



## 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: **Miller Alexis Quintero García**

Numero de documento: **1000214103**

### 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:** Inicialmente alternativas basadas en scripts de código y conexiones a la API de Genesys Cloud suenan atractivas, y efectivamente con aplicables, sin embargo, bajo una premisa de fácil mantenimiento, uso y aprovechamiento de las 3 licencias disponibles, propongo una solución basada en automatización robótica de procesos (RPA).

Para automatizar esta tarea, específicamente se hace uso de **Automation Anywhere**, que es parte de la suite tecnológica de [Bancolombia](#). Se justifica esta elección, ya que permite simular acciones humanas (navegación, descarga de archivos, interacción con interfaces gráficas), además cuenta con una interfaz de usuario, lo cual hace que el proceso sea mantenible por personal de la compañía sin tanto conocimiento técnico, y de una forma más intuitiva y gráfica.

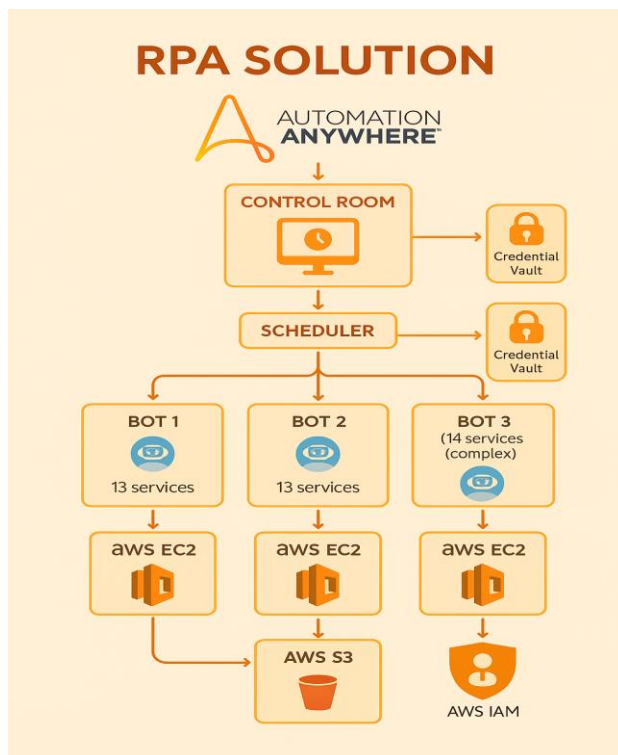
Para la solución se diseñan **3 bots runners para que trabajen en paralelo** (cada uno empleando una licencia de Genesys Cloud), de manera que a 2 bots se les asignan 13 servicios diferentes, y al otro bot 14 servicios. Luego, para garantizar que la eficiencia del proceso logre cumplir el estándar de máximo 10 minutos y también agregar una alta disponibilidad, se asigna a cada bot los recursos necesarios en máquina virtual de cómputo EC2 de AWS ([principal cloud provider de Bancolombia](#)).

Estos bots deberán estar registrados como dispositivos en la **Control Room** de Automation Anywhere, desde donde se gestionarán centralizadamente. Junto a esto, se configura un *scheduler* que ejecute los tres bots cada hora, asegurando la automatización sin intervención manual y en la frecuencia de acciones deseada. Adicionalmente, para agregar una capa de seguridad a la solución, **las credenciales de acceso a Genesys Cloud se guardan encriptadas** en 3 lockers diferentes usando la función de *Credential Vault*, de manera que cada bot solicita acceso únicamente a su respectivo locker, y el usuario común desde la interfaz de Automation Anywhere no podrá ver las credenciales descriptadas.

En cuanto a almacenamiento de los archivos, suponiendo que los archivos deben estar disponibles tanto para el personal del área, como para otros sectores que requieran los datos para hacer analítica y big data; la mejor opción es guardar los archivos en un servicio de almacenamiento como un **bucket S3 de AWS**, de

manera que, se crea previamente un rol con los permisos necesarios para que los bots corriendo en EC2, puedan subir los archivos. Cabe destacar que los bots antes de subir al archivo, pasarán por un script que gestione los nombres, para evitar sobreescrituras, esto implica una nomenclatura de archivos relacionada con la fecha y hora de descarga, así como el numeral del servicio.

Toda esta solución la sustento en servicios de la nube, con el fin de que haya alta disponibilidad y tolerancia a fallos, sin embargo, si en un estudio de precios se demuestra que sale más económico usar recursos locales o por políticas de Bancolombia el interés es tener una architecture on-premise, simplemente los bots correrían en máquina local y los archivos se almacenarían en la unidad de red compartida, accesible fácilmente para el equipo de Gerencia de Servicios de Monitoreo.



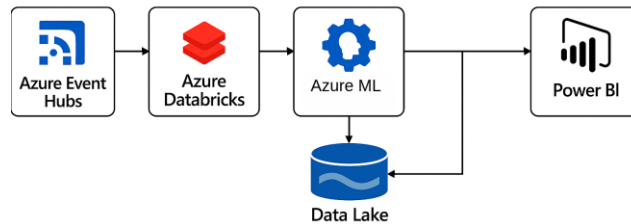
## 2. Ejercicio Práctico Python

¿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:** Para el diseño de esta ETL es importante considerar que la cantidad de datos hace perfectamente que el proyecto entre en el área de Big Data, por lo cual considerando todo el ciclo desde la ingesta de datos hasta la analítica y visualización en Power BI, la mejor opción es plantear una arquitectura con Microsoft Azure, para garantizar algo automático con Power BI incluido ya que este producto también es de Microsoft. (Azure es el segundo cloud provider de Bancolombia).

La arquitectura comienza con [Azure Event Hubs](#) ya que este servicio permite trabajar en tiempo real garantizando escalabilidad para picos de carga de datos naturales en ciertos momentos del día y fechas para Bancolombia, además de también captar los datos de fuentes diversas, ya que los canales de recepción de datos son muy diversos. Luego, con Databricks se pueden aplicar las limpiezas y transformaciones teniendo experticia en como llegan los datos, todo esto en un proceso conectado con Event Hubs.

Para los modelos analíticos, aunque aquí no conozco a detalle cómo se conecta Databricks con Azure ML, quizás con un data lake intermediario se pueden tener los datos almacenados en nube, para que el servicio de machine learning de Azure entrene de forma automática los modelos, que por interés de la Gerencia de Monitoreo de Canales, deben ser centrados en clasificar eventos y detectar anomalías. Finalmente, y como punto importante, desde el propio data lake también se puede hacer la conexión a Power BI, para manejar un tablero de control que, por medio de toda esta conexión de nube, se puede actualizar de forma constante y mantener alertas de tiempo real para todo el equipo.



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 : día de de la transacción
- 5 finaltrxhour : hora de la transacción formato numérico (8)
- 6 transactioncode : código 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

**NOTA:** En el zip se encuentran los .py de la ETL para el .csv, aunque el respaldo de muchos de esos procesos directos está en este Jupyter Notebook de Collab donde se estudió de forma detallada y paso a paso todos los datos: <https://drive.google.com/file/d/1-2XFMenhNg1lKEYPe4VQzsbzPf8XaYYJ/view?usp=sharing>

### 3. 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.

**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.

Enlace al vídeo pitch:

<https://drive.google.com/file/d/1RWMT3VtPtQ5SYVldwUDHuESvpMXrUOJB/view?usp=sharing>

Repositorio GitHub:

[https://github.com/MillerQuintero2001/Reto\\_Bancolombia](https://github.com/MillerQuintero2001/Reto_Bancolombia)

**NOTA:** El archivo CSV para el tablero Power BI y también el archivo del tablero, se entregan en el zip adjunto al correo. Cualquier inconveniente, para visualizar, quedo atento al correo.

¡Muchas gracias!