



SISTEMA DE INFORMACIÓN POKÉMON

ID del Proyecto	20240613_AP	Nombre del Proyecto	SISTEMA DE INFORMACIÓN POKEMON
-----------------	-------------	---------------------	--------------------------------

NOTIFICACIÓN DE PUBLICACIÓN DEL DOCUMENTO

Detalles del Documento			
Versión	1.0	<input type="checkbox"/> Borrador	<input checked="" type="checkbox"/> Final
Jefe de Proyecto	José Arturo Gómez Díaz		
Nombre del Documento	Diseño_Técnico_POKEMON-v1.0		

Detalles de la Revisión:

Num. Rev.	Fecha Rev.	Autor	Descripción
[1.0]	6/28/2024	Avanade / POKEMON	Versión inicial

Equipo de análisis:

(Lista de personas que colaboraron en la elaboración de este documento)

Nombre	Contacto	Area/Equipo	Rol
Jose Gómez	j.g@avanade.com	Intelligent Data Platform	Project Manager
Arturo Diaz	a.d@avanade.com	Intelligent Data Platform	Data Engineer



Contents

1.	Introducción.....	3
	Propósito del documento:	3
	Alcance del proyecto:	3
	Dependencias:.....	3
2.	Diseño de la Solución	3
	Diagrama de flujo de datos:	3
	Descripción de tecnologías:.....	4
3.	Proceso ETL.....	5
	Descripción de los objetos y procesos:	5
4.	Distribución de capas	6
	Bronze:	6
	Silver:.....	6
	Gold:.....	6
5.	Monitorizacion	6

1. Introducción

Propósito del documento: En este documento se incluye la descripción de las tecnologías utilizadas en el proyecto definiendo la funcionalidad de orquestación, ingesta y ETL de datos así como posterior análisis BI para el proyecto de Sistema de Información Pokémon.

Alcance del proyecto:

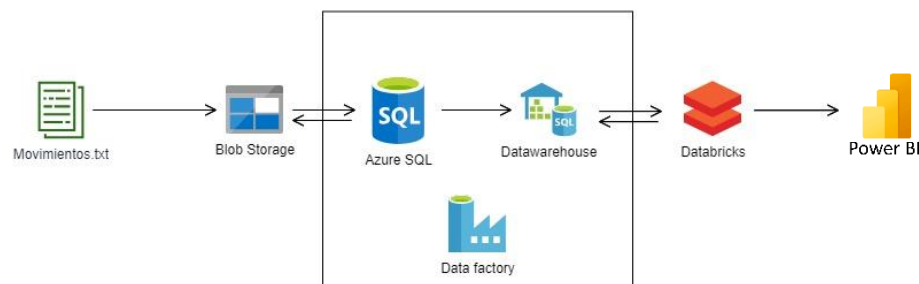
1. Creación de base de datos SQL Server
2. Creación e inserción de tablas y registros
3. Orquestación de ingesta de datos dinámicos en tabla desde archivos .txt
4. Tablas de hechos y dimensiones necesarias para generar esquema estrella y obtener Cubo OLAP
5. Carga de modelo en PowerBI
6. Generación de informes

Dependencias:

1. Data Factory: Orquestación capa Bronce y Silver
2. Cluster DataBricks: Ejecución de ETL en clusters dedicados en notebooks con lenguaje Python. Creación capa Gold a partir de Silver

2. Diseño de la Solución

Diagrama de flujo de datos:





DOCUMENTACIÓN TÉCNICA: SISTEMA DE INFORMACIÓN POKÉMON

Descripción de tecnologías:

1. Azure Storage account: Blob storage para realizar la carga de archivos de manera manual. Contenedor usado “*containerfactorypokemon*”. Las modificaciones requeridas que modificas las tablas de Movimientos se guardan los resultados en otro contenedor “*datasinkflow*”

storagefactorypokemon

1. containerfactorypokemon
 2. datasinkflow
2. Azure SQL Database: Almacenar tablas proporcionadas, receptor de ingesta de datos desde .txt y almacenar modelo OLAP. Para interactuar con la base de datos se ha usado Query Editor (Azure Portal), SSMS (SQL Server Managment Studio) y desde Notebooks en Databricks

- pokemonDB

- datawarehousePOKEMON

3. Data Factory: Orquestación de la base de datos, introducir datos a SQL Database
4. Azure Databricks: Notebooks en lenguaje python para realizar análisis exploratorios y transformaciones de datos. Construcción del modelo estrella para facilitar la posterior sincronización con PowerBI Desktop
5. PowerBI: Conexión con el modelo guardado en la base de datos, medidas sobre tablas de hechos y dimensiones y creación de reportes.

1. PowerBI Desktop
2. PowerBI Service



3. Proceso ETL

Generación de modelo de explotación se ha desarrollado una primera parte en Azure Data Factory ejecutado a modo de trigger manual y corresponde a la ingesta de los ficheros Movimientos.txt, transformaciones de columna y tipos y posterior copia a SQL Database

Posteriormente con Azure Databricks se realizan operaciones de análisis y transformación para combinar las tablas originales y conseguir construir el cubo OLAP que permita posteriormente analizar la información detalla.

Originalmente se dispone de siete tablas en el esquema de datos relacional proporcionado y se realiza las transformaciones para conseguir un esquema estrella entre la tabla hechos *HPokemon2* y las dimensiones *Dpokemon*, *Movimientos* y *Entrenadores*

Descripción de los objetos y procesos:

1. Pipelines Data Factory:

1. Orquestación

Copy Data de .txt a SQL

Dataflow creación de tipo y concatenación

Copy data para salvar la transformacion en la base de datos SQL

2. BD1_BD2

1. Bucle ForEach para realizar el copy data a la nueva base de datos SQL que permitirá realizar las transformaciones necesarias para la visualización de datos

2. Notebooks:

EDA: Creado para realizar un análisis exploratorio, comprobar valores vacíos, tipo de datos, etc.

Querys: Consultas básicas incluidas en los requerimientos del proyecto

DATAFLOW: ETLs para crear el esquema estrella final en la capa Gold



4. Distribución de capas

Los procesos ETL del modelo afectan a 2 de las 3 capas de la arquitectura

Bronze: Capa del primer nivel de información, destino de los ficheros subidos a los contenedores.

Silver: Contiene los datos originales proporcionados que son almacenados en el SQL así como los datos ingestados de archivos .csv o .txt que se suben a los contenedores. Se almacenan en el SQL Datawarehouse y sirven como estado final de las tablas para permitir utilizarlas en otras aplicaciones.

Gold: Esta capa contiene el subconjunto de información del modelo analítico creado sobre la capa Silver. Se ha reducido el numero de tablas respecto al original para conseguir generar un modelo lo más próximo al modelo estrella que formara el datawarehouse que se sincroniza con PowerBI y optimizar las consultas necesarias en las visualizaciones.

5. Monitorización

Los pipelines de Data Factory podrán ser supervisado usando la propia interfaz Monitor de Data Factory pudiendo consultar registro de ejecución de los pipelines, su resultado, y en caso de error la actividad que provoca el conflicto.

Se puede habilitar notificaciones por correo para conocer ejecuciones no fallidas de las pipeline según las métricas seleccionadas.