IPSs, los atributos nombre, dirección y teléfono dependen directamente de la llave primaria NIT. No hay atributos que dependan de otros atributos no llave, y así se mantiene en el resto de tablas, por ello se muestra que la 3NF se cumple sin dependencias transitivas.

### **Forma Normal de Boyce-Codd (BCNF)**

La **Forma Normal de Boyce-Codd (BCNF)** es una versión más estricta de la 3NF y exige que, para toda dependencia funcional  $X \rightarrow Y$ , X debe ser una superllave. En este modelo, se ha logrado cumplir con esta norma al asegurar que todas las dependencias funcionales están basadas en super llaves. Como cada tabla sólo depende de una llave primaria (o una llave compuesta que actúa como superclave), y no hay múltiples claves candidatas que se traslapan, se puede afirmar que el modelo está en BCNF. Esto se debe a que, al estar ya en 3NF y no tener dependencias funcionales problemáticas, el modelo cumple automáticamente con BCNF.

## **Escenarios de Prueba:**

### **1. Registrar un nuevo afiliado**

**Descripción:** Se intenta registrar un nuevo afiliado con datos completos.

**Datos de entrada:** ID\_Afiliado, TipoDocumento, NumeroDocumento, Nombre, FechaNacimiento, Dirección, Teléfono, TipoAfiliado.

**Resultado esperado:** El sistema almacena correctamente la información y el afiliado aparece en el sistema.

### **2. Intentar registrar un afiliado con documento duplicado**

**Descripción:** Se intenta registrar un afiliado con un número de documento ya existente en la base de datos.

**Resultado esperado:** El sistema rechaza el registro e informa al usuario sobre la duplicidad del documento.

3. **Consultar la información de un afiliado existente**

**Descripción:** Se busca un afiliado por su número de documento.

**Resultado esperado:** El sistema muestra la información correcta del afiliado.

4. **Registrar una IPS nueva**

**Descripción:** Se registra una nueva IPS con toda la información necesaria.

**Datos de entrada:** NIT, Nombre, Dirección, Teléfono.

**Resultado esperado:** La IPS se almacena correctamente en la base de datos y está disponible para asignación de servicios.

5. **Consultar una IPS por su NIT**

**Descripción:** Se ingresa un NIT para buscar una IPS existente.

**Resultado esperado:** El sistema muestra la información correspondiente.

6. **Intentar registrar una IPS con un NIT ya registrado**

**Descripción:** Se intenta registrar una IPS con un NIT que ya existe en la base de datos.

**Resultado esperado:** El sistema rechaza el registro y muestra un mensaje de error.

7. **Registrar un médico**

**Descripción:** Se intenta registrar un médico con datos completos.

**Datos de entrada:** ID\_Medico, Nombre, TipoDocumento, NumeroDocumento, Especialidad.

**Resultado esperado:** El médico se almacena correctamente en el sistema.

8. **Consultar médicos asignados a una IPS**

**Descripción:** Se selecciona una IPS y se listan los médicos asociados.

**Resultado esperado:** Se muestra correctamente la lista de médicos asociados a la IPS seleccionada.

9. **Asignar un médico a una IPS**

**Descripción:** Se selecciona un médico y se le asigna a una IPS existente.

**Resultado esperado:** El médico queda vinculado a la IPS correctamente.

10. **Registrar un nuevo servicio de salud**

**Descripción:** Se intenta registrar un servicio con su información completa.

**Datos de entrada:** ID\_Servicio, Nombre, Descripción, TipoServicio.

**Resultado esperado:** El servicio se almacena correctamente en el sistema.

11. **Consultar servicios disponibles en una IPS**

**Descripción:** Se selecciona una IPS y se listan los servicios que ofrece.

**Resultado esperado:** Se muestra la lista de servicios correctamente.

12. **Intentar registrar un servicio de salud con nombre duplicado**

**Descripción:** Se intenta registrar un servicio con el mismo nombre que otro existente en la base de datos.

**Resultado esperado:** El sistema rechaza la operación y muestra un mensaje de error.

13. **Generar una orden de servicio para un afiliado**

**Descripción:** Se intenta generar una orden de servicio para un afiliado con datos correctos.

**Datos de entrada:** ID\_Orden, Fecha, EstadoOrden, TipoServicio.

**Resultado esperado:** La orden de servicio se genera y queda registrada en estado "Vigente".

14. **Consultar órdenes de servicio de un afiliado**

**Descripción:** Se selecciona un afiliado y se listan sus órdenes de servicio.

**Resultado esperado:** Se muestra la lista de órdenes de servicio correctamente.

15. **Completar una orden de servicio**

**Descripción:** Se intenta marcar una orden como "Completada".

**Resultado esperado:** El sistema actualiza correctamente el estado de la orden.

**16. Cancelar una orden de servicio vigente**

**Descripción:** Se intenta cancelar una orden que aún no ha sido completada.

**Resultado esperado:** El sistema cambia el estado de la orden a "Cancelada".