Sistemas de información para el análisis de grandes volúmenes de datos 2024

Grupo 16

FACULTAD DE INGENIERÍA
UDELAR

Integrantes:
Miguelángel Díaz - 5.214.428-4

Introducción

Este informe busca documentar el desarrollo del laboratorio de la materia “Sistemas de información para el análisis de grandes volúmenes de datos”. El objetivo es implementar un sistema para analizar datos relacionados con la educación en Uruguay, utilizando conjuntos de datos disponibles en el Catálogo Nacional de Datos Abiertos de Uruguay [1].

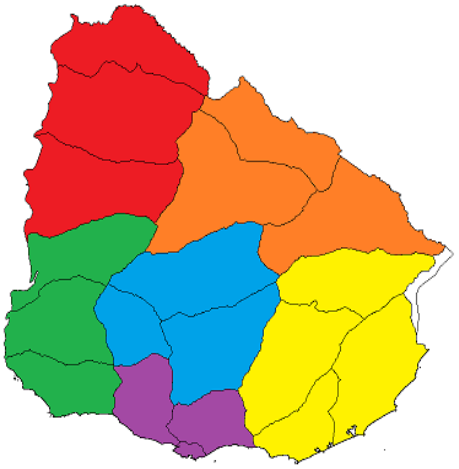
Se trabajará con datos de escuelas y liceos sobre ofertas educativas de educación primaria y secundaria, incluyendo información así como la cantidad de docentes y estudiantes en cada departamento del país. También se analizará el uso de la plataforma CREA del Plan Ceibal y de la biblioteca por parte de docentes y estudiantes.

1. Análisis de requerimientos y Diseño Conceptual

Para esta parte del laboratorio, se opta por la solución presentada por los docentes en el EVA del curso con una única diferencia en la dimensión ubicación, se puede consultar el diagrama de la misma en el anexo.

1.1 Requerimiento 1

Según el SINAE [2]. Nuestro país se puede dividir en las zonas mostradas en la Figura 1. Se utilizará una agrupación de intersección vacía basada en esta división para la zona-país, esto permitirá definir *Zona-país* superior a *Departamento* en la jerarquía de la dimensión *Ubicación*, lo cual ayuda a simplificar el sistema.



[Figura 1] Zonas del país definidas según el SINAE

Además, las localidades pueden pertenecer solo a un departamento [3], esto también hace mucho más sencilla la definición de la jerarquía agregando las mismas por debajo del Departamento.

Se busca información sobre los códigos postales [4] y se utiliza un algoritmo para saber si las localidades pueden tener más de un código postal, buscando por cada código postal y registrando si el mismo está en dos localidades diferentes (se debe tener en cuenta el departamento porque hay localidades con el mismo nombre en distintos departamentos). Sin embargo, al correr otro script que nos dice si un código postal pertenece solo a un departamento, indica que, por ejemplo, el código postal 50200 pertenece tanto a Salto como a Artigas, por lo tanto, no se puede incluir en la jerarquía y como se puede ver en el

diagrama, se opta por dejar la misma queda fuera de la jerarquía Ubicación -> Localidad -> Departamento -> Zona-país.

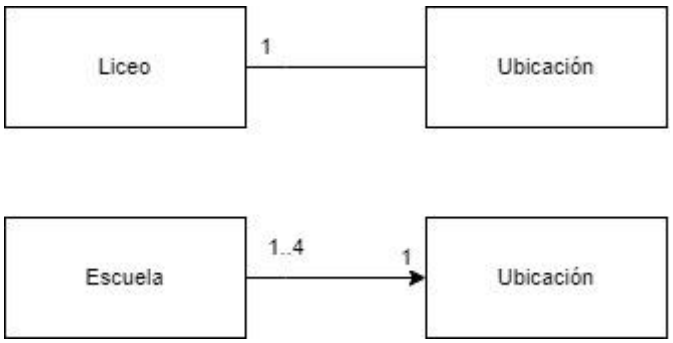
Nota: Una vez se publican los datos de la consigna, se comprueba sobre el conjunto de datos obteniendo los mismos resultados:

- Búsqueda de Localidad y Departamento sobre datos CEIP y CES y en los datos de direcciones para cada departamento.
- Se comprueba en los datos de direcciones que efectivamente el código postal 50200 está en Salto y Artigas.

Se realiza una búsqueda sobre *"CES-2024.xlsx" con el objetivo de determinar si los liceos pueden tener más de una ubicación*, se asume que el nombre identifica al liceo ya que este tiene la localidad; el mismo se toma como identificador para saber si está asignado a uno o más pares *Calle - Nro_de_puerta*, lo cual da resultado negativo, al igual que en la escuela; al buscar de forma inversa, es decir, para una ubicación verificar si esta está asignada a más de un liceo, obtenemos que no es así, y por lo tanto **un liceo está asignado a una ubicación, al igual que una ubicación está asignada a un solo liceo**.

1.2 Requerimiento 2

Se repite el procedimiento sobre "*CEIP-2024.xlsx*, siguiendo el mismo esquema que para los liceos con el objetivo de determinar si las escuelas pueden tener más de una ubicación, en esta búsqueda se obtiene un resultado distinto, en el que, si bien una escuela tiene una sola ubicación, una . Por lo tanto, se asume que **una dirección puede alojar a varias escuelas** en distinto turno, como se puede ver en las filas 21 y 22 de dicho archivo.



[Figura 2] Diagrama UML que representa las relaciones entre los distintos tipos de centros educativos y las ubicaciones de nuestro sistema

1.3 Requerimiento 3

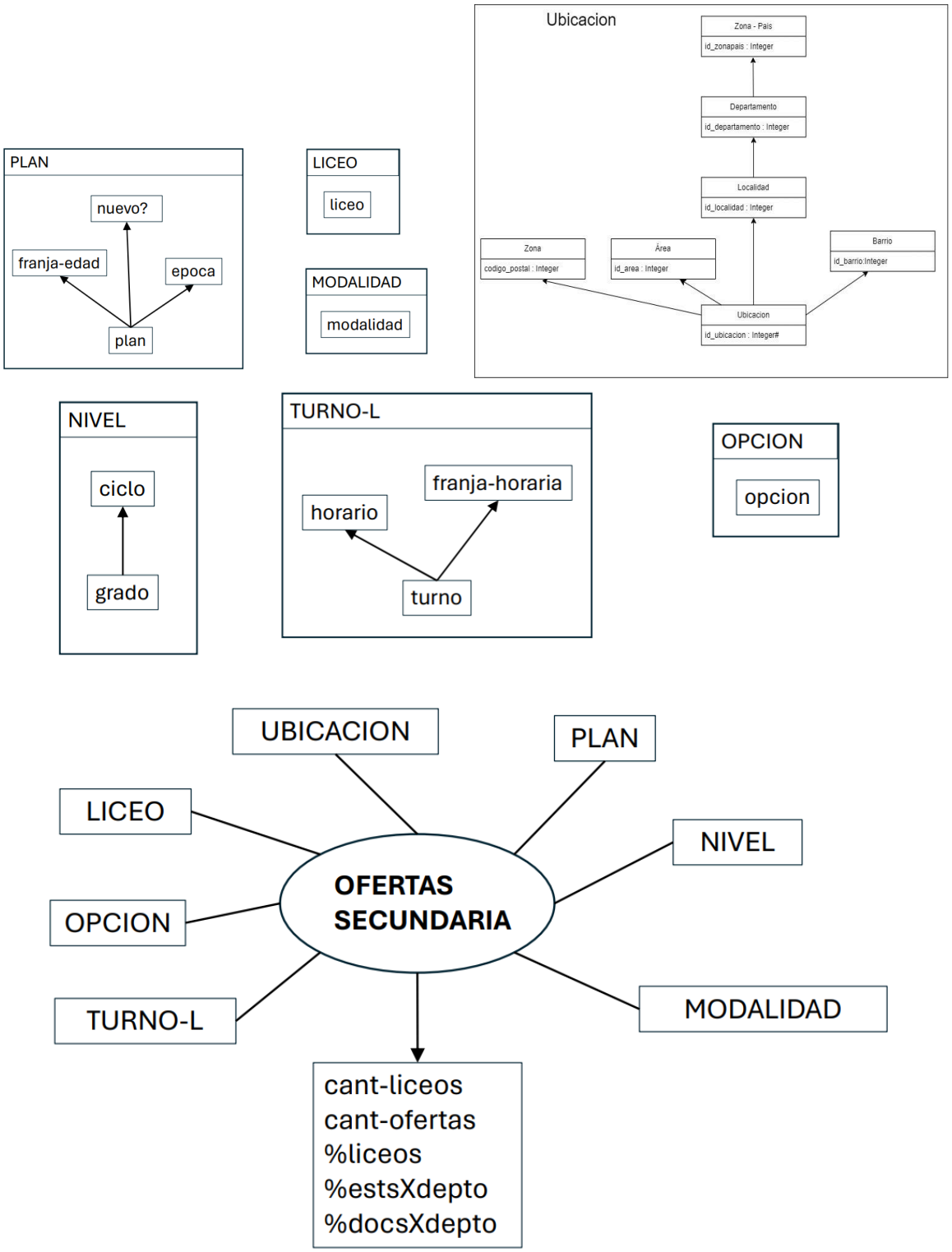
Finalmente, para cumplir con este requerimiento debemos analizar dos nuevas entidades. los subsistemas y los turnos generales. El subsistema es simplemente un indicativo de si el centro es de primaria o secundaria, se considerará en este caso mutuamente excluyente, es

decir, que una ubicación solo puede albergar o bien una escuela o bien un liceo, sin la posibilidad de albergar ambos. Luego, aplicamos el mismo razonamiento que cuando pasamos a representar la escuela, permitiéndole al Centro tener una única ubicación pero a una ubicación pero que este solo pueda tener un centro.

Definimos así la dimensión Centro, con el agregado de subsistemas. Mientras, el turno general es un turno con una semántica distinta tanto al de las escuelas como al de los liceos, por lo tanto, se considerará una nueva dimensión para el mismo. Considero entonces para este requerimiento las dimensiones Centro y Turno General.

1.4 Diseño conceptual

1.4.1 Requerimiento 1



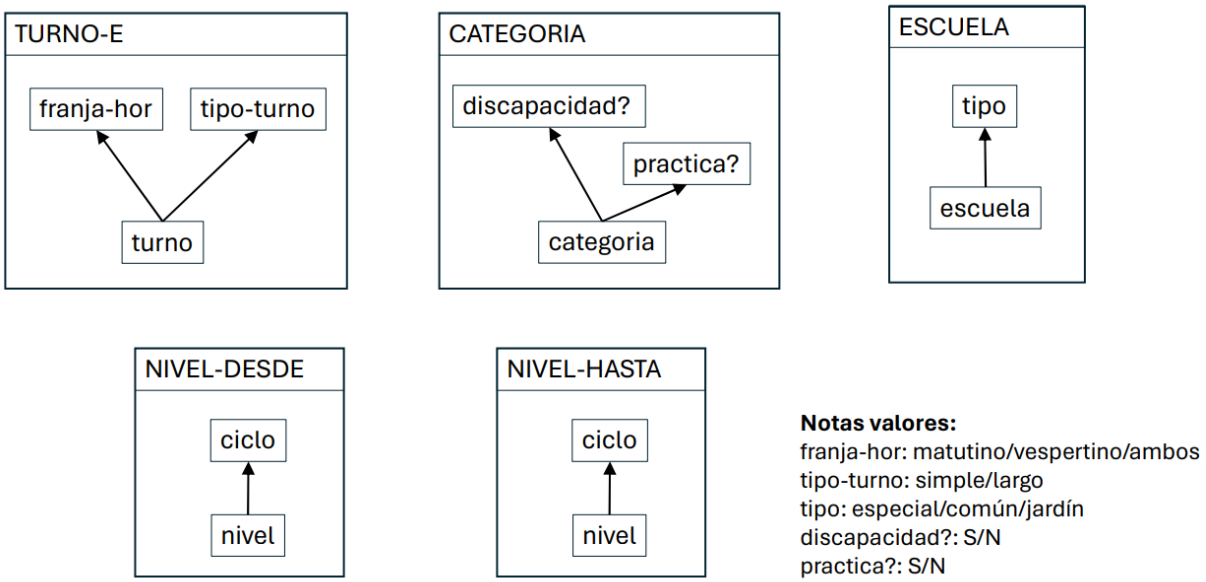
[Figura 3] Diseño conceptual del requerimiento 1, presentada por el equipo docente del curso en la solución. Incluye el cambio comentado en la sección 1.1

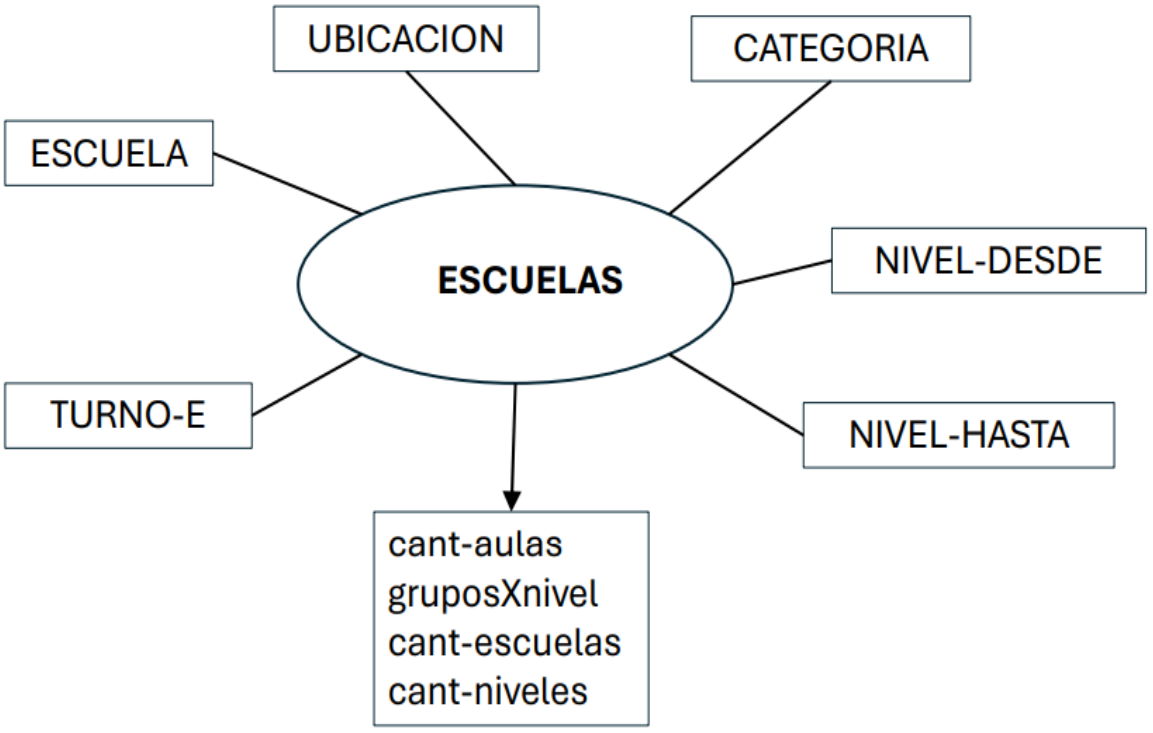
1.4.1.1 Aditividad del Requerimiento

		Medidas				
Dimensión	**Jerarquía**	cant-liceos	cant-ofertas	%liceos	% ests x depto	% docs x depto
Liceo	Liceo -> All	+	+	+	NA	NA
Ubicación	Ubicación -> Localidad	+	+	+	NA	NA
	Localidad -> Departamento	+	+	+	NA	NA
	Departamento -> Zona-País	+	+	+	+ , prom	+ , prom
	Zona-País -> All	+	+	+	+ , prom	+ , prom
	Ubicación -> Zona	+	+	+	NA	NA
	Ubicación -> Área	+	+	+	NA	NA
	Ubicación -> Barrio	+	+	+	NA	NA
	Barrio -> All	+	+	+	NA	NA
	Zona -> All	+	+	+	NA	NA
	Área -> All	+	+	+	NA	NA
TurnoL	Turno -> Horario	NA	+	NA	NA	NA
	Horario -> FranjaHoraria	NA	+	NA	NA	NA
	FranjaHoraria -> All	NA	+	NA	NA	NA
Plan	Plan -> EpocaAprobacion	NA	+	NA	NA	NA
	Plan -> FranjaEdad	NA	+	NA	NA	NA
	Plan -> Tipo	NA	+	NA	NA	NA
	EpocaAprobacion -> All	NA	+	NA	NA	NA
	Plan -> All	NA	+	NA	NA	NA
	FranjaEdad -> All	NA	+	NA	NA	NA
Grado	Grado-> Ciclo	NA	+	NA	NA	NA
	Ciclo -> All	NA	+	NA	NA	NA
Modalidad	Modalidad -> All	NA	+	NA	NA	NA

Opcion	Opcion -> All	NA	+	NA	NA	NA

1.4.2 Requerimiento 2



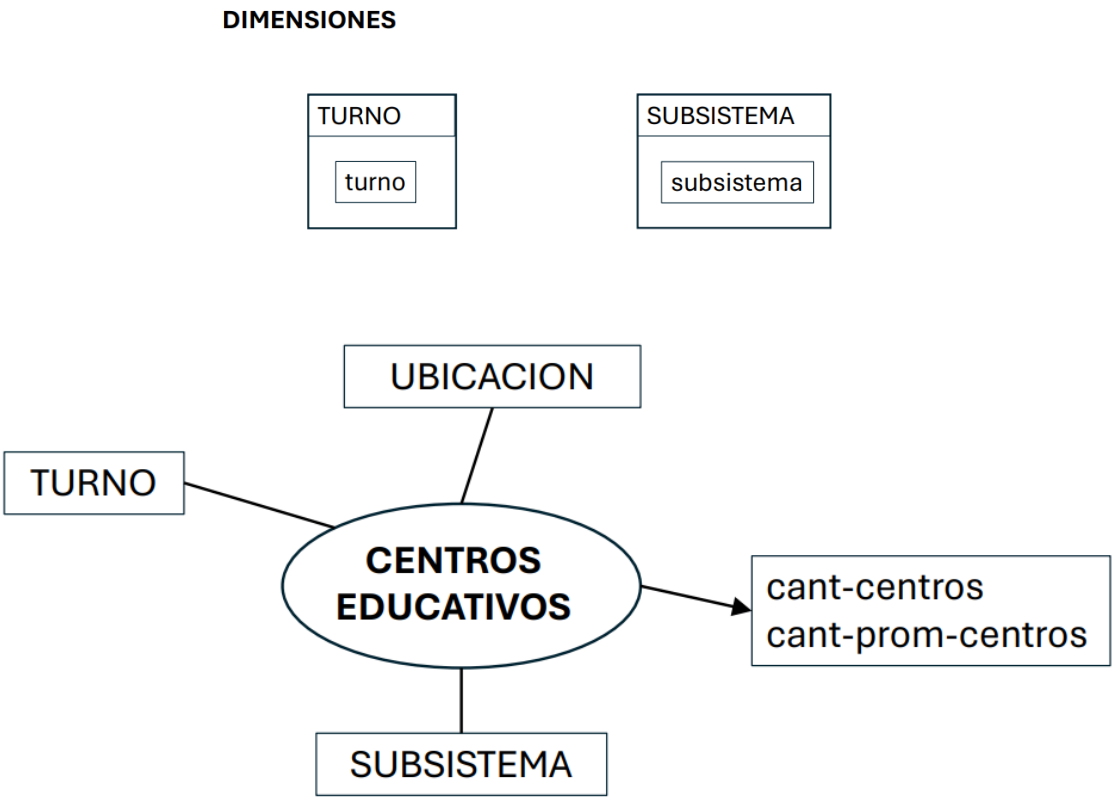


[Figura 4] Diseño conceptual del requerimiento 2, presentada por el equipo docente del curso en la solución. La dimensión Ubicación coincide con la presentada en la sección 1.4.1

1.4.2.1 Aditividad del Requerimiento 2

		Medidas			
Dimensión	**Jerarquía**	cant-aulas	gruposXnivel	cant-escuelas	cant-niveles
Escuela	Escuela -> Ubicación	+, prom	NA, prom	+, prom	NA, prom
Ubicación	Ubicación -> Localidad	+, prom	NA, prom	+, prom	NA, prom
	Localidad -> Departamento	+, prom	NA, prom	+, prom	NA, prom
	Departamento -> Zona-País	+, prom	NA, prom	+, prom	NA, prom
	Ubicación -> Zona	+, prom	NA, prom	+, prom	NA, prom
	Ubicación -> Área	+, prom	NA, prom	+, prom	NA, prom
	Ubicación -> Barrio	+, prom	NA, prom	+, prom	NA, prom
Escuela	Escuela -> Tipo	+, prom	NA, prom	+, prom	NA, prom
	Tipo -> All	+, prom	NA, prom	+, prom	NA, prom
Categoría	Categoría -> Especialización	+, prom	NA, prom	+, prom	NA, prom
	Categoría -> Practica	+, prom	NA, prom	+, prom	NA, prom
	Practica -> All	+, prom	NA, prom	+, prom	NA, prom
	Especialización -> All	+, prom	NA, prom	+, prom	NA, prom
Nivel Desde	NivelDesde-> Ciclo	+, prom	NA, prom	+, prom	NA, prom
	Ciclo -> All	+, prom	NA, prom	+, prom	NA, prom
Nivel Hasta	NivelHasta-> Ciclo	+, prom	NA, prom	+, prom	NA, prom
	Ciclo -> All	+, prom	NA, prom	+, prom	NA, prom
TurnoE	Turno -> Horario	NA, prom	NA, prom	+, prom	NA, prom
	Horario -> FranjaHoraria	NA, prom	NA, prom	+, prom	NA, prom
	FranjaHoraria -> All	NA, prom	NA, prom	+, prom	NA, prom

1.4.3 Requerimiento 3



[Figura 5] Diseño conceptual del requerimiento 3, presentada por el equipo docente del curso en la solución. La dimensión Ubicación coincide con la presentada en la sección 1.4.1

		Medidas	
Dimensión	**Jerarquía**	cant.centros	Cant-prom-centros
Ubicación	Ubicación -> Localidad	+	NA
	Localidad -> Departamento	+	NA
	Departamento -> Zona-País	+	NA
	Ubicación -> Zona	+	NA
	Ubicación -> Área	+	NA
	Ubicación -> Barrio	+	NA
	Zona-Pais -> All	+	NA
Subsistema	Subsistema -> All	NA	NA
TurnoGeneral	TurnoGeneral -> All	NA	NA

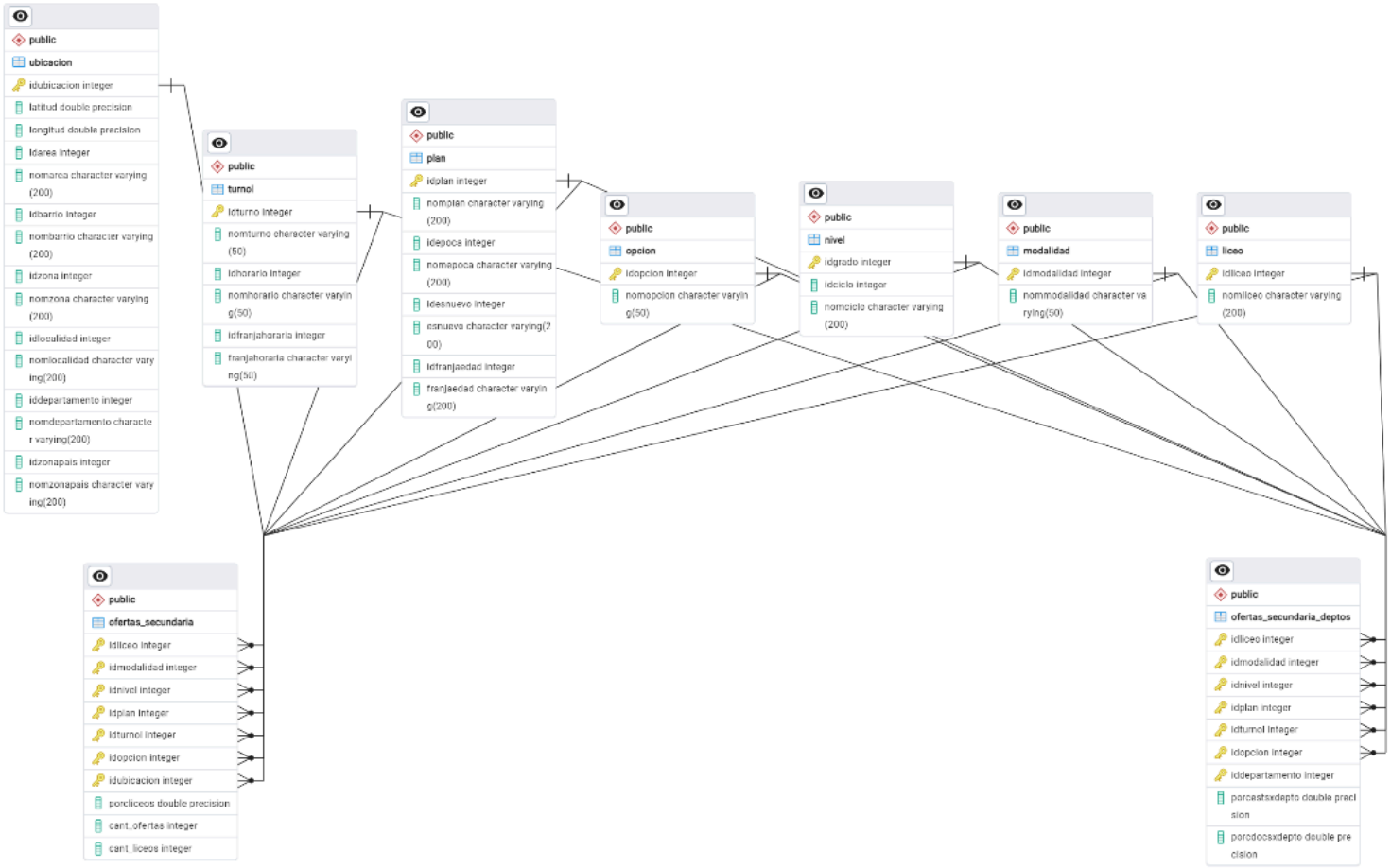
2. Diseño lógico

El diseño lógico está sometido a dos requisitos no funcionales principales, los cuales son utilizar PostgreSQL para implementar los cubos de los requerimientos 1 y 2 mientras que se define un esquema en mondrian para la definición de los cubos virtuales. En el caso del requerimiento 3, se debe implementar un modelo no relacional basado en alguno de los esquemas dados en el curso, en este caso se elige M2 como modelo (al igual que en la solución propuesta por el equipo docente).

Los requerimientos 1 y 2 son implementados siguiendo el esquema que se muestra en la sección 1.2, se define una tabla de agregación `OFERTAS_SECUNDARIA_DEPTOS` para simplificar el esquema, se almacenan en esta tabla las medidas que corresponden a porcentajes por departamento.

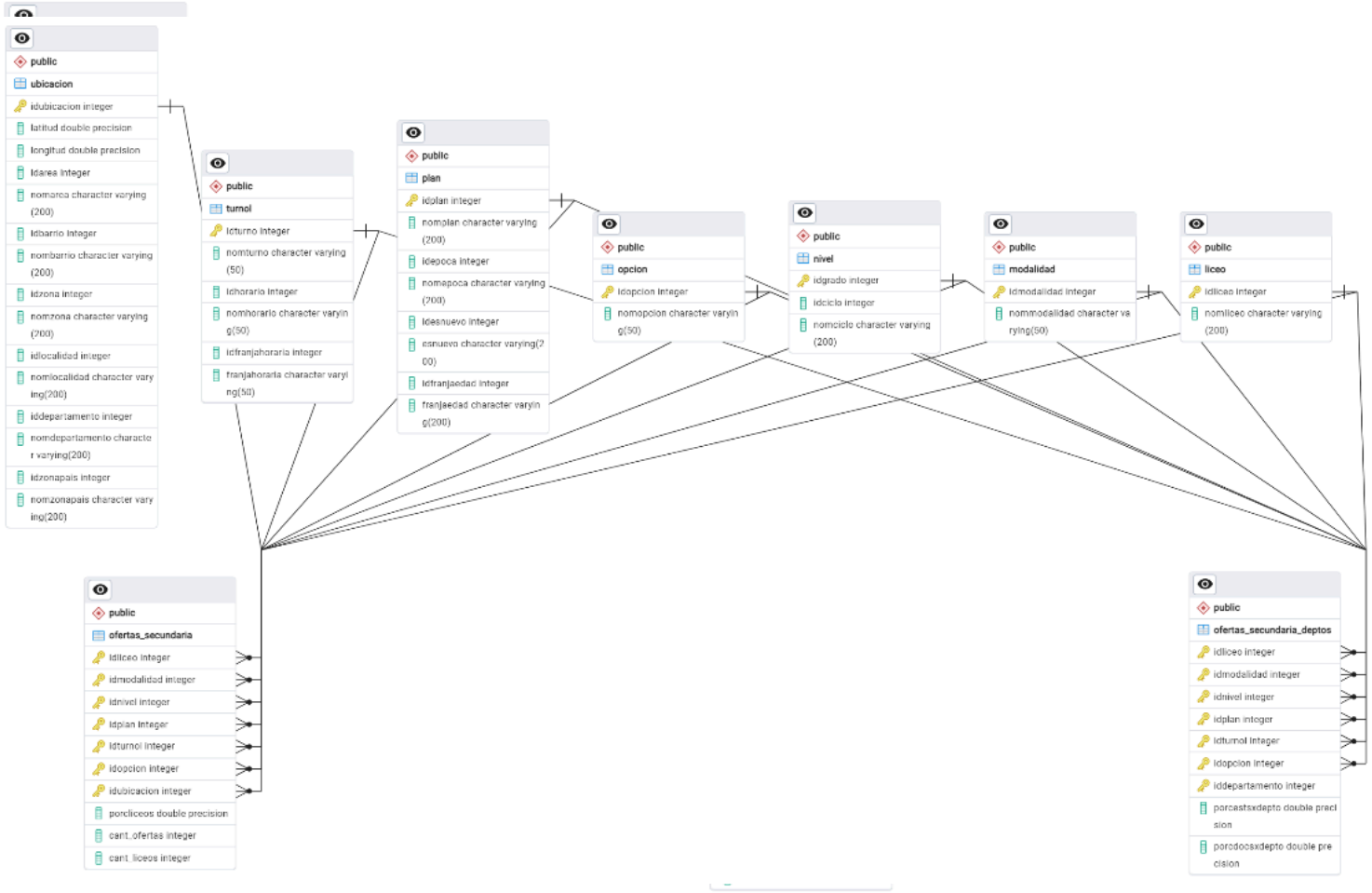
Se separan las ubicaciones de los dos distintos subsistemas en dos tablas `UBICACION` para el caso de los liceos y `UBICACIONE` para las escuelas, esto con el objetivo de simplificar los algoritmos de kettle, a la hora de implementar el Requerimiento 3 se tienen en cuenta las ubicaciones de ambas tablas.

2.1 Requerimiento 1



[Figura 6] Esquema de la base de datos PostgreSQL para el requerimiento 1

2.2 Requerimiento 2



[Figura 7] Esquema de la base de datos PostgreSQL para el requerimiento 2

2.3 Requerimiento 3 (no relacional):

2.3.1 Dimensiones

$D_1 = \{N_1, A_1, H_1\}$
$N_1 = \{\text{"Subsistema"}\}$
$A_1 = \{\text{idSubsistema,nomSubsistema}\}$
$H_1 = \{\text{Subsistema}\}$

$D_2 = \{N_2, A_2, H_2\}$
$N_2 = \{\text{"Turno"}\}$
$A_2 = \{\text{idTurno,nomTurno}\}$
$H_2 = \{\text{Turno}\}$

$D_3 = \{N_3, A_3, H_3\}$
$N_3 = \{\text{"Ubicacion"}\}$
$A_3 = \{\text{idUbicacion,latitud,longitud,}$
$\text{idArea,nomArea,idBarrio,nomBarrio,idZona,nomZona,idLocalidad, nomLocalidad,}$
$\text{idDepartamento,nomDepartamento,idZonaPais,nomZonaPais}\}$
$H_3 = \{\text{Ubicacion,Area,Barrio,Zona,Localidad,Departamento,ZonaPais}\}$

2.3.2 Hechos

$F_1 = \{N_1, M_4\}$
$N_1 = \{\text{"Centro"}\}$
$M_4 = \{\text{cant\_centros,cant\_prom\_centros,cant\_ubicaciones}\}$

Se elige el modelo M2, el cual, según el artículo de la tarea, para nuestra realidad, consiste en:

$S_F = \{id_F\} \cup M_4 \cup \{id_{D_1}, id_{D_2}, id_{D_3}\}$
$= \{id_{Centro}, \text{cant\_centros,cant\_prom\_centros,cant\_ubicaciones}, id_{Subsistema}, id_{Turno}, id_{Ubicacion}\}$

$S_{Subsistema} = A_1$
$S_{Turno} = A_2$
$S_{Ubicacion} = A_3$

Ejemplo: A continuación definimos S_{Centro}, donde C^{Centro} es la colección correspondiente.

{
"idCentro": 1,
"cantCentros": 1,
"cantPromCentros": 1,
"cantUbicaciones": 1
} $\in C^{Centro}$

{ "idSubsistema": 1,
"nomSubsistema": "DGEIP"
} $\in C^{Subsistema}$
"idTurno": 1,
"nomTurno": "MATUTINO”
} $\in C^{Turno}$
{
"idUbicacion": 1,
"latitud": -34.90536,
"longitud": -56.19059,
"idArea": 1,
"nomArea": ”URBANA",
"idBarrio": 2,
"nomBarrio": "Centro",
"codigoPostal": 11100,
"idLocalidad": 1,
"nomLocalidad": "MONTEVIDEO",
"idDepartamento": 1,
"nomDepartamento": "MONTEVIDEO",
"idZonaPais": 1,
"nomZonaPais": "METROPOLITANA"
} $\in C^{Ubicacion}$

3. Implementación

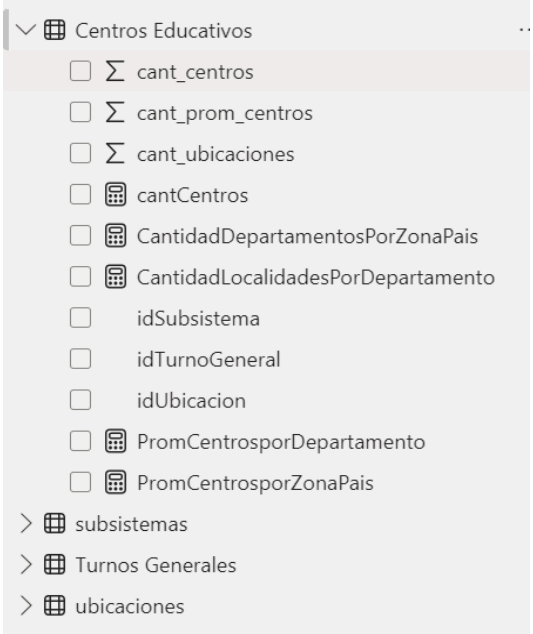
Se utilizó postgresql como servidor de base de datos como se indicaba en los requerimientos no funcionales de la consigna, se provee el archivo de creación de la base de datos en el directorio `base_sql\create_database.sql`. A su vez, se utiliza mondrian para crear los esquemas de los cubos virtuales y visualizer para crear tableros interactivos.

3.1 Requerimientos 1 y 2

Para estos dos primeros requerimientos, se menciona en la letra que se deben presentar los archivos XML que indican las dimensiones y las relaciones dimensionales. Se presentan estos archivos en el directorio *`mondrian/requerimiento1.xml`* y *`mondrian/requerimiento2.xml`*

Con el objetivo de simplificar el sistema a la hora de implementar, en el caso del requerimiento 1 se implementa un cubo para las medidas: %liceos, Cant-liceos y Cant-ofertas y otro cubo que implementa las medidas de cantidad de estudiantes y docentes por departamento. Sin embargo, se entiende que se desestima esa práctica, por lo que se comenta en esta sección y no se modifica el diagrama presentado en la sección anterior debido a que se considera el más correcto porque, si bien en este caso no tenemos muchos datos, si se tratara de un sistema con BigData, no sería óptimo implementar dicho cambio.

3.2 Requerimiento 3



[Figura 8] Consulta PowerBI para el Requerimiento 3

Se implementa el tercer requerimiento utilizando PowerBI debido a la facilidad de uso respecto a los archivos JSON utilizados para la carga. La consulta mostrada en la Figura 8 permite ingresar las dimensiones a utilizar y permite crear medidas mediante consultas en DAX para manejar los problemas de aditividad.

Las Dimensiones en la Figura 8 son Subsistema, TurnoGeneral y Ubicación, respetando la definición del esquema M2 presentado en la sección 2.3. El subsistema y los turnos generales contienen la ID y el nombre, mientras que la ubicación contiene Area, Zona, Barrio y Localidad, Departamento y Zona País.

- Se calculan las medidas presentadas en el análisis de aditividad
  - Se implementa una consulta en DAX (Ver Figura 9) para solucionar el problema de aditividad en cantidad de centros que implica el tener varios turnos para un centro. Se utiliza SUMX que permite sumar el valor en el segundo parámetro (En este caso 1) por cada resultado del primer parámetro, en el que se encuentra el sumario de idSubsistema e idUbicacion que funciona como un count distinct.

```
cantCentros = SUMX(SUMMARIZE
('CentrosEducativos','CentrosEducativos'[idSubsistema],'CentrosEducativos'
[idUbicacion]),1)
```

[Figura 9] Consulta DAX para cantCentros del requerimiento 3

4. Proceso de carga

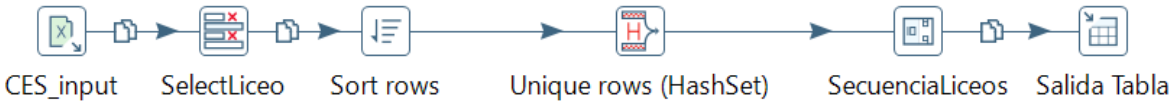
El proceso de carga está definido por las transformaciones que se encuentran en el directorio **`transformaciones/`**. El procedimiento que se sigue es el siguiente:

- Primero, se crean diferentes archivos JSON en los cuales se definen las distintas agrupaciones que no estarán en los datos, por ejemplo, como se puede observar en la Figura 10, para las zonas del país obtenidas como se describió en la sección 1, se define un JSON que lista las entidades del nivel inferior en la jerarquía (es decir, los departamentos) y les asigna una Zona-País junto con un código a cada una de dichas instancias. Esto permite que el sistema sea más amigable con una configuración externa, ya que, como se mencionará en la sección 6

```
[
    {
      "Departamento": "DURAZNO",
      "ZonaPais": "Centro",
      "idZonaPais": 1
    },
    {
      "Departamento": "FLORES",
      "ZonaPais": "Centro",
      "idZonaPais": 1
    }, …
]
```

*[Figura 10] Archivo JSON que contiene la definición de Zona-Pais, disponible en **`entrada\zonas-pais.json`***

- Posteriormente, como se puede ver en la Figura 11, se seleccionan los campos relevantes de cada dimensión, se ordenan en función del campo a tener en cuenta, en este caso el nombre, luego se les aplica una función para eliminar valores repetidos y posteriormente se les asigna una secuencia **mediante Kettle**. Esta estructura se repite en todas las dimensiones de todos los requerimientos, variando únicamente en sí se deben cargar otros campos (por ejemplo, zonas-pais). Una posible mejora a dicha metodología es el uso de secuencias definidas en la base de datos, ya que permitiría una mejor integración y manejo a la hora de cargar nuevas tuplas.



[Figura 11] Secuencia de procesamiento para definir los valores de cada dimensión

- Finalmente, se crean las Fact Table tomando como input las tablas cargadas en la base postgres y asignando de forma inversa a como se generaron las secuencias. Esto tiene como principal ventaja la modularidad de los pasos para facilitar la corrección de errores mediante la posible manipulación de los datos de forma separada en los pasos de normalización o estandarización y posterior generación de tabla de hechos.
  Sin embargo, cabe destacar la principal desventaja de este método que es la alta redundancia a nivel de algoritmo, es decir, se cargan los datos en las tablas para posteriormente volver a tomarlos para producir las tablas de hechos, esta sobrecarga puede no ser bienvenida en un paradigma de tipo big data y se deben reevaluar los algoritmos presentados si se trata de tal contexto.

Otro tema que se puede sacar de factor común sobre los tres requerimientos es sobre la obtención de los códigos postales y los barrios para las ubicaciones, en estos casos se utiliza la API [6] del sistema único de datos de direcciones geográficas del Uruguay [7].

4.1 Requerimiento 1

Respecto a este requerimiento, se implementa rigurosamente el procedimiento antes descrito:
Se definen las transformaciones `Liceo.ktr`, `Modalidad.ktr`, `Nivel.ktr`, `Opcion.ktr`, `Plan.ktr`, `Turno.ktr`, `Ubicaciones.ktr`, a su vez, se crea `niveles-ciclos.json` para definir los ciclos de cada nivel, `plan-epoca.json` para definir las épocas de los planes y `zonas-pais.json` para indicar a qué zona país pertenece cada departamento. Finalmente, como paso número 2 se utiliza la transformación `FactTableR1.ktr` para crear la tabla de hechos de este cubo.

4.2 Requerimiento 2

Se definen las transformaciones `Categoria.ktr`, `Escuelas.ktr`, `NivelE.ktr`, `TurnoE.ktr`, `UbicacionEscuelas.ktr` además de los archivos JSON `escuelas-categoria-tipos.json` para configurar los tipos de categoría en las escuelas, `escuelas-niveles.json` para determinar cuáles son los ciclos de cada nivel y `escuelas-turnos-tipos.json` y `escuelas-turnos-franjas.json` para definir las agrupaciones sobre los turnos de las escuelas. Se carga la tabla de hechos mediante la transformacion `FactTable2.ktr`.

4.3 Requerimiento 3

Para implementar este requerimiento se opta por utilizar archivos JSON con el objetivo de simplificar el trabajo, ya que como se mencionó en la sección 2.3, el programa de frontend elegido permite un manejo sencillo e intuitivo de este tipo de archivos, lo cual supone el no tener que incluir una tecnología adicional de base de datos al proyecto. Para obtener los subsistemas, estos se indican de forma manual ya que son dos, en el archivo *`salida\r3\subsistemas.json`*.

Se define el mapeo de turnos presentado previamente como un JSON *entrada\turnos-generales.json* en el que cada turno, ya sea de escuela o liceo es relacionado con un turno de los cuatro indicados en la propuesta, los cuales se pueden encontrar en *salida\r3\turnos_generales.json* a estos, se les agrega un valor auxiliar “Sin Dato” para considerar los valores vacíos de los datos de entrada.

Para las ubicaciones, se opta por tomar las de los dos primeros requerimientos y unirlas generando una nueva secuencia de ubicaciones. *salida\r3\ubicaciones.json*. En este caso se define una única transformación para generar los datos en la que se implementa lo ya descrito: `transformaciones\Requerimiento3.ktr`.

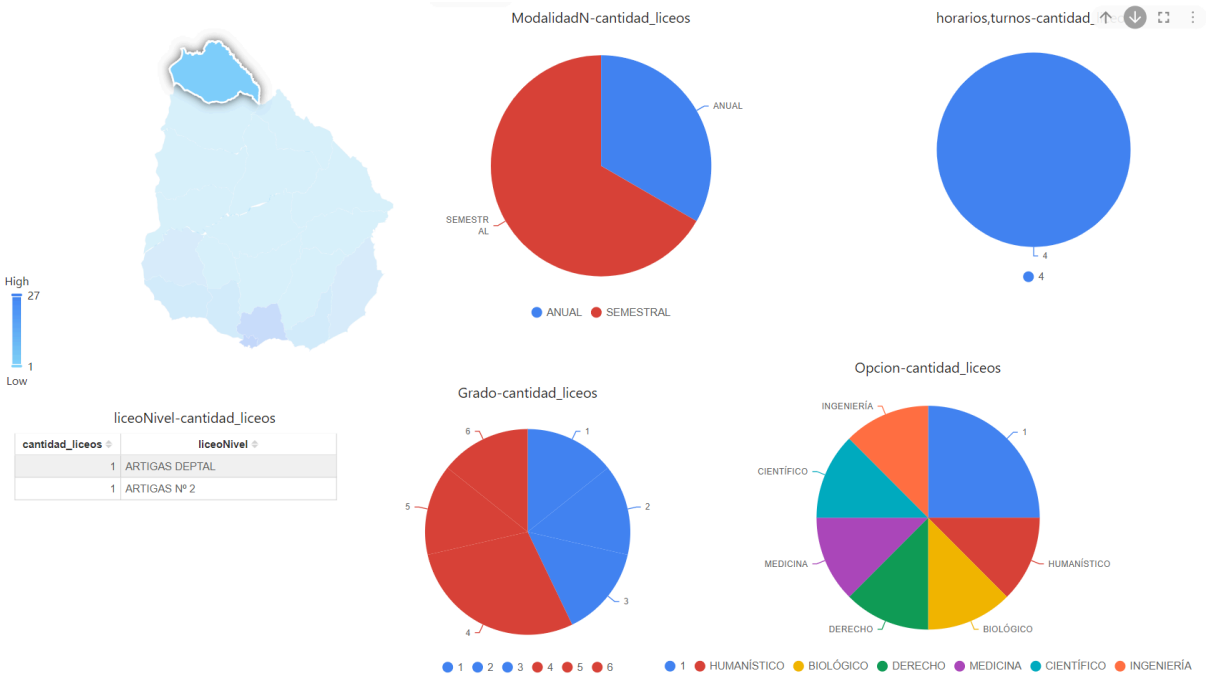
5. Front-end

Se implementan dashboards sencillos que permiten apreciar el funcionamiento de las medidas y poder hacer apreciaciones sobre si estas tienen sentido y se contrastan con los datos fuente.

5.1 Requerimiento 1

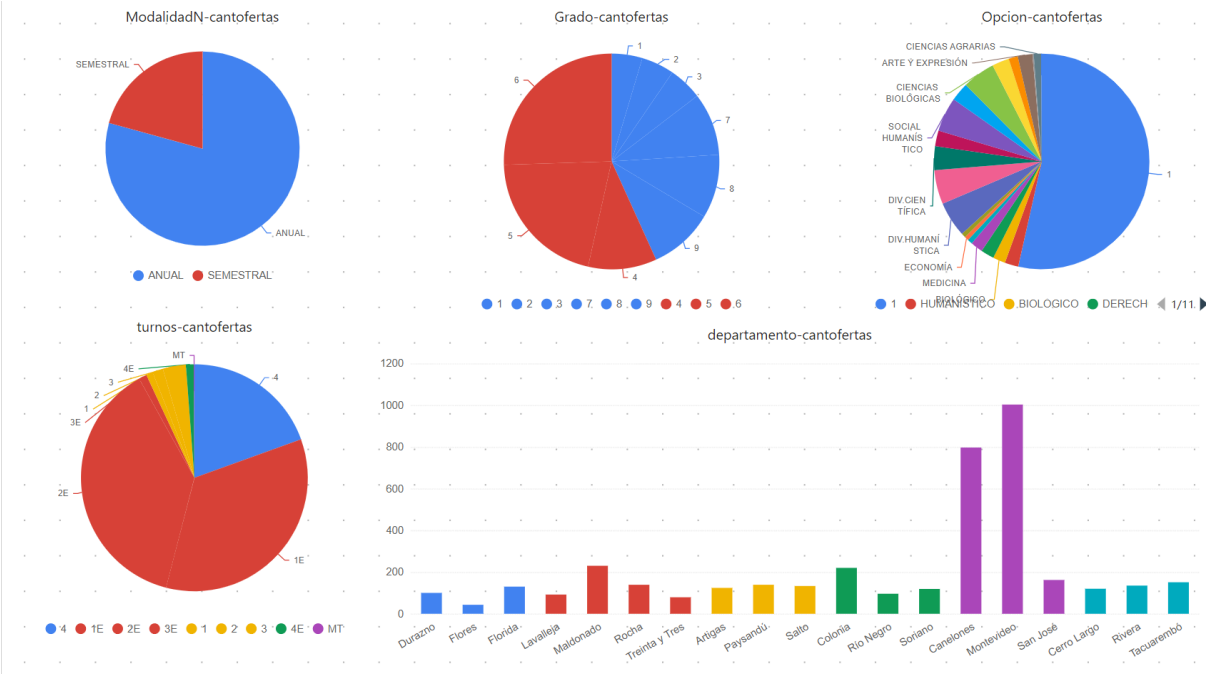
Se opta por utilizar Visualizer desde pentaho para poder realizar las consultas a los cubos de mondrian. En los tableros, se utiliza la leyenda para mostrar el nivel más específico, mientras que el color se emplea para indicar el más general, por ejemplo, en el caso del grado, se utiliza color azul para indicar ciclo básico y rojo en bachillerato.

En la Figura 12 se muestra el tablero que contiene la información sobre la cantidad de liceos, en este se puede apreciar un mapa interactivo que permite seleccionar un departamento, en la tabla de la izquierda se muestran los resultados de la consulta, se consulta por departamento y se hace drill up por turno, eligiendo turno nocturno, se puede observar que hay dos liceos en Artigas que satisfacen esta consulta.



[Figura 12] Tablero para la consulta del requerimiento 1 en cantidad de liceos

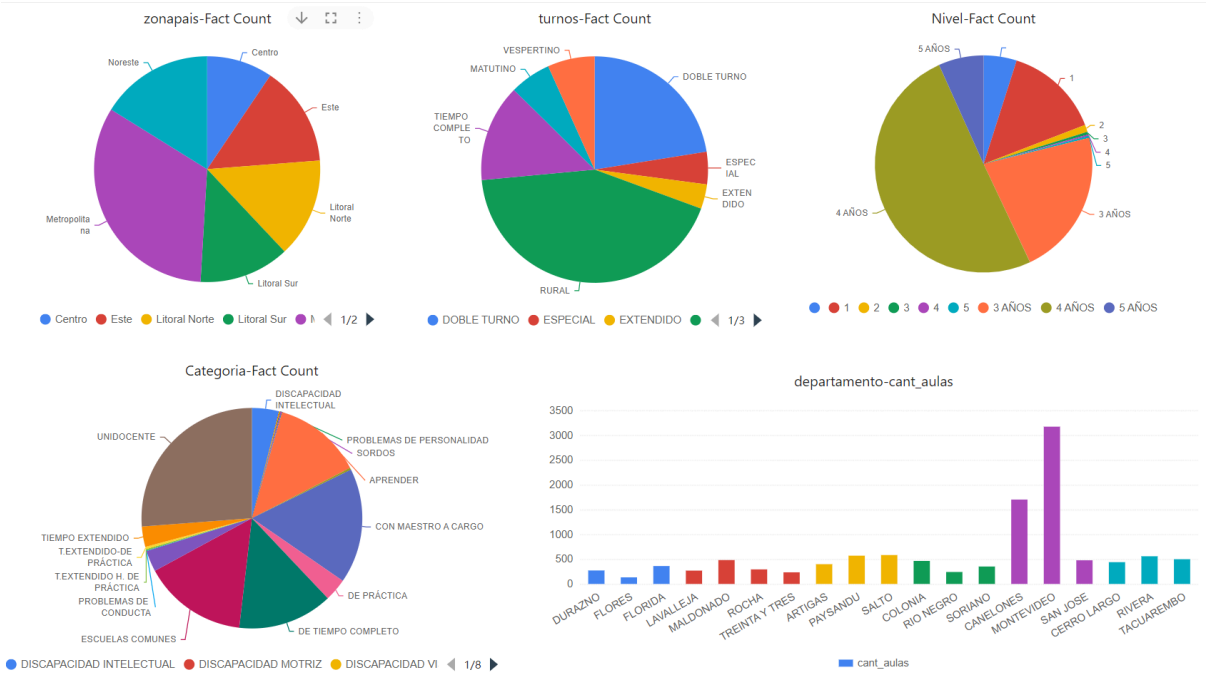
En la figura 13, se muestra el tablero que implementa la consulta de cantidad de ofertas, en este caso se utiliza un gráfico de barras para mostrar la cantidad por departamento.



[Figura 13] Tablero para la consulta del requerimiento 1 en cantidad de ofertas

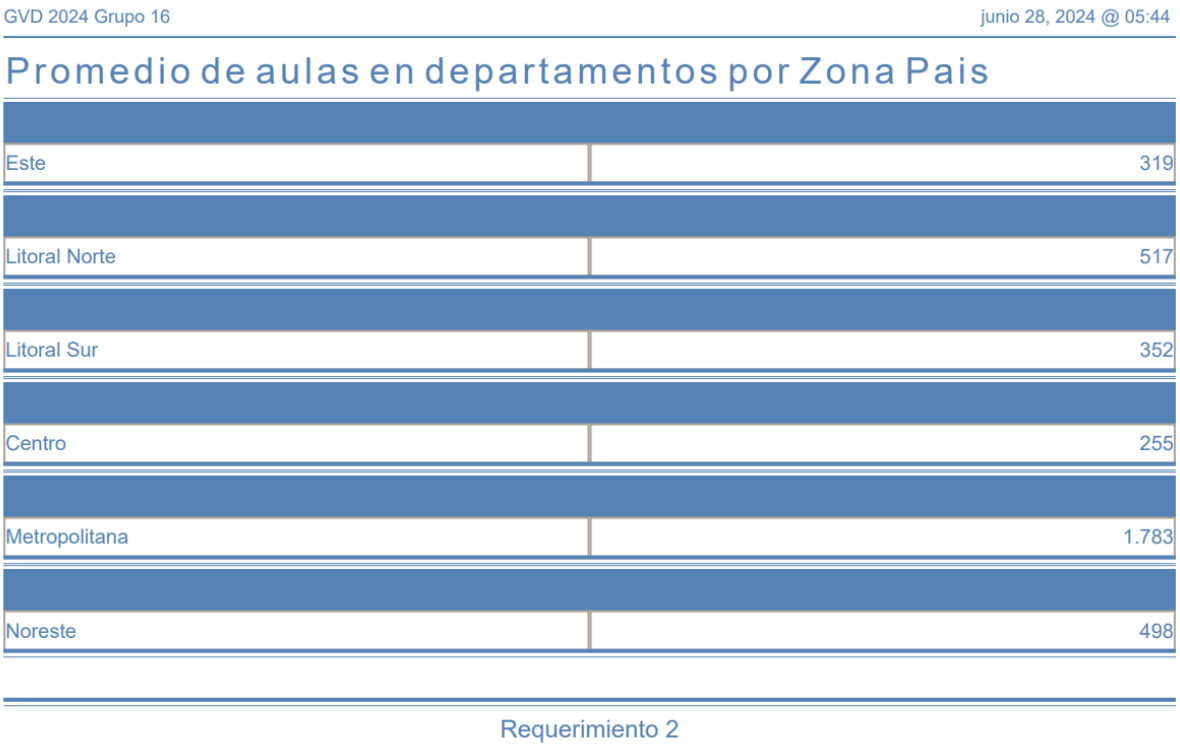
5.2 Requerimiento 2

Se implementó un tablero con la misma estructura comentada en la sección 5.1, como podemos observar en la Figura 14, hay una predominancia importante de los 3 y 4 años en el nivel de inicio de las ofertas, seria interesante ejecutar una consulta sobre los datos crudos que nos diga cómo se relacionan estos datos y si es verdad que esto ocurre ya que lo más natural es imaginar que, al haber muchas escuelas, el valor 1 tenga mas volumen.



[Figura 14] Tablero para la consulta del requerimiento 2 en cantidad de aulas

Se implementa en esta parte del laboratorio el reporte obligatorio que se debe crear con pentaho report designer, en este caso obtenemos los promedios de cantidad de aulas por departamento al subir al nivel zona país



[Figura 15] Reporte hecho en pentaho report designer del promedio de aulas en departamentos por zona país.

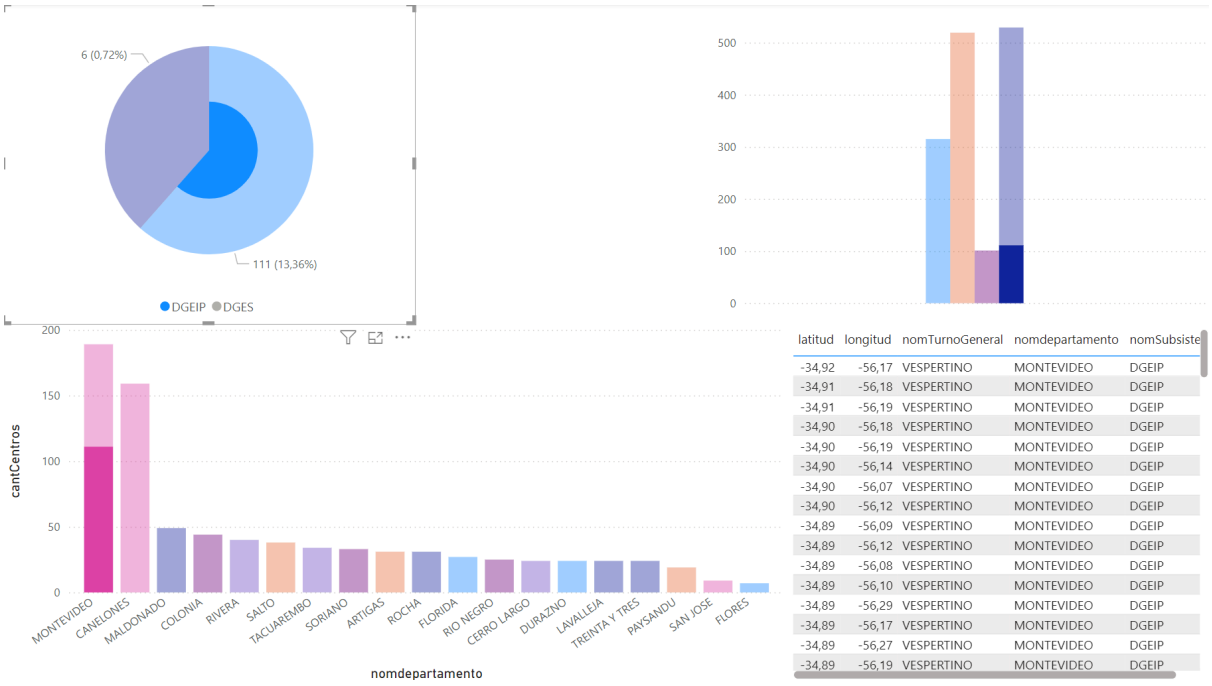
Se realiza de forma muy sencilla, destacando el poder de la herramienta para llevar a cabo reportes rápidos y fáciles de entender para el cliente. A continuación algunos detalles de la consulta utilizada:

```
SELECT nomZonaPais, avg(total_aulas) as promedio
FROM (
    SELECT nomZonaPais, nomDepartamento, sum(cant_aulas) as total_aulas
    FROM UBICACIONE
    JOIN (
      SELECT DISTINCT idUbicacion, idEscuela, idCategoria, idNivelDesde,
idNivelHasta, cant_aulas, gruposXnivel, cant_niveles
    FROM OFERTAS_INICIAL_PRIMARIA
    ) a ON UBICACIONE.IDUBICACION = a.idUbicacion
    GROUP BY nomZonaPais, nomDepartamento
) b
GROUP BY nomZonaPais;
```

En esta consulta, primero obtenemos las tuplas que podemos agrupar bien segun el analisis, es decir, la parte en negrita nos asegura que no tenemos problemas con el turno ya que si miramos en el análisis de aditividad presentado en el EVA del curso, este tiene aditividad en todo menos en turnos. Luego, se selecciona la cantidad de aulas para cada departamento y zona país y se saca el average en la consulta de más afuera.

5.3 Requerimiento 3

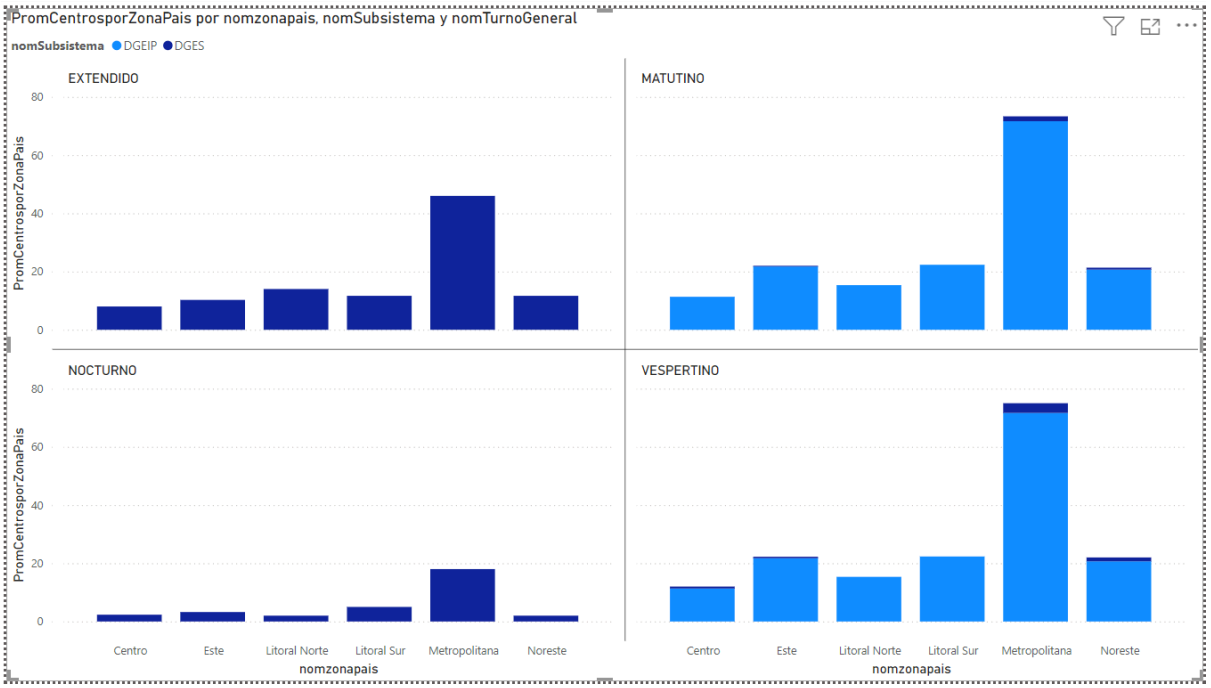
Se crean dos tableros en PowerBI para mostrar las medidas solicitadas, primero uno para la cantidad de centros y otro para la cantidad promedio de los mismos. El primer tablero, el cual se puede observar en la Figura 16, se puede cruzar por subsistema, turno y departamento, el resultado se despliega en la tabla de abajo a la derecha en dicha figura, la cual actualmente indica el resultado de consultar por Ubicación en Montevideo, subsistema DGEIP y turno vespertino (la leyenda de dicha dimensión se despliega al pasar el cursor por encima de cada valor).



[Figura 16] Tablero de cantidad de liceos siendo consultado por el departamento montevideo, subsistema DGEIP y Turno Vespertino.

Para la segunda medida, es decir, el promedio de departamentos, se implementó el caso específico al subir al nivel zona-país, se calcula por separado la suma de la cantidad de centros por departamento en cada zona-país y se divide por el número de departamentos que contiene la misma, esta consulta esta implementada con el código de la figura 18, donde `CantidadDepartamentosPorZonaPais` es simplemente la cantidad de departamentos que tiene la zona-país seleccionada. Como se puede ver en la Figura 17, el color representa el subsistema, las barras indican la cantidad promedio obtenida y están separadas por Zona-país, los cuatro cuadrantes indican el turno general seleccionado.

Se observa una diferencia notoria en la cantidad de centros en la zona metropolitana, esto se debe a que esta engloba a los dos departamentos con más centros (Montevideo y Canelones), por lo que tendrá un promedio mucho mayor al resto. Además se observa que el subsistema DGES (secundaria) es el único que tiene horarios nocturnos y extendidos.



[Figura 17] Tablero de la medida cant-prom-centros

```
PromCentrosporZonaPais = DIVIDE(SUMX(SUMMARIZE('Centros Educativos','Centros
Educativos'[idSubsistema],'Centros
Educativos'[idUbicacion]),1),[CantidadDepartamentosPorZonaPais])
```

[Figura 18] Consulta de promedio de centros por departamento al hacer roll-up al nivel Zona-País

6. Capacidad de soportar nuevas cargas

Cuando hablamos de soportar nuevas cargas, se entiende que estas cargas serán con un formato similar a las de los datos de entrada, para lo cual la solución presentada tiene una gran capacidad para soportar nuevas cargas debido a la implementación del proceso de carga en Kettle, en el que se intenta minimizar la complejidad a la hora de introducir nuevos datos al sistema.

Se identifican tres clases de nuevas cargas:

- **Nuevas tuplas que no cambian los requerimientos:**

  Con esto se hace alusión a las tuplas que no modifican en absoluto el esquema jerárquico, que no agregan ningún valor a una dimensión de los valores que se pueden obtener desde la misma tabla, por ejemplo, agregar un liceo en una localidad no existente no trae más problemas que solo ejecutar el kettle proporcionado, sin embargo agregar una nueva zona-país sí, porque este dato al estar definido por nosotros no se puede deducir de los datos y se debe intervenir como se detalla en el caso a continuación.

- **Nuevas tuplas que cambian los requerimientos pero no agregan nuevas jerarquías (solo agregan valores dentro de las mismas) ni dimensiones:**

  Con esta clasificación, se refiere a aquellos cambios que adicionan nuevos miembros de las jerarquías pero no nuevas jerarquías completas, por ejemplo, agregar un departamento es un cambio de este tipo. Para este caso no solo alcanza con usar el proceso de carga entregado de Kettle, sino que también es posible que se deba ajustar previamente los archivos JSON que describen los datos sobre las jerarquías. Estos archivos se encuentran en el directorio entrada/. Con la única excepción de los subsistemas, los cuales se indican de forma manual en salida\r3\subsistemas.json. En este caso, basta con añadirlo en el JSON para las jerarquías superiores, sin tener que llevar a cabo cambios en la estructura de la base de datos ni otros ajustes en el proceso de ETL. Por ejemplo, si se agrega un nuevo departamento, entonces debemos ir al archivo entrada\zonas-pais.json y definir a qué zona-país refiere el mismo.

- **Nuevas tuplas que sí cambian los requerimientos y agregan nuevas jerarquías, dimensiones o medidas:**

  Por supuesto que en este caso no solo se debe modificar el proceso de carga de Kettle, sino que también debemos ajustar la estructura de la base de datos y realizar los cambios pertinentes en el proceso de ETL y en el esquema mondrian para tener en cuenta las jerarquías o dimensiones añadidas.

7. Calidad de datos

En esta tarea se presentaron varios problemas de calidad de datos de diferente naturaleza. Se presentan a continuación en el siguiente formato:

- Problema
  - Solución encontrada (o propuesta)

- Ausencia de campos de latitud y Longitud de las planillas de datos
  - En las planillas de datos no se habían definido estos campos por lo que se extrajeron con este valor en el proyecto SIG-ANEP disponible en [8]

- A la hora de definir distintas relaciones con entidades con nombre (Liceos, Departamentos, etc), se tienen problemas de normalización de dichos nombres, es decir, están en formatos diferentes ya sea mayúsculas, minúsculas o abreviaturas
  - Para el caso de los departamentos, se optó por utilizar la convención de dejar los nombres literales sin acentos y en mayúsculas.

- Los datos proporcionados carecen de una forma de obtener localmente el código postal de las ubicaciones.
  - Se intenta solucionar mediante https://direcciones.ide.uy/swagger-ui.html#/Geocode/postReverseGeocodingUsingPOST pero no es posible debido a que este solo retornaba tres valores diferentes para cada consulta (probablemente se trate de un problema del código implementado por parte de este laboratorio).
  - Se propone utilizar la página de correo uruguayo que contiene las idZona de las localidades y permite descargar los datos en csv.
- Se observa que el csv proporcionado `distribucion-territorial-est-doc` que indica los porcentajes de docentes y estudiantes por departamento, para el caso de los docentes no suma 100, por lo que debe haber algún error en la fuente de los datos.

8. Conclusiones

8.1 Sobre el trabajo realizado

Como se puede observar, se optó por realizar la tarea de forma individual debido a que ya se había realizado un pequeño avance sobre el laboratorio en sus etapas iniciales, esto conllevo a que se debiera priorizar ciertas partes del laboratorio para dar lugar al aprendizaje de nuevos conocimientos. El requerimiento 3 se considera implementado en su totalidad, mientras que los requerimientos 1 y 2 carecen de un buen manejo de los problemas de aditividad debido a la falta de experiencia con el uso de mondrian, se utilizan como ejemplo igualmente las medidas con aditividad total en todas las dimensiones para implementar el front y adquirir los conocimientos necesarios sobre las herramientas.

8.2 Sobre el aprendizaje

El trabajo llevado a cabo permitió al estudiante adquirir conocimientos muy importantes sobre la manipulación de datos para su posterior uso en diferentes aplicaciones. Respecto a la **fase de ETL**, se aprendieron a utilizar tecnologías que permiten mejorar por varios órdenes de magnitud el tiempo que insume el manipular o realizar consultas sobre distintas fuentes de datos ya sea desde internet o local, utilizando normalización y carga en una base de datos. Respecto a la **parte de OLAP,** se consiguieron definir cubos tanto virtuales mediante mondrian y definición de un esquema, como mediante JSON y modelos de datos lógicos no relacionales. Por último, señalando la **parte de frontend**, se logró cumplir parcialmente con los requerimientos 1 y 2; y casi totalmente con el requerimiento 3, sin embargo, se hizo énfasis en intentar aprender todas las tecnologías y definiciones posibles para asimilar el funcionamiento de un sistema de bases de datos totalmente nuevo para el estudiante.

Se espera poder culminar el proceso de aprendizaje más allá de los límites del curso mediante la implementación de los temas que faltaron implementar y posterior testing mediante ejecución de consultas.

9. Bibliografía

1. Catálogo de datos abiertos del uruguay https://catalogodatos.gub.uy/
2. SINAE, Regiones. https://www.gub.uy/sistema-nacional-emergencias/node/726
3. Catalogo de datos abiertos, localidades del Uruguay https://catalogodatos.gub.uy/dataset/ide-localidades-del-uruguay
4. Códigos postales del correo uruguayo https://www.correo.com.uy/codigospostales
5. Consigna de la tarea de laboratorio granvoldat 2024 Grupo GEMA - InCo - FIng - UdelaR https://eva.fing.edu.uy/pluginfile.php/508778/mod_resource/content/2/letra-proyecto-2024_final.pdf
6. https://direcciones.ide.uy/swagger-ui.html
7. https://www.gub.uy/infraestructura-datos-espaciales/comunicacion/noticias/nuevo-sistema-unico-direcciones-geograficas-del-uruguay
8. https://geoportal-siganep.hub.arcgis.com/