

Diseño del DAaaS

Definición la estrategia del DAaaS

- **Se va a crear un servicio de comparación de precios entre varios supermercados, Este te va a permitir seleccionar productos para simular tu compra, mediante distribución de productos por categorías, o un buscador. A medida que añadas los productos, la web comprobará el precio en todos los supermercados, e irá completando tu carrito de compra. A medida que se van añadiendo los productos, te muestre el coste total de tu cesta en las diversas combinaciones de los supermercados. Gracias a eso, conseguiremos tener el conocimiento de en qué super sale más económica mi compra.**

Arquitectura DAaaS

- 3 API de supermercados, necesitaremos una para cada supermercado a analizar, "Mercadona, Carrefour España y DIA" tienen API disponible.
- Mediante una Cloud Function "CF1", descargaremos la información de cada supermercado y así actualizar precios, crear o eliminar productos.
- Un Task Scheduler que lance la "CF1" a actualizar los datos de las API.
- Un Bucket de Google Storage donde almacenaremos la información que nos llega desde la API.
Una Cloud Function "CF2" que enviará la información alojada en el bucket a Hadoop cuando este tenga cambios en sus archivos.
- Una máquina virtual que contendrá
 - Un Hadoop que almacenará en el HDFS la información siguiente:
 - La imagen de producto
 - El nombre de producto.
 - El código EAN.
 - Precio total.
 - Precio por KG.
 - Categoría de producto.
 - Supermercado que vende producto.
 - Un Cloud Function "CF3" que lanza Elastic Search.
 - Un Elastic Search que filtra y procesa.
 - Un Scoop que convertirá la bbdd a SQL y separará las fotos en archivos y la bdd a SQL.
- Una Cloud Function "CF4" que con un Script de Python, envía las fotos del Hadoop al Bucket, y obtiene los links de cada una, y las inserta en la bbdd. Para enviarlas al SQL Cloud.

- Un SQL Cloud, donde se aloja todo el "Data" procesado.
- Una Cloud Function "CF5" que cargue todos los datos al Backend a petición.
- Un Backend para conectar la bbdd con la web cliente.

DAaaS Operating Model Design and Rollout

- Iniciamos la el funcionamiento comandado por el "Scheduler" cada 24h.
- Se llama a una cloud function "CF1" que Solicita la información de las API de los supermercados.
- Esta carga toda la información en un bucket de Google Sotrage, donde tendremos la información en "sucio". Y también, por otro lado las fotos de los productos.
- A continuación, tenemos una cloud function "CF2" que ordena la carga de nuevos datos al HDFS de Hadoop alojado en una Máquina Virtual. Aquí cargamos la información en "formato BBDD" para poder trabajarla.
- Para trabajarla y procesarla, hay una Cloud Function que envía la información que tenemos a un Elastic Search, para así preparar, filtrar y organizar la información en base a nuestros gustos.
 - Filtramos las categorías, con un nombre común para los 2 supermercados, aprovechando la tecnología de detección de duplicados, nombres parecidos...
 - Filtramos los productos revisando código EAN, y revisamos nombres unificándolos. Y así, evitamos duplicar productos.
- A continuación, devolvemos la información a HDFS procesada y envíamos la bbdd Scoop sin lás imágenes, pero poniendo nombre a las imágenes para relacionarlas con la PK de los productos.
- Ahora, una Cloud Function llama a HDFS para que envíe las imágenesal bucket y devuelva los enlaces para añadirlos a la BBDD. Tras eso, envía la BBDD al SQL Cloud.
- A partir de esto, disponemos de otra Cloud Function que está lista para enviar los datos de BBDD y imágenes al Website Backend cuando este lo solicite.

Diagrama:

