Challenge Data Engineer

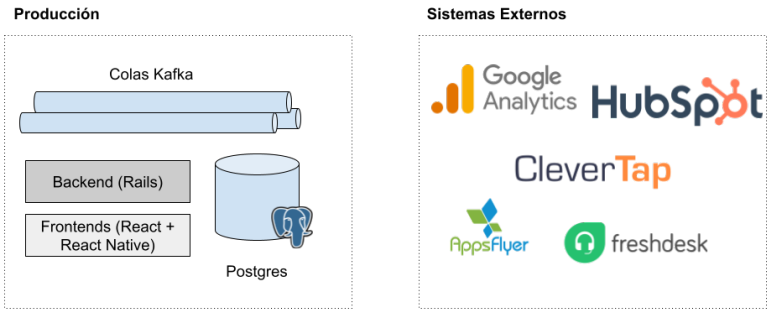
A continuación se presentan dos actividades, una de arquitectura en la cual solo deberá responder utilizando texto y diagramas; y otra de implementación en la que deberá realizar una entrega de código.

Formato de entrega: PPT o PDF + Código relacionado

Diseño y Arquitectura

LimpiApp es una startup dedicada a la venta de servicios de limpieza para personas y empresas, y están en una etapa cercana a la marcha blanca. Dentro de los distintos servicios que ofrecerán están la limpieza de casas y departamentos, personal de aseo diario o semanal, lavado de autos, lavado de ropa y cobertores otros.

Ellos ofrecen a través de una app y una web donde los clientes podrán solicitar los servicios de limpieza, y hacerle seguimiento al proceso de ejecución de las tareas en tiempo real, las cuales varían dependiendo del servicio contratado.



Su infraestructura productiva y de soporte está prácticamente lista, donde tienen los siguientes componentes:

- **Productivo:**
  - Backend en RoR
  - Frontends realizados en React (Web) y React Native (App)
  - Bases de datos en PostgreSQL
  - Kafka y tópicos relacionados a la App en tiempo real.
- **Externo:**
  - Google Analytics para los flujos comunes de la app/web.
  - CleverTap para manejar eventos en los frontends y hacer campañas de marketing digital.
  - Freshdesk para atender tickets de servicio al cliente.
  - Hubspot para manejar leads de venta con empresas.

- Appsflyer para ver atribución de campañas de marketing.

Sin embargo, no tienen infraestructura para procesamiento y analítica de datos, y la necesitarán para ciertas cosas como poder mejorar los servicios, hacer crecer el producto y hacer ofertas relevantes a los usuarios.

Por lo anterior se le asigna a ud. la tarea de **realizar una arquitectura de alto nivel para resolver este problema,** la cual debe satisfacer los siguientes requisitos:

- Debe existir algún repositorio central de datos consultable vía SQL, visualizable con herramientas de BI e integrable con algún tipo de catálogo.
- Debe permitir procesamiento tanto de tablas estructuradas como de ficheros en almacenamiento de objetos.
- El SLA mediano para las ingestas es de 2 horas, pero existen ciertos procesos que necesitarán mayores velocidades (ver apartado de modelos)
- Debe permitir procesamiento batch y real-time, ambos con algún tipo de sistema de monitoreo de procesos.
- Debe permitir enviar datos de vuelta a los servicios productivos y a los servicios externos.
- Se deberá poder correr dos modelos necesarios en el corto plazo:
    - **Modelo de fuga en tiempo real:** Analizar si es que un cliente cancelará un servicio en curso (i.e. personal en camino al domicilio)
    - **Modelo de fuga batch:** Analizar si es que un cliente no utilizará más LimpiApp.

Preguntas:

1. ¿Qué arquitectura de alto nivel propondría ud. para poder procesar los datos según los requerimientos entregados? Determine los elementos, como se conectan y los procesos que correrían sobre este a grandes rasgos.
    a. Recuerde conectar esta pregunta con los requerimientos solicitados arriba.
2. Para cada elemento de la arquitectura propuesta por ud. entregue un par de implementaciones que podrían cubrir ese espacio.
    a. Ejemplo: Si ud. propone utilizar un agendador como elemento, una implementación podría ser Apache Airflow o Prefect.

Algunas notas:

- Anote todos sus supuestos
- Puede ser SaaS, IaaS o algo intermedio.
- No es necesario que baje a detalles como vía que protocolo exportará la data desde cada sistema, solo enfocarse en las herramientas generales.
- En el caso de implementaciones puede ser cualquier solución, independiente de si eso implicara una arquitectura multi-cloud.

Implementación de soluciones

En Banco Finanzas están avanzando en su estrategia de modelos de riesgo, y sus Data Scientists ya tienen un primer modelo de propensión que quieren publicar. Este está publicado en un Jupyter Notebook que se comparte con este documento, el cual genera un Pickle con el modelo a utilizar.

A ud. como Data Engineer se le solicita su ayuda implementando un API sobre este modelo para que sea consumido por otras servicios de la empresa. Se le solicita los siguientes requerimientos:

1. Implementar API "remota" para consumir el modelo
    a. Si es REST, ofrezca endpoints para batch y single prediction.
    b. Ofrezca algún tipo de autenticación básica sobre los endpoints o API, o al menos justifique porque no es necesario.
2. Medir y reportar el rendimiento de su API
    a. Estime la complejidad, y que recursos cree que debería escalar en conjunto con la volumetría de consultas.
    b. Para esto se le entrega un set de datos en JSON para esto.
3. Planee el deployment de esta API

Bonus points:

- Usar Docker Compose o similar.
- Medir rendimiento del modelo desplegado, y comparar con el rendimiento determinado en el notebook.
- Mencionar alternativas y/o mejoras a su implementación.

Notas:

- En el caso de que su ambiente Python no acepte el Pickle, se incluye el notebook original y los datos para construir el Pickle entregado.
- En el caso de que no maneje Python, pero podría realizar este desafío en otro lenguaje, por favor comuníquese con german.oviedo@tenpo.cl para conseguir otra actividad alternativa.