**PRUEBA TÉCNICA APRENDIZ U**

Gracias por participar en el proceso de selección para Bancolombia, como parte del proceso queremos conocer el nivel de tus conocimientos técnicos y análisis de casos, es importante que nos muestres el nivel real de tus conocimientos, si no sabes la respuesta siéntete con la tranquilidad de responder que no conoces.

**Nombre: Maria Camila Zapata Arrubla**

**Numero de documento: 1000921385**

1. **Pregunta De Conocimiento Técnico**

Explique, con sus palabras, cómo resolvería técnicamente el siguiente problema presentado en la Gerencia de Servicios de Monitoreo a los Canales:

La plataforma web Genesys Cloud se utiliza para la gestión y el seguimiento del ANS en las llamadas realizadas por los clientes a la línea telefónica. En esta plataforma existen aproximadamente 40 servicios, y para cada uno de ellos es necesario ingresar manualmente, uno por uno, para generar un archivo exportado en Excel que contiene el detalle de cada llamada y el tipo de necesidad del cliente.

Actualmente, la gerencia cuenta con 3 licencias para acceder a Genesys Cloud y descargar manualmente estos archivos. El desafío consiste en automatizar este proceso, de manera que se ejecute cada hora y se complete en un máximo de 10 minutos para los 40 servicios.

¿Qué alternativas propondría para automatizar este proceso de la manera más eficiente posible?

**Respuesta:** Para dar solución a este problema pensaría en primero consultar si la Genesys Cloud dispone de APIs para poder hacer uso de ella (<https://www.genesys.com/developer>), de estar esta opción disponible plantearía lo siguiente:

* **Automatización mediante la API:** Como tenemos 3 licencias para acceder a Genesys Cloud, al hacer uso de la API se puede crear un script, por ejemplo en Python, el cual ejecute o incluya los siguientes pasos:
  + Autenticación con las credenciales.
  + Generar un bucle en el cual ingrese a cada uno de los 40 servicios existentes.
    - Dentro del bucle, cuando esté ingresando a uno de los servicios se debe verificar que este esté disponible y que muestre el error. En caso tal de no poder acceder debería de almacenarlo en un espacio de memoria aparte para posteriormente volver a intentarlo con esos servicios. Cuando sea un error que indique que simplemente no existe o algo por el estilo, entonces sería eliminarlo o actualizar la referencia del servicio, para continuar con el siguiente servicio.
    - De estar todo correcto, empezaría con el proceso de generar un archivo de Excel con la información detallada de cada llamada e identificar el tipo de necesidad del cliente.
      * Para esto sería necesario verificar si la API incluye la posibilidad de exportar esta información en archivo de Excel. De no ser así, entonces se extrae la información necesaria y se almacena en un archivo .json y después con ayuda de pandas convertirlo en un archivo .xlsx.
      * Por cada proceso es bueno mostrar mensajes que permitan saber si sí está funcionando correctamente. Por ejemplo: “Extrayendo información de la llamada {name}”. “Se extrajo correctamente la información de la llamada {name} para el servicio {name\_servicio}”.
  + Con respecto a ejecutar el script a cada hora, se podría usar servidores en la nube que permitan la programación de estas actividades. También se podría ensayar teniendo en cuenta el sistema operativo, decirle que cada cierto tiempo ejecute algún archivo .py u otro que contenga todo lo necesario para la realización de este proceso, por ejemplo se podría crear un paquete el cual contenga todas las funciones, todos los archivos organizados y que sea solo llamar dicho paquete para que ejecute todo.
  + Con respecto al tema de ejecutarlo en 10 minutos, es una tarea que requiere de varias pruebas iniciales, entender el tema de los llamados cuánto tiempo puede tomar esto, cuántos llamados a la vez permite la API, si es posible usar paralelismo para ejecutar varias tareas a la vez, también revisar cómo se puede mejorar el código existente, identificar llamados innecesarios o maneras óptimas de hacerlo.
* **Creación de agente:** Para el proceso de automatización se podría hacer uso de la plataforma n8n para automatizar el proceso anterior o de alguna herramienta que pueda suplir esta necesidad.

1. **Ejercicio Práctico Python**

¿Cómo diseñarías una ETL para habilitar el monitoreo en tiempo real en un entorno que procesa más de 100 millones de transacciones diarias, asegurando una latencia máxima de 1 minuto e integrando modelos analíticos desarrollados internamente? Además, ¿qué modelos analíticos implementarías para garantizar tanto la escalabilidad como la tolerancia a fallos?

**Respuesta:** En el proyecto he creado un README donde se explica cómo ejecutarlo y qué contiene el proyecto.

El proyecto no trata la parte de manejar datos en tiempo real ya que esta es una parte que no domino muy bien, pero me encantaría y tengo la disposición de aprender sobre esta área.

En el proyecto hay un archivo que se llama ETL.py donde con ayuda de Inteligencia Artificial estuve explorando un poquito esta área de manejo de librerías que son útiles para tratar grandes volúmenes de datos, este archivo es solo una simulación y un intento de entender mejor este campo.

Con base en tu respuesta anterior, desarrolla la ETL propuesta utilizando la base de datos proporcionada: data.csv

**Metadata:**

0 channel : Canal por el cual se realiza la transacción

1 devicenameid : dispositivo por el cual se realiza la transacción

2 finaltrxyear : año de la transacción

3 finaltrxmonth : mes de la transacción

4 finaltrxday :dia de de la transacción

5 finaltrxhour : hora de la transacción formato numérico (8)

6 transactioncode :codigo que identifica una transacción

7 transactioncodedesc :nombre de la transacción

8 responsecode :respuesta de la transacción cuando en cero en exitosa, diferente de cero no exitosa

9 responsecodedesc : nombre de respuesta de la transacción

10 transactiontype : si es una transacción monetaria o no

11 transactionvouchernumber comprobante físico de una transacción

1. **Ejercicio Práctico PowerBI**

Con base en los resultados de la ETL del punto anterior, debes desarrollar un tablero de monitoreo funcional que permita visualizar, analizar e interpretar los principales indicadores clave de desempeño (KPIs) necesarios para:

* **Identificación**: Detectar oportunamente las alertas relacionadas con anomalías, fallos o eventos críticos en los procesos monitoreados.
* **Análisis**: Profundizar en las causas de las alertas mediante visualizaciones detalladas que permitan comprender las tendencias y patrones de los datos.
* **Comunicación**: Presentar de manera clara y accesible la información relevante para los diferentes niveles de usuarios involucrados, desde operadores hasta tomadores de decisiones.

**Criterios de evaluación:**

* Automatización de la tarea.
* Uso eficiente de librerías para manipular datos.
* Gestión adecuada de excepciones y estructura del código.
* Generación de resultados con la estructura especificada.

**Nota:** No me fue posible implementar el dashboard para este proyecto debido al tiempo. Sin embargo, me gustaría compartir con ustedes un dashboard que desarrollé previamente en otra actividad, para que puedan tener una referencia de mi trabajo en ese tipo de entregables.

**Entregables de los puntos 2 y 3:**

**Vas a entregar este proyecto a dos equipos, uno funcional y uno técnico**

**Para el equipo técnico**

Preparar para presentación:

* Documento con las respuestas a las preguntas y el análisis correspondiente del caso.
* Archivo .py con el desarrollo de las ETL.
* Tablero de visualización de Power BI, así como los archivos de conexión de este.

**Para el Equipo de operación de monitoreo**

Para la entrega de la prueba técnica, debes desarrollar un tablero de monitoreo funcional que sea intuitivo y fácil de entender para el equipo de la operación de monitoreo, quienes no tienen conocimientos técnicos. El tablero debe incluir indicadores y alertas visuales claras que permitan identificar rápidamente problemas críticos. Además, debe ser interactivo, con filtros que faciliten la exploración de datos.

Adicionalmente, necesitas preparar un pitch breve para presentar el tablero al equipo. Este debe enfocarse en los beneficios clave del tablero, explicando de manera sencilla qué mide cada indicador, cómo las alertas generan acciones rápidas y cuál es el impacto positivo en la operación diaria. El objetivo es destacar cómo esta herramienta ayuda a optimizar procesos y a mejorar la experiencia del cliente.