Ingeniero de datos en NEQUI: Prueba Técnica

Propósito

El propósito de esta prueba es verificar tus conocimientos en el campo de la ingeniería de datos (indiferente de los “framework” o lenguajes de programación que utilices), esperamos que sea la oportunidad de combinar lo que haz aprendido a lo largo de tu experiencia profesional y mezclarlo con tu capacidad de abstracción y analítica.

El resultado será la base.

En este test, tienes libertad de elegir el lenguaje de programación de su preferencia
[Sugerimos **Python** ;)].

Criterios:

- Para el proceso de documentación una guía o archivo “.readme” (**Markdown**) que tenga un paso a paso de entendimiento que pueda vincular los diferentes diagramas construidos en el los diferentes pasos y la explicación y respuesta de cada paso.

- Para el proceso de administración de excepciones y logs, una carpeta dentro de a la estructura del código fuente que denote su administración y captura.

- El diseño de la arquitectura se debe realizar con recursos en la nube en un entorno **AWS**, Azure o GCP, (Linode es otra alternativa si ya te consumiste el periodo de prueba en estas nubes) donde se enumeren los pasos necesarios para canalizar los datos en el modelo de datos elegido.

- Para la entrega de la prueba publicar el proyecto final en un repositorio de **GitHub** y compartirnos el enlace.

- Debes tener como minimo 1 dataset con datos públicos.

- El dataset debe tener mas de 1 millón de datos (filas).

Continua…

Recursos [Dataset]

Recopilar los datos correctos es una de las tareas más importantes para los ingenieros de datos. A continuación, encontrará algunos recursos para encontrar conjuntos de datos que pueden servir para desarrollar la prueba.

- ⇒ Google: Dataset Search
- ⇒ Kaggle Datasets
- ⇒ Github: Awesome Public Datasets
- ⇒ Github: Public APIs
- ⇒ **Data.gov**
- ⇒ Dataquest: 18 places to find data sets for data science projects
- ⇒ KDnuggets: Datasets for Data Mining and Data Science
- ⇒ UCI Machine Learning Repository
- ⇒ Reddit: rdatasets
- ⇒ Last Call: Top 50 Most Popular APIs on RapidAPI (2018)
- ⇒ Facebook: Graph API

Continua…

Instrucciones

Paso 1: Alcance del proyecto y captura de datos

Dado que el alcance la prueba dependerá en gran medida de los datos, En este paso, debes:

- Identificar y recopilar los datos que usaras para tu proyecto.
- Explicar para qué casos de uso final deseas preparar los datos, por ejemplo: tabla de análisis, aplicación de fondo, base de datos de fuentes de verdad, etc.)

Paso 2: Explorar y evaluar los datos, el EDA.

- Explorar los datos para identificar problemas de calidad de los datos, como valores perdidos, datos duplicados, problemas de formato etc.
- Documentar los pasos necesarios para limpiar los datos, indicar que tipo de pasos se sugieren para la limpieza. **Tip** se puede usar un diagrama, mapa mental o adición en la arquitectura del paso siguiente con el fin de dejar claro este paso.

Paso 3: Definir el modelo de datos

- Trazar el modelo de datos conceptual y explicar por qué se eligió ese modelo.
- Diseñar la arquitectura y los recursos utilizados.
- Indique claramente los motivos de la elección de las herramientas y tecnologías para el proyecto.
- Proponga con qué frecuencia deben actualizarse los datos y por qué.

Paso 4: Ejecutar la ETL

Asignaremos puntos extra por cada uno de los ítems cubiertos.

- Crear las tuberías de datos y el modelo de datos
- Ejecutar controles de calidad de los datos para asegurar que la tubería funcionó como se esperaba
- Control de calidad en los datos con la integridad en la base de datos relacional (por ejemplo, clave única, tipo de datos, etc.)
- Pruebas de unidad para los “Script” para asegurar que están haciendo lo correcto.
- Comprobaciones de fuente/conteo para asegurar la integridad de los datos.
- Incluir un diccionario de datos
- Criterio de reproducibilidad

Continua…

Paso 5: Completar la redacción del proyecto

- ¿Cuál es el objetivo del proyecto?
- ¿Qué preguntas quieres hacer?
- ¿Por qué eligió el modelo que eligió?
- Incluya una descripción de cómo abordaría el problema de manera diferente en los siguientes escenarios:
    - Si los datos se incrementaran en 100x.
    - Si las tuberías se ejecutaran diariamente en una ventana de tiempo especifica.
    - Si la base de datos necesitara ser accedido por más de 100 usuarios funcionales.
    - Si se requiere hacer analítica en tiempo real, ¿cuales componentes cambiaria a su arquitectura propuesta?

> Si llegaste hasta este punto, sabrás, al igual que nosotros que la complejidad de la prueba es alta, no te desanimes. Queremos conocer tu capacidad de respuesta ante escenarios de alta presión.

¡Gracias!