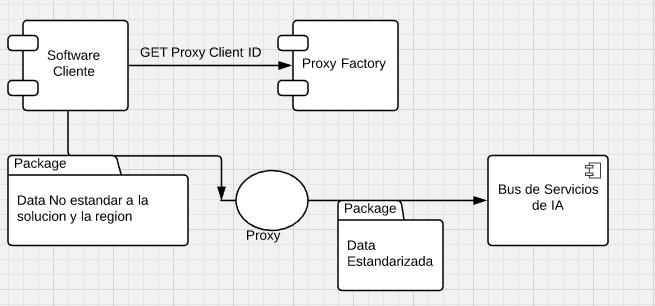
1. Una empresa que presta servicios de inteligencia artificial para grandes empresas del sector médico alrededor del mundo, se enfoca en procesar imágenes de rayos X para dar diagnósticos sobre un número de enfermedades determinado. Este objetivo es logrado por distintos servicios en la nube que provee esta empresa. Dada la naturaleza de las distintas regiones, unas enfermedades se presentan con mayor frecuencia y en circunstancias distintas que en otras regiones. Adicionalmente, cada cliente cuenta con sistemas distintos para proveer información del paciente y de sus imágenes. Actualmente la empresa solo cuenta con 3 grandes clientes en 3 regiones y le cuesta mucho adaptar su tecnología a cada nuevo cliente en otras regiones. Dado este escenario, usted fue contratado para llevar la tecnología de esta empresa al siguiente nivel reduciendo sus tiempos de implementación de meses a semanas con cada cliente. Responda las siguientes preguntas:

● Cuales son los principales drivers que tendría en cuenta para diseñar una solución?

1. La disparidad Tecnología, puede causar serios inconvenientes en la implementación.
2. Todo debe estar en la nube para que pueda ser accesible por cada cliente, se sugiere sistemas de alta disponibilidad.
3. La región es clave a la hora de enviar la información para que esta sea usada en los diagnósticos estadísticos de las imágenes médicas.

● Qué patrones o tácticas aplicaría?

1. Montaría un patrón proxy el cual me permita estandarizar la entrada de datos al sistema de información recibiendo información distinta de cada cliente, así para cada cliente nuevo solo se debe extender el proxy para que transforme la nueva información a la entrada estándar del sistema.
2. Generaría un patrón fabrica que me permita cambiar de proxy para cada cliente manteniendo altos niveles de abstracción.

● Muestre una vista lógica de la solución propuesta y describa la razón de ella.

La idea de la solución es estandarizar los imputs a los servicios de IA sin tener que obligar a cada cliente a generar un desarrollo la idea es permitirle seguir enviando su data en el modo que la tenga y hacerle una transformada en el proxy para que esta se estandarice a los servicios cambiando la complejidad de tener que rehacer todo el core de IA para cada cliente nuevo a tan solo implementar un nuevo proxy traductor de información.

2. Escriba el código base con el que implementaría un patrón de diseño Observer para un escenario donde se generan notificaciones push cada vez que se encuentra una actualización en los datos de la cuenta de un usuario.

Observador:

<https://github.com/AlterCodex/Alkanza/blob/master/Observer.py>

Sujeto:

<https://github.com/AlterCodex/Alkanza/blob/master/Subject.py>

Usuario:

<https://github.com/AlterCodex/Alkanza/blob/master/User.py>

3. ¿Cómo modelarías el reino animal (con especies y su comportamiento) como un sistema de clases?

La clasificación taxonómica se haría mediante herencia, las clases se deben mantener abstractas por todo el árbol de herencia hasta encontrar el ultimo nivel de herencia donde deben ser concretas, las características comunes en las especies se mantienen como atributos durante la herencia y los comportamientos se encapsularían en interfases para su reusó en diferentes especies

Ejemplo de un modelamiento parcial:

<https://go.gliffy.com/go/publish/12625182>.

4. Describa el proceso que utiliza para desarrollar código desde la captura de requerimientos hasta la salida a producción.

1. Entendimiento del negocio: lo principal es entender cuál es el proceso de negocio a realizar, esto se puede hacer hablando con las personas que usaran el software, y en caso de que el proceso ya exista (sea manual o con otro software) se debe hablar con aquellos que realizan el proceso para determinar qué es lo que la solución debe hacer y que sea la forma más simple para el usuario.
2. Historias de Usuario: dado el análisis inicial se deben escribir las tareas del proceso para el resultado final de la solución, se sugiere el formato <<Actor>> <<Acción>>: <<Resultado>>
3. Se deben Priorizar las tareas para cumplir la meta según la importación para el proceso y definir modelo de Arquitectura e Infraestructura para la carga esperada, siempre dejando la puerta para escalar cuando la solución crezca.
4. Se agrupan las historias de usuario o tareas para cada iteración intentando mantener entre 1 a 4 semanas el tiempo que tome realizarlas y teniendo en cuenta que al final de cada iteración el producto parcial debe ser un entregable que se pueda probar.
5. Al final de cada iteración se debe probar el producto como unidad y en conjunto a todo lo que se haya realizado en iteraciones anteriores, dado el avance se retroalimenta para evitar repetir errores o impedimentos en la siguiente iteración.
6. Para cada final de iteración que se aprueben los cambios realizados se debe desplegar ya sea usando algún sistema automatizado o manualmente la solución para ser probada en ambientes de QA antes de llegar a producción.
7. Según el modelo de arquitectura es posible entregar soluciones parciales al cliente en el ciclo de software para que este las use en conjunto al negocio esto es clave para que se pueda tener retroalimentación del usuario final.
8. Repetir desde el punto 4 hasta cumplir todos los procesos de negocio y terminar por completo el producto.