# **Evaluación para Candidatos a Ingeniero de plataforma y Dataops de Banco Base**

**Cuestionario**

El siguiente cuestionario tiene como objetivo explorar tu experiencia y habilidades como ingeniero de plataforma de datos. Las preguntas son abiertas y buscan una reflexión profunda más allá de un simple "sí" o "no". No hay respuestas correctas o incorrectas absolutas; lo que nos interesa es entender tu proceso de pensamiento y enfoque para resolver los desafíos planteados. Te animamos a detallar tus respuestas y a explicar tu razonamiento. Si encuentras alguna pregunta sobre un tema que no dominas, no te preocupes, puedes dejarla en blanco. Valoramos tu honestidad y disposición para compartir lo que sabes y lo que estás aprendiendo.

1. ¿Cuál es tu metodología para abordar un problema complejo relacionado con datos?

A continuación, enlisto los puntos:

* Lo primero es tener la mayor cantidad de detalles del requerimiento, entender lo que ese esta pidiendo y finalidad del requerimiento, para mitigar retrabajos.
* Documentarse, en medida de lo posible: leer memorias técnicas si aplica, procesos o casos de uso, principalmente en fuentes internas.
* Realizar Data Discovery, Trace de datos, análisis exploratorio y si aplica también hacer uso de la estadística descriptiva, porque, puede darse el caso, por ejemplo, que se requiera información de un año ya pasado y no se tenga en los datos/tablas/archivos.
* Segmentar el problema, en este punto ya se tienen los elementos para determinar tareas a corto, mediano o largo plazo, stakeholders y viabilidad de lo requerido.
* Documentar el desarrollo del requerimiento, desde el punto 1, es importante la documentación para el seguimiento adecuado.
* Entrega/revisión de un Mínimo Producto Viable (MVP), es de suma importancia realizar esto para mostrar un esbozo del trabajo final, en algunas situaciones, con esto se atiendo lo que se requiere, en algunas otras sirve como base para ideas para proyectos mas robustos, en la mayoría de los casos, se concluye y se le da seguimiento de acuerdo con el plan generado.

1. ¿Cómo te aseguras de que la calidad de los datos sea consistente y confiable en una plataforma?

Para garantizar esto podemos asegurándolo en dos puntos:

1. Datos

* Todo lo que conlleva el ETL, sería una correcta implantación del pipeline, con su respectivo Peer review.
* Cifras Control, comparativo del periodo en cuestión contra histórico, utilizando estadística descriptiva, establecer un monitoreo automático o semiautomático, por ejemplo: numero de registros, suma acumulada de algún campo del periodo revisado, numero de archivos, etc.
* Envió de alertas automáticas

1. Plataforma

* De acuerdo con los requerimientos, una adecuada arquitectura, tomando en cuenta si se requieren procesar grandes volúmenes de datos y de igual forma si se requiere procesamiento distribuido.
* Cifras Control, pero de la plataforma, por ejemplo; latencia, disponibilidad, storage, uso de cpu, memoria, etc.
* Envió de alertas automáticas

En ambos puntos de debe documentar y realizar diagramas de las soluciones, tanto del pipe line de datos como de la plataforma, por que en ocasiones puede no ser interno de la plataforma si no, por ejemplo, una actualización de firewall y/o directivas.

1. ¿Cómo lidias con errores o fallas en los procesos de datos?

* Lo primero es tener documentado toda la solución, para poder diseñar un plan de validación
* Identificar la falla; no están llegando los datos, son parciales, el layout es distinto, etc.
* Determinar el impacto para determinar el nivel de urgencia y SLA.
* Determinar si la solución depende de nuestra área o se tiene que reportar a Soporte con Proveedor, etc.
* Implementar solución o work around, ya sea ejecución de nuestra área o externa; reiniciar/finalizar procesos, reestablecer comunicación entre módulos, rollback a actualizaciones, etc.
* Para la perdida de datos, si es la primera vez que pasa, las mismas condiciones de la plataforma te permiten tomar decisiones en base a la capacidad de procesamiento, recursos disponibles, umbrales óptimos de uso de recursos, horarios de mayor procesamiento, si es alta disponibilidad, etc. Determinar si aplica una recarga de datos en cualquier momento o gradual,
* Notificar en todo momento a los stakeholders, del inicio de la falla, posibles sistemas afectados y solución.
* En todas las posibles fallas generar un plan de acción para que los incidentes no se conviertan en problemas.
* En algunos casos no es viable implementar una solución optima y toca realizar un análisis de riesgos: para aceptarlos, trasladarlos o mitigarlos, ejemplifico; estas en una renovación tecnológica y tus plataformas legacy, necesitan reiniciar o borrar archivos diarios para su correcto funcionamiento.

1. ¿Qué consideras que es más importante: velocidad de procesamiento o precisión de los datos? ¿Por qué?

En mi experiencia es un equilibrio entre ambos puntos y se prioriza uno del otro dependiendo del fin de los datos y recursos monetarios disponibles, por ejemplo, si vas a pagar impuestos, se prioriza la precisión y no necesariamente tiene que ser en tiempo real, pero si atiendes un servicio de llamadas de voz o realidad virtual priorizas la velocidad.

1. ¿Cómo abordas la optimización del rendimiento de una plataforma de datos?

* El primer punto a considerar es no ocupar recursos con datos innecesarios. Tener un plan de autogestión de datos, borrando los que no son necesarios, por ejemplo, en una tabla que guarda el uso de cpu cada 5 minutos, para la toma de decisiones se puede generar una tabla que solo guarde un registro por hora que puede ser un promedio de las muestras o un máximo.
* Si es necesario un modelo relacional, debe estar normalizado.
* Para el entorno de bigdata se emplea el procesamiento en paralelo, optimizadores de consultas y/o particionamiento.
* Correcta comunicación entre componentes de la plataforma.

1. ¿Qué tipo de métricas consideras importantes para monitorear el estado de una plataforma de datos?

En orden de prioridad a continuación las enlisto:

* Disponibilidad, no se puede garantizar integridad si no se tiene una alta disponibilidad.
* Latencia, entre mas baja sea la latencia se garantiza una respuesta optima y se reduce una posible pérdida de paquetes.
* Si no se tiene una escalabilidad o elasticidad dinámica, monitoreo de recursos como uso de cpu, storage, memoria, etc.

1. ¿Cómo te ves contribuyendo al equipo y a la empresa en un rol de ingeniero de plataforma de datos?

Colaborando con un equipo multidisciplinario garantizando que los datos cumplan con la disponibilidad y calidad para ser de apoyo al negocio, cumpliendo con las mejores prácticas en la implementación de pipelines gestión de plataformas de datos y cumpliendo con las normas de seguridad.

1. ¿Qué estrategias utilizas para mejorar la eficiencia de los procesos de extracción, transformación y carga (ETL)?

* Detalle del funcionamiento de la fuente, como por ejemplo los horarios de actualización de datos.
* Implementar validación de datos, como por ejemplo codificación, duplicados y/o líneas vacías.
* Automatización y monitoreo de procesos, por ejemplo, en apache airflow, puedes configurar alertas si el proceso no se ejecuto de inicio a fin, revisión de logs, reiniciar el proceso en una tarea intermedia además del monitoreo de recursos asignados.
* Y adicional, es importante saber la carga de procesos simultáneos, para en medida de lo posible distribuir la carga, si son por ejemplo cargas diarias, por lo regular se programan en horarios de baja demanda de recursos, por lo general en la madrugada.

1. Describe tu experiencia con CI/CD en el contexto de la ingeniería de datos.

En este punto, lo único que he utilizado es el control de versiones en trabajo colaborativo, documentando los cambios en GIT.

1. ¿Qué ventajas ofrecen los contenedores (Docker) en el desarrollo y despliegue de plataformas de datos?

Rapidez de implementación y escalabilidad, antes se debía tener equipo físico para alguna solución, ahora no, puedes tener equipo y virtualizar o bien ya no tener equipo dedicado.

1. ¿Cómo podrías garantizar que existe una comunicación segura entre dos componentes de la plataforma?

A nivel de red que pertenezcan a la misma red y saber los puertos configurados.

1. Explica cómo gestionas el versionado de código y despliegue de los mismos (pipelines de datos).

Con el control de versiones, usando git, se puede tener el registro de los cambios y la última versión de código utilizada.

1. ¿Qué estrategias utilizas para optimizar consultas y procesos analíticos en grandes conjuntos de datos?

Lo primero es revisar que solo se consideren las columnas/datos necesarios para el análisis, el segundo es de administración que los datos que no se utilicen en ningún proceso se borren de la tabla, esto evita tener grandes volumenes en tablas y que las tablas este particionadas.

1. ¿Qué herramientas has utilizado para validar comunicaciones entre servidores ?

La principal el ping y tracert, si el primero funciona el segundo casi nunca lo ejecuto.

1. ¿Cómo sería tu proceso de *debuggeo / depurar código* en una falla que se presente en PostgreSQL o MySQL dockerizado?

Revisar los logs para identificar el código/causa de error, si es posible, ejecutar consultas aisladas cuando se contemplan varias fuentes.

1. ¿Qué se debe tomar en cuenta al momento de considerar la inclusión de una nueva tecnología a la plataforma de datos?

Considerando que ya se realizo el estudio financiero y se aprobó la compra y también el caso de que módulos se compraran, niveles de SLA’s, Soporte, etc. Lo primero es el plan de desarrollo/tiempos de la convivencia con la plataforma existente, ya que no se puede interrumpir la operación en dado caso que sea critica para el negocio, también se debe tener contemplado que funciones de deben replicar y/o mejorar en esta nueva plataforma y el periodo de pruebas y siempre considerar mejoras de acuerdo a las nuevas tecnologías.

**Casos de Uso**

1. Incidente: Se detecta un aumento repentino en el tiempo de procesamiento de un flujo de datos crítico. Describe los pasos que seguirías para determinar la causa del problema y restaurar el servicio.

* Se debe tener un proceso de notificación para todos los stakeholders
* Identificar el proceso/aplicativo
* Identificar todos los elementos involucrados/afectados.
* Si es una solución interna, ejecutarla, como por ejemplo reiniciar un proceso, detener un select \* sin condiciones en una tabla de alto volumen de datos, etc
* En dado caso que corresponda a un proveedor, escalar de forma rápida el incidente y darle seguimiento.
* Realizar pruebas
* Realizar un post-mortem.

1. Diseño de Solución: Se requiere diseñar una solución para ingerir y procesar datos de múltiples fuentes en tiempo real y ponerlos a disposición para análisis. Describe la arquitectura que propondrías y justifica tus decisiones.

* Los datos de distintas fuentes pueden proceder de apis, transacciones, bd´s, archivos, etc.
* Para la ingesta en tiempo real, he estudiado un poco, pero podría ser Kafka
* Se debe tener considerar al menos dos módulos, el de procesamiento y el de almacenamiento, el almacenamiento un datalake en GCP, y procesamiento Spark
* Utilizar un orquestador como airflow
* Un DHW, podría ser en GCP Bigquery
* Un optimizador de consultas como Trino
* Visualizar datos, con Looker, Power BI,

# **Prueba Técnica: Extracción, Transformación y Carga (ETL) con Docker**

## **Objetivo**

Desarrollar una solución completa de ETL que extraiga datos de un archivo CSV, los transforme y los cargue en una base de datos relacional (PostgreSQL o MySQL). La solución debe estar contenerizada usando Docker, y se requiere evidencia del proceso y resultados.

## **Requisitos**

1. **Archivo CSV:** Se proporcionará un archivo CSV de ejemplo con datos. Este archivo se almacenará en un contenedor Docker. (https://www.kaggle.com/datasets/cankatsrc/financial-transactions-dataset?resource=download&select=financial\_transactions.csv)
2. **ETL en Python:** Desarrollar un script de ETL en Python que realice las siguientes tareas:
   * Localizar y leer el archivo CSV desde el contenedor correspondiente.
   * Transformar los datos según sea necesario (ej., limpieza, validación, enriquecimiento).**(Eliminar los registros existentes del año 2018)**
   * Cargar los datos transformados en una base de datos PostgreSQL o MySQL (a elección del participante).
3. **Base de Datos:** Utilizar un contenedor Docker para desplegar la base de datos (PostgreSQL o MySQL).
4. **Contenedores Docker:** Crear los Dockerfiles y scripts de orquestación necesarios para construir y ejecutar los contenedores para el archivo CSV, el script ETL y la base de datos.
5. **Log de Contenedor:** Registrar en los logs del contenedor la ejecución completa del ETL.

## **Entregables**

1. **Diagrama de Solución:** Un diagrama que muestre la arquitectura general de la solución, incluyendo los contenedores, la comunicación entre ellos y el flujo de datos en Draw.io
2. **Código Fuente:** El script ETL en Python completamente comentado.
3. **Dockerfiles y Scripts de Orquestación:** Los archivos necesarios para construir y ejecutar los contenedores.
4. **Evidencia de Ejecución:**
   * Pantallas mostrando la ejecución de `docker-compose up` o comandos similares.
   * Pantallas mostrando la inserción de datos en la base de datos.
   * Pantallas del log de ejecución del ETL.

## **Criterios de Evaluación**

* **Funcionalidad:** La solución debe extraer, transformar y cargar los datos correctamente.
* **Calidad del Código:** El código debe ser legible, bien comentado y seguir buenas prácticas de programación.
* **Contenedorización:** Uso adecuado de Docker para contenerizar los componentes.
* **Log de Ejecución:** Implementación correcta del log de ejecución en el contenedor.
* **Documentación:** Claridad y completitud de la documentación entregada (diagrama, capturas de pantalla, etc.).

## **Notas Adicionales**

* Se puede utilizar cualquier librería de Python para trabajar con CSV y bases de datos (ej., `pandas`, `psycopg2`, `mysql-connector-python`).