**Examen Teórico**

**Sección teórica**

1.- Describe tu experiencia en Python

2.- ¿Qué diferencia existe entre una clase y un objeto?

3.- ¿Qué entiendes por la mantenibilidad de código?¿Es importante, si es así, mencionar el porqué?

4.- ¿Qué son los olores de código? ¿Qué utilizarías para eliminarlos?

5.- ¿Qué es un patrón de diseño? ¿Es importante, si es así, mencionar el porqué?

6.- ¿Qué es la metodología TDD? ¿Es importante, si es así, mencionar el porqué?

7.- ¿Qué es la metodología BDD? ¿Es importante, si es así, mencionar el porqué?

8.- Menciona que tipos de prueba de código conoces

9.- Describe el paradigma de programación orientado a objetos

10.- ¿Qué es la modularidad? ¿Es importante, si es así, mencionar el porqué?

11.- ¿Qué es la abstracción? ¿Es importante, si es así, mencionar el porqué?

12.- ¿Qué es el encapsulamiento? ¿Es importante, si es así, mencionar el porqué?

13.- ¿Qué es la herencia? ¿Es importante, si es así, mencionar el porqué?

14.- ¿Qué es el polimorfismo? ¿Es importante, si es así, mencionar el porqué?

15.- Describe que es un diagrama entidad-relación

**Sección práctica**

1.- Si te asignan el rol de líder técnico en un nuevo proyecto de desarrollo de Software, sabiendo que el alcance del proyecto es grande, te consultan sobre si se debe dar prioridad a la mantenibilidad o al rendimiento, también se indica que los dos vitales para el cliente, en base a tu experiencia ¿Qué recomendarías al equipo?.

2.- Si en el transcurso del trabajo encuentras un problema en tu código, es algo que puedes arreglar investigando la documentación relacionada, sin embargo ya has gastado un buen tiempo revisando el defecto. ¿Comunicarías el obstáculo o ahorrarías tiempo investigando rápido la solución?

3.- Tu compañero de trabajo ha faltado por temas imprevistos, se te ha asignado ajustar su código por tus altos conocimientos en el tema, sin embargo, cuando revisas su código, detectas que este es casi incomprensible, pero si funciona. ¿Qué acción realizarías para ajustar de la mejor forma el código de tu compañero?

4.- Reflexionando sobre la pregunta 3, ¿Poniéndote en el lugar de tu compañero, qué harías para que tus compañeros de trabajo entiendan tu código?

5.- Imagina que te encuentras trabajando con un equipo que no conoce la importancia de la mantenibilidad, lamentablemente por su falta de conocimiento creen que invertir tiempo en la mantenibilidad de código es una pérdida de tiempo, y gracias a esto el proyecto está teniendo malos resultados, el gestor de proyecto inicia un reunión con todos los integrantes del proyecto para realizar una retrospectiva, cuando te toque hablar, ¿Qué le dirías a tu equipo para que entiendan la gran importancia de invertir tiempo en la mantenibilidad de código?

6.- Te encuentras trabajando en un proyecto de desarrollo de Software con una duración de dos años, donde no se utilizan metodologías de prueba y el flujo de trabajo está bien definido, el área de análisis de negocio se encarga de entender las funcionalidades del sistema, el área de UX/UI se encarga de entender la documentación del área de negocio y genera prototipos de interfaces gráficas, las áreas de Backend y Frontend revisan la documentación y los prototipos, en base a estos generan sus códigos, vistas y pruebas, finalmente el área de Calidad verifica lo desarrollado.El flujo de trabajo funcionaba correctamente durante los seis primeros meses del proyecto, sin embargo, poco tiempo después se encontraron muchos errores, Calidad siempre solicita ajustes en el código realizado, el área de desarrollo genera siempre funcionalidades que no están alineadas con lo que necesita el cliente, es como si existiera un teléfono malogrado en el flujo de trabajo, existen muchas causas posibles de este problema, pero tu misión es detectar problemas en relacionadas al área de Backend, ¿Qué recomendarías a tu equipo?

**Sección ética, valores y habilidades blandas**

1.- Describe los tres valores humanos más importantes que crees que debe tener un desarrollador Backend

2.- ¿Por qué es importante la comunicación en un equipo?

3.- ¿Por qué es importante la empatía en un equipo?

4.- ¿Cómo manejarías información delicada en el trabajo?

5.- Actualmente los proyectos ágiles se caracterizan por la alta colaboración entre el equipo, ¿Que es la colaboración?¿Es importante, si es así, mencionar el porqué?

**Buenas prácticas**

1.- El mundo del desarrollo de Software está en constante cambio, ¿Crees que es importante conocer las tendencias de desarrollo Backend?, si es así, ¿Cómo te mantendrás al día en el desarrollo Backend?

2.- ¿Por qué es importante conocer las buenas prácticas?

**Documentación**

1.- ¿Crees que es importante la documentación técnica en un proyecto? Si es así, mencionar el porqué

**Examen Práctico**

Como reto de ingreso se ha seleccionado el siguiente problema para que lo aplique utilizando sus conocimientos en desarrollo Backend:

* Se debe crear un servidor REST API utilizando el framework Flask
* La conexión y consultas con la BD Postgres se realizarán utilizando el ORM SQLAlchemy, este se debe conectar con Flask
* El servidor está pensado para satisfacer servicios relacionados a una veterinaria, se desea desarrollar un entretenido juego para sus clientes sobre gestión de mascotas y enseñarle información sobre animales, para eso se ha realizado un documento que explica algunas características del sistema que se desea, las cuales se describen a continuación:
  + Métodos para Persona:
    - 1: Nombrar los nombre de todas sus mascotas
    - 2: Dar comida a una mascota (La mascota solo aceptará la comida que le gusta, no se acepta comida)
    - 3: Preparar comida
    - 4: Obtener una nueva mascota
    - 5: Revisar si la comida se ha podrido (verificar fecha de caducidad y podrir la comida si se paso)
    - 6: Comer (Aumenta su salud)
    - 7: Dormir (Aumenta su estado de sueño)
    - 8: Revisar mascotas (Si la mascota tiene salud <=0 debe morir)
    - 9: Presentarse (Mostrar su información)
  + Métodos para Mascotas
    - 1: Comer comida (Si come comida podrida, se le restara 40 de salud, sino aumentar 70 de salud)
    - 2: Saludar a la persona que lo cuida diciendo su comida favorita
    - 3: Comunicarse (Se puede usar un print)
    - 3: Jugar con otra mascota (Las mascotas solo juegan con otras mascotas del mismo tipo)
    - 4: Dormir (Aumenta su estado de sueño)
    - 5: Morir
  + Métodos para la comida
    - 1: Podrirse (La comida se pudre)
    - 2: Información (Describe quien lo preparó, y que mascotas pueden alimentarse de esta)
* Tu tarea es entender los requisitos del documento que se entregó y convertirlo en un conjunto de APIs REST
* Se debe también realizar algunas pruebas de las APIs que se han generado, estas se desarrollarán utilizando Behave
* Se debe mapear las APIs utilizando la librería Swagger
* Las validaciones de los tipos de datos de entrada deben realizarse utilizando la librería Cerberus
* Al finalizar el entregable, se validará la calidad realizando las siguientes actividades:
  + Se revisará el código desarrollado utilizando la librería Pylint para realizar un análisis estático
  + Se analizará la mantenibilidad del código
  + Se analizará la aplicación del paradigma de programación orientado a objetos
  + El uso de patrones de diseño dará un plus en la calificación