**IIC3103 - Taller de Integración**
Departamento Ciencia de la Computación
Escuela de Ingeniería
Pontificia Universidad Católica

# Enunciado **Tarea 3**

## Objetivo

El objetivo de esta tarea es crear un programa capaz de integrarse con un sistema de bases de datos cloud y mediante la manipulación de los mismos, realizar un análisis pertinente.

## Trabajo a realizar

A grandes rasgos, deben generar un programa relativamente simple que consuma una base de datos instalada en la nube y que los almacene en una base de datos creada por ustedes, para luego hacer uso de dichos datos y poder tomar decisiones.

Más específicamente, tendrán que hacer un programa capaz de recopilar datos almacenados en un almacenamiento privado de Google Cloud Platform (GCP). Dichos datos están guardados en un *bucket* al cual deberán acceder mediante la información acá provista y el método que estimen conveniente. Luego, guardar la información recopilada en una base de datos creada por ustedes para hacer uso de la misma.

Hecho lo anterior, y mediante la utilización de herramientas para manipulación y visualización de datos, deben llevar a cabo un análisis de estos mismos, para finalmente acompañar con una recomendación que esté alineada y sea coherente con el análisis anteriormente realizado.

# Índice

# Contexto

La OMT (Organización mundial del turismo) se ha dado cuenta de que el turismo mundial se ha recuperado en cierta medida luego del coronavirus. Sin embargo, los países de Europa, que tienen una fuerte dependencia económica del turismo, quieren seguir potenciando su industria. Para esto le han pedido al organismo internacional que los ayude a cumplir su objetivo.

La OMT ve en esta consultoría una oportunidad sustentable. Dándole un foco medioambiental a la recomendación a realizar, puede lograr crear fuertes incentivos para que los países disminuyan sus emisiones de gases contaminantes y a la vez fomenten el reciclaje entre sus habitantes. Para lograr lo anterior, debe recomendar a los turistas países responsables con el ecosistema y amigables con el entorno.

Su trabajo será ayudar a la OMT a recomendarle a los potenciales turistas qué países serían una buena opción para visitar, tomando en cuenta el criterio por el cual busca guiarse la organización para lograr estimular a los países a desarrollar medidas ecológicas.

Para efectuar esto habrá datos disponibles para el análisis en un *Bucket* en S3, estos datos deberán ser guardados en una base de datos de una aplicación web que será finalmente consumida por un servicio de análisis de datos BI como Google Data Studio.

# Fuente de datos

Para esta entrega se ocupará únicamente la información entregada por el *Bucket*. Para conectarse a este se ocupan los siguientes datos:

## Conexión a la fuente de datos

- **Nombre bucket: 2022-01-tarea-3**
- Llave privada de cuenta de servicio, publicada en archivo por separado. Para utilizar este archivo desde el código, se recomienda revisar el siguiente enlace: Authenticating as a service account (ver sección “Pasa las credenciales mediante código”).
- Para la conexión con la fuente de datos, utilice alguna de las librerías oficiales de storage de GCP. La documentación la puede encontrar en el link: Cloud Storage client libraries (ver sección “Using the client library”).
- Para listar los objetos en un bucket, revise la documentación en List objects
- Para descargar y leer un objeto, revise la documentación en Download objects (ver sección “Descarga un objeto de un bucket”).

La conexión de datos debe realizarse cada 4 horas. Lo anterior es importante porque durante el desarrollo de la tarea y la evaluación de esta última, se irán agregando datos para testear que el script sea capaz de actualizar su base de datos con los datos actualizados del *bucket.*

Si despliega su código en Heroku, esto se puede realizar fácilmente mediante el *add-on* de Heroku Scheduler. Documentación en Heroku Scheduler.

# Descripción de herramientas relevantes

## *Bucket*

Un bucket es un contenedor de objetos. Como saben, un objeto puede corresponder a archivos de cualquier tipo. En esta tarea, los datos que deberá desplegar y analizar se encuentran almacenados en un *bucket* de GCP. En esta sección se encuentran los datos necesarios para ejecutar la conexión al *bucket*. Luego, para obtener acceso a dicha fuente de datos existen diversas herramientas que pueden utilizar y que probablemente conozcan[1].

## Google Data Studio

Queda a elección de cada uno la herramienta de visualización a utilizar. Sin embargo, recomendamos Google Data Studio, software construido por Google para generación de reportes y funciones de Business Intelligence (BI). La construcción de reportes se puede realizar mediante su interfaz gráfica web.

Data Studio ofrece, por defecto, múltiples conectores para emplear distintas fuentes de datos, incluyendo bases de datos SQL, NoSQL, productos de Google (Sheets, BigQuery, BigTable) y datos estáticos. Además, permite crear conectores personalizados para una mayor flexibilidad.

También entrega una serie de visualizaciones por defecto para los principales casos de uso de reportes (mapas, gráficos de línea, barra, torta, área, etc.), así como también visualizaciones de la comunidad, que son módulos de visualización externos generados por terceros.

Con esta herramienta podrán generar fácilmente un dashboard con los análisis que serán pedidos más adelante.

## Propuesta de arquitectura de la solución

A modo de referencia, se propone una arquitectura de solución a implementar, con los siguientes componentes:

- Capa de almacenamiento de datos: base de datos u otro que permita guardar los datos que se obtienen desde el bucket
- Capa de lógica:
    - Script que lea datos de fuente de datos de origen y guarde en capa de almacenamiento de datos.
    - Scheduler o reloj que ejecute el script con la frecuencia deseada.
- Capa de visualización: software de BI (como Google Data Studio) que sea capaz de leer la capa de almacenamiento de datos y generar las visualizaciones solicitadas.

Esta arquitectura se detalla a modo de propuesta, ya que existen múltiples otras arquitecturas que resuelven el problema solicitado.

---

[1] Pueden usar frameworks como Express, Koa, Django.

# Descripción detallada de los datos

Los datos presentados en el *Bucket* deberán ser mostrados en un dashboard, usando la herramienta de análisis de datos que estimen conveniente. El formato de los datos será de tipo *Data Lake*.

## Formato Datalake

Un *Data Lake* es un estándar de diseño que tiene como objetivo estandarizar el almacenamiento de datos, con el fin de permitir el acceso a estos datos de manera controlada. En otras palabras, la idea es tener un único repositorio para cualquier dato en una organización[2]. En este sentido, el formato de directorios dentro del Bucket será:

- **bucket/categoría/{año}/{mes}**

### *Bucket*

Se ha realizado un pequeño resumen de este concepto en el ítem anterior. Por otro lado, en esta sección se encuentran los datos necesarios para hacer la conexión con la fuente de datos.

### Categoría

Este atributo representa la categoría de datos que están siendo mostrados. A continuación se nombran y describen brevemente las categorías existentes:

- Aire: Estos datos representan las toneladas de polución de aire que un país ha producido por kilómetro cuadrado en una cantidad determinada de tiempo.
- Reciclaje: Estos datos representan el porcentaje de reciclaje domiciliario en un país.
- Turismo: Estos datos representan la cantidad de noches en las que personas extranjeras se quedaron en algún establecimiento turístico del país.

En cada una de estas categorías (que finalmente pueden ser vistas como directorios), existen más carpetas. A continuación se detalla una descripción del contenido al interior de las categorías.

### Año

Como se dijo en el ítem anterior, en cada categoría existirán carpetas con la información a utilizar correspondiente al año. La idea es que recopilen la información de cada una de estas carpetas

[2] Más información en https://aws.amazon.com/es/solutions/implementations/data-lake-solution/

para tener una visión aproximada de la evolución de las categorías en Europa a lo largo del tiempo.

Además, existe la posibilidad de que las carpetas de los años para una determinada categoría también contengan subcarpetas, que en este caso representen los meses. En dicho caso existirán doce carpetas enumeradas del uno al doce y donde cada una representa un mes (uno -> enero, dos -> febrero, ..., doce -> diciembre ), deben hacer el trabajo de profundizar en la información de dichas subcarpetas. En caso contrario, se encontrarán directamente con los objetos que contienen la información de la categoría explorada.

## Mes

En el caso de que los datos contengan frecuencia mensual, al interior de cada una de esas carpetas hallarán la base de datos que representa la categoría, año y mes seleccionado. Es parte de su trabajo ver qué categorías cumplen con dichas subdivisiones más profundas y cómo trabajar con ellas.

## Ejemplo

Para dar a entender de mejor manera la forma en que se diseña y estructura un bucket, y explicar de manera más visual cómo está construido el *bucket* que ustedes deberán consumir, se ha implementado el siguiente diagrama. En este caso en particular (hipotético), la categoría 'Turismo' está dividida en años, los cuales a su vez se subdividen en meses. Por otro lado, la categoría 'Aire' está dividida solo en años. Además, vale la pena señalar que no se incorporaron la cantidad de años, ni de meses que en realidad contiene la información alojada.


Figura 1: Ejemplo estructura *bucket* tarea

# Presentación de datos

No todos los datos serán presentados de la misma forma. De esta manera, objetos (u equivalentemente archivos) de una determinada categoría estarán con un formato específico para dicha categoría, y que será diferente a los de las otras. A continuación se detalla el formato de los datos para las diferentes categorías.

## Polución Aire:

- Formato
  - Para esta clase la información vendrá en formato de archivo ***.parquet***[3].

- Atributos
  - Esta categoría cuenta con dos atributos separados por coma:
    - Country: Nombre del país
    - Tonnes/square km: Toneladas de gases de efecto invernadero por kilómetro cuadrado

- Ejemplo
  - Por ejemplo, la siguiente tabla muestra las toneladas 'Y' de polución en el aire por kilómetro cuadrado para el país 'X' en el año 'Z'.

| | Country,Tonnes/ square km |
|---|---|
| 1 | País Uno, Porcentaje Uno |
| 2 | País Dos, Porcentaje Dos |
| 3 | País Dos, Porcentaje Dos |
| 4 | .... |
| 5 | .... |

Figura 2: Ejemplo presentación de datos para Polución de Aire el año X

## Tasa de reciclaje:

- Formato
  - Para esta clase la información vendrá en formato de archivo ***.json***.

- Atributos

[3] Para más información sobre este formato visitar el siguiente link: https://parquet.apache.org/

- Para esta categoría los datos serán almacenados en una lista, donde cada elemento contiene un diccionario con las llaves:
  - Country: Nombre del país
  - Recycling Rate: Tasa de reciclaje municipal del país.

- Ejemplo
  - A continuación un ejemplo de cómo podría verse un archivo en esta categoría.

```
[
        {
           "Country": "País Uno",
           "Recycling rate": 54.8
         },
        {
           "Country": "País Dos",
           "Recycling rate": 24.5
        },
        {
           "Country": "País Tres",
           "Recycling rate": 15.8
        },
          ...
]
```

## Noches pasadas en establecimientos de alojamiento turísticos:

- Formato
  - Para esta clase la información vendrá en formato ***.csv***.

- Atributos
  - Cada instancia de esta categoría contiene dos atributos separados por punto coma:
    - Country**:** Nombre del país
    - Noches pasadas en residencias turísticas: Número que representa la cantidad de noches pasadas en alojamientos de residencias turísticas de dicho país

- Ejemplo
  - A continuación un ejemplo de cómo podría verse un archivo en esta categoría. En este caso, el número de alojamientos en residencias turísticas en el país X y para el año Y.

| | A |
|---|---|
| 1 | País Uno; 118.553 |
| 2 | País Dos; 618.553 |
| 3 | País Tres; 1.018.553 |
| 4 | ... |
| 5 | ... |
| 6 | ... |

Figura 3: Ejemplo presentación de datos para Turismo el año X

## Presentación de Resultados

Una vez hayan recolectado los datos necesarios para realizar el reporte, cada alumno deberá generar un dashboard que muestre los datos, permita la manipulación de los mismos, la obtención de métricas y posibilite apoyar la toma de decisión basándose en los datos reportados.

El reporte deberá contener, a lo menos, un control de período[4], que permita seleccionar los datos a mostrar en el resto del reporte, y un filtro por país[5], que permita filtrar por uno o varios países. Por defecto, se deberán mostrar todos los países como activos. Para complementar su reporte y dependiendo la herramienta escogida, también pueden utilizar las visualizaciones de mapa[6].

Datos y resultados que deben incorporarse en el reporte:

- Tabla con todos los países que están presentes en la base de datos.
- Tabla que permita visualizar el ranking de los 10 países que lograron la menor contaminación del aire entre los años 2015 y 2019 (basándose en el promedio).
- Tabla que permita visualizar el ranking de los 10 países con la mayor cantidad de noches hospedadas en acomodaciones turísticas entre los años 2015 y 2019 (basándose en el promedio).
- Tabla que permita visualizar el ranking de los 10 países que más reciclaron entre los años 2015 y 2019 (nuevamente basándose en el promedio).
- Asignándole un ponderador que indique la importancia que le otorga a cada una de las categorías, cree una tabla que considere los 5 países que tengan menor contaminación y mayor número de noches en alojamientos turísticos anualmente. Justifica el valor que le diste a cada categoría para generar la escala.

[4] Documentación del control de periodo en Data Studio: Control de periodos
[5] Documentación filtros Data Studio
[6] Documentación visualización de mapas para Data Studio aquí

Un ejemplo sería:

- **num_noches_pais** = Número de noches que los turistas se alojan en acomodación turística por país
- **polucion_pais** = polución del aire del país por kilometro cuadrado
- **Puntaje_pais** = 0.5 * **num_noches_pais** + 0.5 * **polucion_pais**

Así, de acuerdo al puntaje obtenido por el país, se debe construir un ranking ordenando de mayor a menor. Se debe mostrar el top 5 de países con menor polución y mayor turismo de acuerdo a tu criterio de importancia para cada categoría.

- Un histograma que permita visualizar la tasa de reciclaje desde el año 2016 al año 2019 de Alemania.
- Gráfico de líneas que muestre la cantidad de noches pasadas en alojamiento turístico respecto a los meses en el año 2020, consideren la suma de todos los países.
- Un gráfico de torta dividido en intervalos del porcentaje de reciclaje entre los años 2010 y 2019, donde el tamaño de cada pedazo de la torta viene dado por el número de países que reciclan un porcentaje perteneciente al rango que representa el pedazo.

Finalmente, de acuerdo a todos los datos que han extraído, deberán hacer un análisis con la finalidad de poder contestarle a la OMT la siguiente pregunta:

**¿Qué países deberíamos recomendar a los turistas para que vayan este 2022? ¿Por qué?**

**¡Se espera que la respuesta esté basada en un análisis crítico y justificado, y a su vez sea consistente con los resultados de su dashboard!**

## Consideraciones

Los datos usados para esta tarea son reales. Por lo anterior, encontrarán en algunos archivos datos con ":"; esto significa que no se registraron datos de esa categoría, para ese país y momento, y por ende deberían lograr manejar estas situaciones, evitando guardar este tipo de datos.

## Versionamiento de código

El sitio y todo su código fuente deberá estar versionado en un repositorio en GitHub, creado en el classroom del curso:

- Link para creación de repositorio: https://classroom.github.com/a/VVWxS-98

## Entregables

Cada alumno deberá entregar, mediante un formulario publicado en el sitio del curso, las siguientes *url's*:

- URL de acceso al dashboard[7].
- URL del repositorio a GitHub.
- URL del script web que consumirá el bucket.

La evaluación de la tarea se realizará bajo una rúbrica a publicar. Se evaluará la completitud del dashboard generado y la correctitud de los resultados presentados.

## Fecha de entrega

La tarea deberá ser entregada a más tardar el día 13 de mayo antes de las 18:00, y deberá ser capaz de seguir recolectando datos hasta 24 horas después para la validación de resultados obtenidos.

## Requisitos mínimos

Las tareas que no cumplan con las siguientes condiciones no serán corregidas y serán evaluados con la nota mínima:

- El código deberá estar versionado en su totalidad en un repositorio de GitHub
- El script debe reflejar fielmente el código entregado en el repositorio. Para la revisión, más de una tarea serán corridas localmente por los ayudantes para comprobar el cumplimiento de este punto.

## Penalizaciones

Se descontarán 0,2 puntos de la nota de la tarea por cada hora de atraso en la entrega, contados a partir de la fecha estipulada en el punto anterior.

Cualquier intento de copia, plagio o acto deshonesto en el desarrollo de la tarea, será penalizado con nota 1,1 de acuerdo a la política de integridad académica del DCC.

---

[7] El link de visualización con permisos de edición.