UNIVERSIDAD DE MÁLAGA

E.T.S. INGENIERÍA INFORMÁTICA
UNIVERSIDAD DE MÁLAGA

Trabajo Almacenes de Datos

Diseño y Explotación de un almacén de UCI Sanitaria

Realizado por
De Pablo Diego y Soriano Juan

Profesor encargado:
Luque Baena Rafael Marcos

Departamento
Lenguajes y Ciencias de la Computación

MÁLAGA, noviembre de 2024

UNIVERSIDAD DE MÁLAGA

E.T.S. INGENIERÍA INFORMÁTICA
UNIVERSIDAD DE MÁLAGA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA

ESTUDIANTES DE INGENIERÍA BIOINFORMÁTICA

Diseño y Explotación de un almacén de UCI Sanitaria

Almacenes de Datos

Realizado por
De Pablo Diego y Soriano Juan

Profesor encargado:
Navas Luque Baena Rafael Marcos

Departamento
Lenguajes y Ciencias de la Computación

UNIVERSIDAD DE MÁLAGA

MÁLAGA, NOVIEMBRE DE 2024

Contents

1 Introducción

El diseño y desarrollo de **almacenes de datos** es crucial para el análisis clínico, mejorando la toma de decisiones médicas y la calidad de atención a los pacientes. Para este proyecto, se utilizará una visión parcial de un almacén de datos basado en la *Base de Datos de Investigación Colaborativa eICU*, la cual contiene una vasta cantidad de información sobre los ingresos en **Unidades de Cuidados Intensivos (UCI)** en diversos hospitales de los Estados Unidos.[1]

El enfoque de este trabajo está dirigido específicamente a los **pacientes con problemas respiratorios**. Estos casos son de especial relevancia en entornos de cuidados críticos, ya que las afecciones respiratorias representan una de las principales causas de ingreso en las UCI. Por lo cual un almacén de datos centrada en estas patologías sigue representando una gran fuente de información y con mucho interés para hacer un análisis de datos. [4, 5].

En este trabajo, el **hecho principal** está constituido por los ingresos en la UCI, complementado por un conjunto de dimensiones que permiten analizar diversas variables. La selección de las tablas y atributos más relevantes de la base de datos eICU será el mayor reto para construir un **almacén de datos eficiente y de interés analítico**, que no solo facilite la consulta de información clave, a su vez soporte y facilité tanto la investigación clínica como la optimización de los protocolos de tratamiento.

2 Objetivos

El objetivo de este trabajo es desarrollar un **diseño conceptual y lógico** de un almacén de datos centrado en los pacientes ingresados en la **UCI** con afecciones respiratorias, seleccionando las partes más relevantes de la base de datos para el análisis de estas patologías.

Se procederá a la restauración de dicha base de datos en *SQL Server*, para realizar una selección de las tablas más relevantes que contribuyan a la construcción del almacén de datos centrado en estos pacientes. En cada una de las tablas seleccionadas, se identificarán los atributos más significativos para el análisis clínico y el seguimiento de los pacientes con problemas respiratorios.

Una vez completada la selección de dimensiones y tablas de hechos, se diseñará el modelo conceptual utilizando la herramienta *draw.io*, y posteriormente se desarrollará el modelo lógico mediante un diagrama de base de datos en el entorno de *SQL Server*.

3 Descripción de los datos utilizados (eICU)

La base de datos *eICU-CRD*, desarrollada por el Philips eICU Research Institute (eRI), contiene datos desidentificados de pacientes ingresados en Unidades de Cuidados Intensivos (UCI). Esta base de datos, que forma parte del programa de telesalud en cuidados críticos de Philips, documenta detalles sobre diagnósticos, tratamientos, pruebas y resultados.

La *eICU-CRD* recopila una amplia cantidad de datos clínicos con fines de investigación, permitiendo estudiar los resultados de los pacientes, identificar tendencias

clínicas y evaluar protocolos de mejores prácticas utilizados en distintos centros de atención médica. Esta base de datos tiene como fin facilitar la investigación para mejorar la calidad de la atención en UCI.

La *eICU-CRD* contiene un total de 31 tablas, que agrupan 391 columnas de datos y un total de 457,325,320 filas. Esta vasta cantidad de información abarca diversos aspectos clínicos, como datos demográficos de los pacientes, información clínica obtenida durante el ingreso, pruebas y diagnósticos realizados, tratamientos y terapias administradas, así como la evolución y resultados durante la estancia en la UCI. Dada la magnitud de los datos. En este trabajo, se ha optado por simplificar el almacén, enfocándose en aquellos datos más relevantes para el análisis de pacientes ingresados en la UCI con problemas respiratorios. A continuación, se presentan las 9 tablas seleccionadas del almacén original:

- `ApacheApsVar:` Contiene diversas métricas utilizadas para evaluar la gravedad de la enfermedad de los pacientes al ingreso en la UCI como parte del sistema APACHE.
- `Admissiondx:` Contiene el diagnóstico principal por el cual el paciente fue admitido en la UCI, de acuerdo con los criterios de puntuación APACHE.
- `Allergy:` Contiene información sobre las alergias del paciente.
- `Diagnosis`: Contiene los diagnósticos documentados para cada paciente. Permite identificar las enfermedades documentadas durante la estancia en UCI, así como el momento en que se registraron.
- `Hospital:` contiene información sobre los hospitales participantes en el programa y sus pacientes vinculados.
- `Medication:` Contiene las órdenes activas de medicamentos para los pacientes.
- `Patient:` Contiene los datos demográficos del paciente, así como los detalles de su admisión y alta del hospital y la UCI.
- `RespiratoryCare:` Contiene información sobre el cuidado respiratorio.
- `RespiratoryCharting:` Contiene datos sobre la configuración respiratoria, como el tipo de gráfico y valores respiratorios.

Además de reducir el número de tablas también se encontró oportuno reducir el número de atributos de las mismas, al contener información poco esencial o que no se ve utilidad para las siguientes fases del trabajo. (Vea la figura 1 para detallar las tablas y sus atributos de manera global)

4 Diseño conceptual

El diseño conceptual define las entidades, relaciones y requisitos de información de manera abstracta, sin detalles técnicos. Se enfoca en representar las necesidades y cómo organizar los datos para soportar la toma de decisiones.

Recordando que se modela un almacén que trata sobre una población de **pacientes con problemas respiratorios** las tablas elegidas son las siguientes:

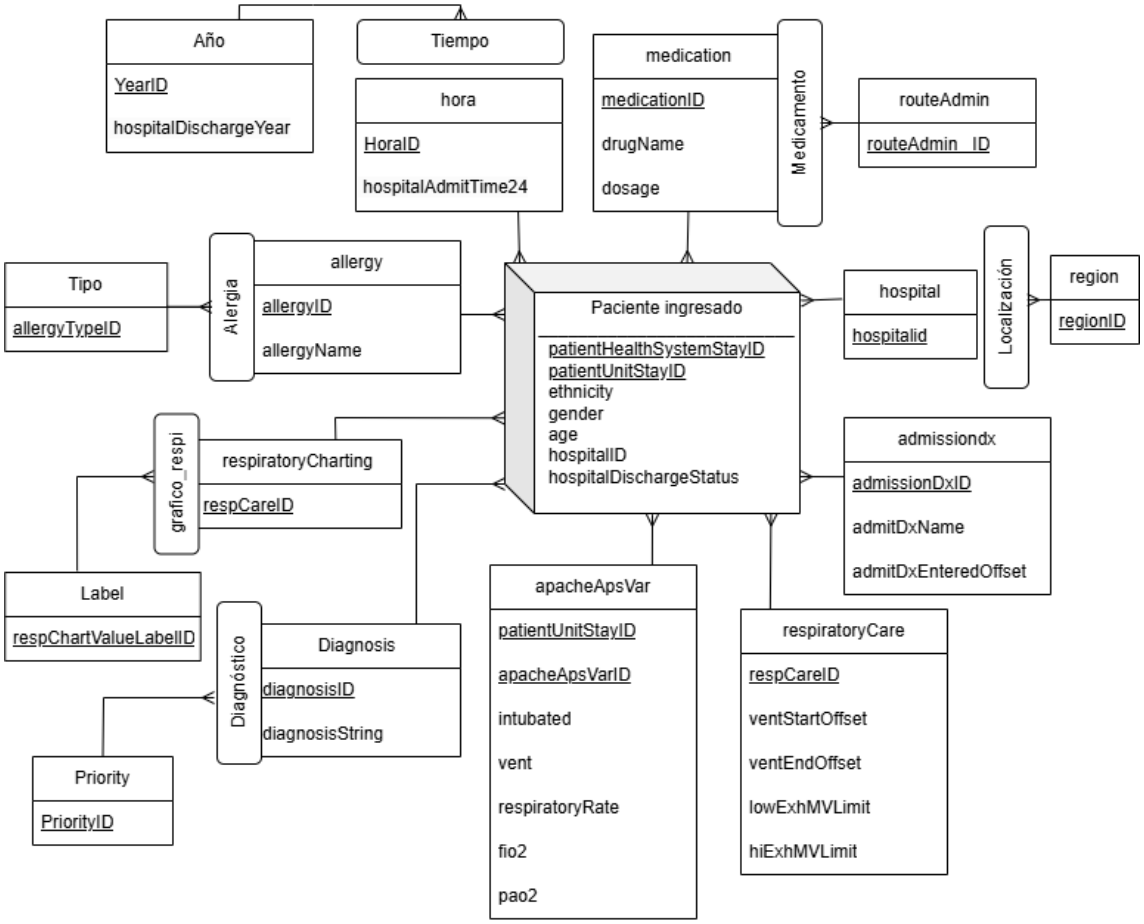

Figure 1: Diseño conceptual

4.1 Tablas

4.1.1 PacienteIngresado

- **Relevancia**: La tabla `pacienteIngresado` es el hecho que almacena información demográfica de los pacientes y detalles relacionados con sus ingresos y egresos del hospital y la UCI. Es fundamental para el análisis de la estancia en la UCI y la hospitalización, proporcionando una visión global de la evolución clínica y de los tiempos de atención.

- **Selección de atributos**:

  - `patientUnitStayID`: Clave primaria que identifica de manera única la estancia del paciente en la UCI. Relaciona con el registro de `patient` mediante este identificador.
  - `patientHealthSystemStayID`: Identificador del ingreso hospitalario, vincula las estancias hospitalarias de un paciente durante un mismo periodo.
  - `ethnicity`: Etnia del paciente, una variable importante para el análisis demográfico y la disparidad en la atención.
  - `gender`: Género del paciente, relevante para estudios epidemiológicos y de salud pública.

- `age`: Edad del paciente, crucial para la estratificación del riesgo y el análisis de comorbilidades.
- `hospitalID`: Identificador único del hospital, esencial para la segmentación por centro de atención.
- `hospitalDischargeStatus`: Estado del paciente al momento del alta hospitalaria, indicando si está vivo, fallecido, o en otro estado.
- `hospitalAdmitTime24`: Hora exacta del ingreso al hospital, importante para el análisis temporal de la atención.
- `hospitalDischargeYear`: Año de alta hospitalaria, relevante para la evaluación de tendencias a lo largo del tiempo.
- `UniquePID`: Identificador único del paciente, utilizado para distinguir a los pacientes en el sistema.

Algunos atributos no se han incluido debido a la duplicación o irrelevancia en el contexto del hecho pacienteIngresado. Por ejemplo, `hospitalAdmitTime24` y `hospitalDischargeYear` ya cubren la información temporal necesaria, por lo que atributos como `unitAdmitTime24` y `unitDischargeTime24` no son necesarios en este nivel, ya que hacen referencia a eventos más específicos de la UCI, no del ingreso hospitalario. Igualmente, `wardID` y `unitType` se han excluido porque son detalles más relevantes para el contexto de la unidad de UCI, no para el hecho principal del paciente ingresado. [2]

4.1.2 Diagnosis

- **Relevancia**: La tabla `Diagnosis` es clave para analizar diagnósticos de enfermedades respiratorias en pacientes, permitiendo filtrar y clasificar las condiciones por severidad.
- **Selección de atributos**:
  - `patientUnitStayID`: Relaciona el diagnóstico con un paciente específico en la UCI.
  - `diagnosisID`: Clave primaria para diferenciar cada diagnóstico.
  - `diagnosisString`: Descripción completa del diagnóstico para consultas específicas.
  - `diagnosisPriority`: Indica la prioridad del diagnóstico (Primario, Mayor, Otro).

No se han incluido atributos como `diagnosisOffset` e `ICD9Code` debido a que este proyecto se centra más en la naturaleza y prioridad del diagnóstico que en el tiempo específico de entrada o el código ICD-9. [2]

4.1.3 respiratoryCharting

- **Relevancia**: La tabla `RespiratoryCharting` es fundamental para el monitoreo de los valores respiratorios de los pacientes en UCI, especialmente en aquellos con enfermedades respiratorias graves.

- **Selección de atributos**:
  - `patientUnitStayID`: Asocia los datos respiratorios con un paciente específico.
  - `respCareID`: Identificador único del registro respiratorio.
  - `respChartValueLabel`: Describe el tipo de valor respiratorio (ej. HR, I:E Ratio), útil para categorizar los datos.

Se decidió que datos como `respChartOffset` y `respChartValue` añadían detalles muy específicos, estos se descartaron para enfocar la relevancia clínica general sobre estos detalles puntuales, ya que el objetivo es estudiar tendencias en vez del seguimiento minuto a minuto.[2]

4.1.4 respiratoryCare

- **Relevancia**: La tabla `RespiratoryCare` es clave para el análisis de los cuidados respiratorios proporcionados a los pacientes en la UCI, ya que permite evaluar las intervenciones de ventilación mecánica y otras terapias respiratorias.
- **Selección de atributos**:
  - `patientUnitStayID`: Asocia el cuidado respiratorio a un paciente específico.
  - `respCareID`: Identificador único para cada intervención de cuidado respiratorio.
  - `ventStartOffset`: Refleja el inicio de la ventilación, crucial para analizar la relación entre el inicio de la intervención y la evolución del paciente.
  - `ventEndOffset`: Indica el final de la ventilación, permitiendo estudiar la duración de las intervenciones respiratorias.
  - `lowExhMVLimit`: Establece el límite inferior del volumen minuto expiratorio, importante para evaluar la efectividad de la ventilación.
  - `hiExhMVLimit`: Define el límite superior del volumen minuto expiratorio, igualmente relevante para evaluar la capacidad de los sistemas respiratorios.

Se han excluido atributos como `airwayType` y `cuffPressure` porque el estudio se centra en parámetros que permiten observar el impacto de la ventilación mecánica en la evolución de los pacientes y no tanto en los detalles técnicos de cada intervención.[2]

4.1.5 apacheApsVar

- **Relevancia**: La tabla `apacheApsVar` es esencial para el cálculo del `Acute Physiology Score` (APS) III, un sistema ampliamente utilizado para evaluar la gravedad de la enfermedad de los pacientes al ingreso en la UCI. Esta puntuación es parte del sistema `APACHE` para predecir los resultados de los pacientes críticos.

- **Selección de atributos**:
  - `patientUnitStayID`: Relaciona a cada entrada de paciente en la UCI con su respectivo registro en la tabla de pacientes.
  - `apacheApsVarID`: Clave primaria que identifica de manera única cada conjunto de variables de `APACHE APS`.
  - `intubated`: Indica si el paciente fue intubado al momento de obtener el peor valor de gasometría (ABG), crucial para evaluar la necesidad de intervención respiratoria.
  - `vent`: Indica si el paciente fue ventilado al momento de registrar la peor frecuencia respiratoria, reflejando la gravedad de la insuficiencia respiratoria.
  - `respiratoryRate`: Refleja la frecuencia respiratoria más baja durante el período de `APACHE`, utilizada para medir la función respiratoria del paciente.
  - `fio2`: Mide la fracción inspirada de oxígeno, importante para evaluar la insuficiencia respiratoria del paciente y su respuesta al tratamiento con oxígeno.
  - `pao2`: Mide la presión parcial de oxígeno en sangre, un indicador clave de la gravedad de la hipoxia en los pacientes.

Atributos como la puntuación de `Glasgow (GCS)`, los valores de `creatinina`, `glucosa`, o `hematocrito`, aunque son importantes para el cálculo del puntaje APACHE, no están tan directamente relacionados al análisis de los problemas respiratorios. [2]

4.1.6 Hospital

- **Relevancia**: La tabla `hospital` es relevante ya que almacena información sobre los hospitales donde fueron ingresados los pacientes. Al estar vinculada mediante una clave foránea en `pacienteIngresado`, permite realizar análisis comparativos entre las instituciones médicas que forman parte del programa eICU.

- **Selección de atributos**:
  - `hospitalID`: Clave primaria que identifica de manera única cada hospital. Es fundamental para relacionar las estancias de los pacientes con el centro médico correspondiente.
  - `region`: Identificador de la región geográfica del hospital, útil para realizar análisis comparativos entre áreas geográficas.

  No se han incluido atributos como `numbedsCategory` y `teachingStatus`, ya que aunque pueden aportar información sobre el tamaño y la categoría del hospital, se consideran irrelevantes para el enfoque principal de este proyecto, que se centra más en los pacientes y su tratamiento que en las características institucionales de los hospitales. [2]

4.1.7 AdmissionDx

- **Relevancia**: La tabla `admissionDx` contiene el diagnóstico principal de admisión a la UCI, esencial para los cálculos del sistema de puntuación APACHE. Es fundamental para identificar la causa principal de ingreso y analizar cómo los diagnósticos iniciales impactan en la evolución de los pacientes.

- **Selección de atributos**:

  - `patientUnitStayID`: Identificador único que enlaza con la estancia del paciente en la UCI, crucial para asociar el diagnóstico de admisión con el paciente correspondiente.
  - `admissionDxID`: Clave primaria que identifica de manera única cada diagnóstico de admisión.
  - `admitDxName`: Nombre del diagnóstico de admisión, que proporciona una descripción del motivo clínico principal de ingreso.
  - `admitDxEnteredOffset`: Minutos desde la admisión a la unidad cuando se ingresó el diagnóstico, relevante para analizar la prontitud del diagnóstico al ingreso en la UCI.

No se han incluido atributos como `admitDxPath` y `admitDxText`, ya que se consideran menos relevantes para el enfoque de este proyecto, que prioriza el diagnóstico principal y el tiempo de entrada del mismo, sin profundizar en los detalles jerárquicos o valores amplificadores del diagnóstico. [2]

4.1.8 Allergy

- **Relevancia**: La tabla `allergy` contiene información sobre las alergias de los pacientes, incluyendo alergias a medicamentos, lo que resulta crucial para la gestión segura de tratamientos en la UCI.

- **Selección de atributos**:

  - `allergyID`: Clave primaria que identifica de manera única cada registro de alergia.
  - `patientUnitStayID`: Identificador único que enlaza con la estancia del paciente en la UCI, permitiendo asociar las alergias al paciente correspondiente.
  - `allergyType`: Tipo de alergia, indicando si es a un medicamento o no, un dato de gran importancia para la selección de tratamientos adecuados.
  - `allergyName`: Nombre de la alergia, que proporciona información específica sobre el alérgeno.

No se han incluido atributos como `allergyOffset`, `specialtyType` o `userType`, ya que se consideran menos relevantes al ser detalles adicionales sobre cuándo y por quién fueron registradas. [2]

4.1.9 Medication

- **Relevancia**: La tabla `medication` almacena información sobre los medicamentos prescritos para los pacientes en la UCI. Esto es fundamental para realizar análisis sobre los tratamientos administrados y su relación con los resultados clínicos de los pacientes.
- **Selección de atributos**:
  - `patientUnitStayID`: Identificador único que enlaza la estancia del paciente en la UCI, permitiendo asociar los medicamentos prescritos al paciente correspondiente.
  - `medicationID`: Clave primaria que identifica de manera única cada registro de medicamento.
  - `drugName`: Nombre del medicamento prescrito, proporcionando información sobre el tipo de tratamiento administrado.
  - `routeAdmin`: Ruta de administración del medicamento.
  - `dosage`: La dosis del medicamento, que permite realizar un seguimiento detallado de los tratamientos y evaluar su efectividad o posibles efectos adversos.

No se han incluido atributos como `drugOrderOffset`, `frequency` o `PRN`, al ser detalles temporales o la frecuencia con la que fue recetado puede llegar aumentar la complejidad del estudio. [2]

Al crear un diagrama conceptual y modelar una base de datos específica, la reducción de información es clave. Este enfoque optimiza el espacio, simplifica el sistema y se enfoca en los datos relevantes, eliminando los innecesarios. La selección de datos en este modelo facilita la integración, análisis y las recomendaciones de estudios previos. Aunque esta implementación es adecuada para este estudio, existen enfoques alternativos que pueden considerar otros aspectos sin implicar una mayor o menor calidad.

5 Diseño Lógico - Copo de nieve

El **modelo lógico** en un almacén de datos define la estructura y relaciones entre tablas, organizando los datos de forma eficiente y sin redundancias, podría decirse que es la evolución del diseño conceptual previamente visto. En este proyecto, se ha implementado un **modelo en copo de nieve**, caracterizado por la alta normalización de las dimensiones. Esto descompone las tablas en varios niveles, reduciendo la duplicación de datos, lo que optimiza el espacio y mejora la consistencia, aunque podría aumentar la complejidad en las consultas.

El modelo lógico, representado en la figura 2, fue desarrollado utilizando *Oracle SQL Developer Data Modeler*. Este modelo originalmente contenía la estructura lógica del almacén de datos, y se ha aplicado un proceso de normalización para organizar las tablas de manera eficiente y reducir la redundancia. Se ha propuesto la normalización de tablas como `PacienteIngresado`, `Medication`, `RespiratoryCharting`, entre otras, exceptuando la dimensión de tiempo, que se ha mantenido denormalizada.

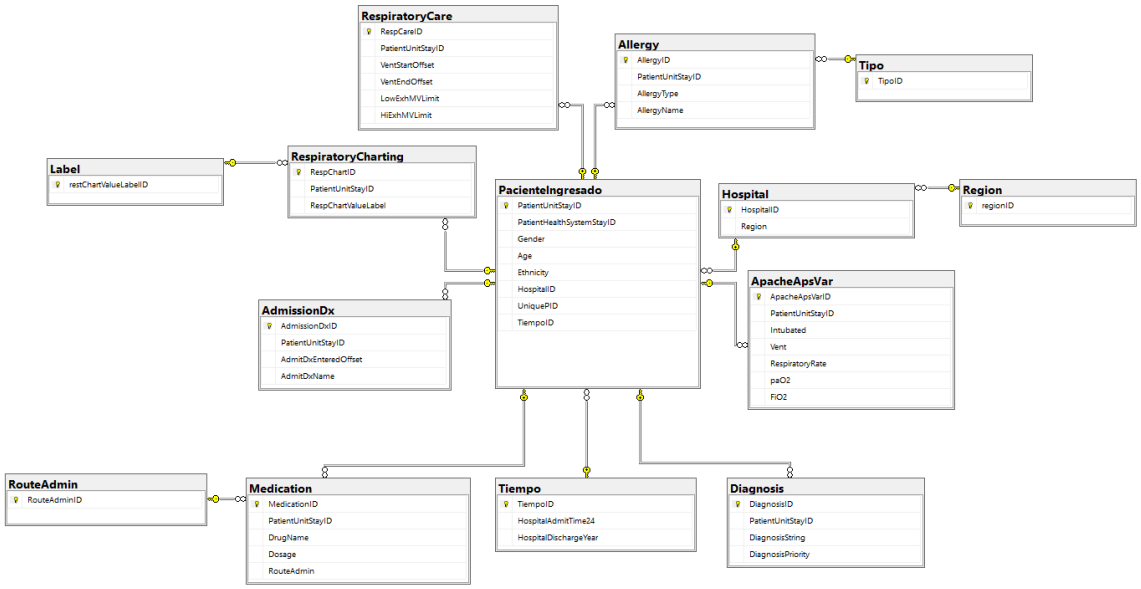

Figure 2: Diseño lógico

5.1 FK's

- **PatientUnitStayID** en:
  - `RespiratoryCare` → `PacienteIngresado.PatientUnitStayID`
  - `RespiratoryCharting` → `PacienteIngresado.PatientUnitStayID`
  - `AdmissionDx` → `PacienteIngresado.PatientUnitStayID`
  - `ApacheApsVar` → `PacienteIngresado.PatientUnitStayID`
  - `Medication` → `PacienteIngresado.PatientUnitStayID`
  - `Diagnosis` → `PacienteIngresado.PatientUnitStayID`

  Esto vincula las tablas que contienen información específica sobre los cuidados y condiciones del paciente con su estancia en UCI.

- **HospitalID** en:
  - `PacienteIngresado` → `Hospital.HospitalID`

  Relaciona cada estancia del paciente con el hospital correspondiente.

- **Region** en:
  - `Hospital` → `Region.regionID`

  Define la región en la que se encuentra el hospital.

- **RouteAdmin** en:
  - `Medication` → `RouteAdmin.RouteAdminID`

  Especifica la vía de administración de cada medicamento.

- **RespChartValueLabel** en:
  - `RespiratoryCharting` → `Label.restChartValueLabelID`

  Define el tipo de valor en el gráfico de respiración.

- **TiempoID** en:
  - `PacienteIngresado` → `Tiempo.TiempoID`

  Permite asociar información de tiempo como la admisión y el alta.

- **TipoID** en:
  - `Allergy` → `Tipo.TipoID`

  Clasifica el tipo de alergia mediante una FK hacia `Tipo`.

En lugar de incluir un ID genérico y un atributo que describa detalles (como en Region), optamos por almacenar el valor descriptivo directamente en regionID, eliminando la necesidad de otro campo y simplificando el modelo.

6 Dificultades encontradas

Una de las principales dificultades fue la complejidad de la base de datos eICU, que incluye una gran cantidad de tablas y atributos. Esto exigió un análisis detallado para identificar las tablas y campos clave en un modelo centrado en pacientes con enfermedades respiratorias. Además, enfrentamos problemas de permisos al intentar visualizar el modelo relacional en SQL Server, lo que requirió modificar las autorizaciones del propietario de la base de datos para acceder a los diagramas de relación.

Además el comienzo del trabajo podría ser lo más angustioso, al tener tanta información y opciones llega a ser un poco abrumador, desde la selección de una población concreta y modelar un almacén para dicha población termina dejando muchas dudas sobre cuantas tablas es esperable eliminar, si se esta simplificando de más o se esta tomando una decisión que afectará los siguientes apartados.

7 Conclusiones

A lo largo de este proyecto, se ha permitido superar múltiples retos asociados con el modelado y diseño de un almacén de datos a partir de la base de datos eICU. El análisis de las tablas y atributos permitió entender lo complejo que puede llegar a tener un sistema de salud, y llegando a apreciar la importancia de la creación de estructura sólida como el modelo copo de nieve, optimizando la organización y normalización de las dimensiones.

Uno de los principales desafíos fue gestionar la complejidad de la base de datos con una gran cantidad de tablas. Para abordar este problema, se ha realizado una cuidadosa selección de las tablas clave centradas en los pacientes con afecciones respiratorias, lo que permitió reducir la cantidad de información irrelevante una vez realizado un amplió estudió de trabajos similares.

Entre los logros más destacados se encuentra la correcta integración de las dimensiones relacionadas con el tiempo, los medicamentos y las características respiratorias, claves para el estudio de la población seleccionada. La estructura final, respaldada por un diseño conceptual y lógico sólido, se ha mantenido alineada con el objetivo de ofrecer un modelo flexible que permita futuras ampliaciones y análisis específicos. A pesar de las dificultades con los permisos de acceso en SQL Server y la complejidad en la selección de las tablas más relevantes, hemos logrado completar un modelo escalable y aplicable en escenarios clínicos reales. Este almacén de datos no solo facilita un análisis detallado de los pacientes, sino que sienta las bases para la incorporación de nuevos datos y la expansión del análisis hacia otros perfiles clínicos.

8 Github

Todo el proyecto está accesible en github [3]

References

[1] Chamsi Bah, Sultan Alharthi, and Ashraf El Metwally. "Clinical data warehousing: A review of current systems and future directions". In: *Journal of Healthcare Engineering* 2017 (2017), pp. 1–11. DOI: 10.1155/2017/8326740.

[2] eICU Collaborative Research Database. *eICU Collaborative Research Database*. https://eicu-crd.mit.edu/about/eicu/. Accessed: 2024-11-14. 2024. URL: https://eicu-crd.mit.edu/about/eicu/.

[3] Diegodepab. *Almacén UCI Sanitaria*. https://github.com/Diegodepab/almacen_UCI_Sanitaria. Accessed: 2024-11-14. 2024. URL: https://github.com/Diegodepab/almacen_UCI_Sanitaria.

[4] Christopher Pekar, Sven Gordan, and Ewan Goligher. "Epidemiology of respiratory failure in the intensive care unit: A review". In: *Critical Care* 25.1 (2021), pp. 1–9. DOI: 10.1186/s13054-021-03772-y.

[5] Jean-Louis Vincent, Yasser Sakr, and V Marco Ranieri. "The epidemiology of respiratory failure in the ICU". In: *Chest* 129.1 (2006), pp. 90–99. DOI: 10.1378/chest.129.1.90.

UNIVERSIDAD DE MÁLAGA | uma.es

E.T.S de Ingeniería Informática
Bulevar Louis Pasteur, 35
Campus de Teatinos
29071 Málaga

E.T.S. DE INGENIERÍA INFORMÁTICA