**Procesamiento de datos del api de SpaceflightNews:**

El presente documento describe el proceso implementado para la extracción, procesamiento almacenamiento, modelado de Datawarehouse de los datos obtenidos del api de SpaceflightNews. Todo este proceso esta orquestad mediante Apache Airflow y culmina con el poblado de las dimensiones y tabla de hechos utilizando la tecnología de unity catalog en Databricks para contar con un proceso de gobernanza de datos.

**Arquitectura del proceso ETL implementado:**

El proceso implementado es orquestado a través de un DAG en pache Airflow llamado INETUM\_ETL el cual sigue la siguiente secuencia de tareas:

start >> extract\_data >> preprocess\_data\_task >> load\_data\_task >> clear\_data\_task >> run\_job\_spark >> end.

Cada tarea contiene un propósito clave los cuales son:

* Extract\_data: este proceso se encarga de la extracción de los datos mediante la clase ExtractAPI el cual se encarga de hacer la petición al api de SpaceflightNews.
* preprocess\_data\_task: Este proceso se encarga de a eliminación de duplicados de los artículos extraidos
* load\_data\_task: Este proceso se encarga de de guardar los documentos extraídos en formato json para posteriormente ser almacenados en un datalake.
* clear\_data\_task: Este proceso se encarga de limpiar los archivos temporales utilizados en apache airflow
* run\_job\_spark: Este proceso se encarga de invocar el Job construido en Databricks mediante el operador DatabricksRunNowOperator, el cual se encarga de procesar, modelar y poblar las dimensiones y tabla de hechos creada.

**Proceso Databricks:**

El Job construido en Databricks se encarga de invocar un notebook el cual es el encargado de:

* Lectura de los datos en formato json almacenados en el datalake
* Aplanado de los datos, en el cual se aplanan los datos en formato array para la extracción del nombre del autor.
* Procesamiento nlp mediante la implementación de la librería de spacy el cual se encarga de la clasificación de los artículos, extracción de palabras claves y extracción de entidades.
* Generación de un dataframe limpio el cual cuenta con una estructura optima y datos relevantes para el proceso de modelado del Datawarehouse
* Modelamiento de los datos para la creación del Datawarehouse
* Poblado de las dimensiones y tabla de hechos teniendo como base el dataframe final que cuenta con una data limpia y estructurada.

**Tecnologías implementadas:**

* Lenguaje de programación Python
* Apache Airflow: orquestador de flujos de trabajo
* Apache Spark en Databricks: Sistema de procesamiento de datos
* Datalake: Capa de almacenamiento de data estructurada, semiestructurada y no estructurada
* Unity Catalog en Databricks: Capa de gobernanza de datos y modelado para el Datawarehouse
* Spacy: Librería utilizada para el procesamiento de lenguaje natural(NLP)