**Conocimientos QA**

1. **¿Qué es QA y por qué es importante en el desarrollo de software?**

QA o por sus siglas en ingles Quality assurance es un conjunto de practicas utilizadas en el desarrollo de software para garantizar que el producto, programa y/o software entregado al usuario final, cumpla con la calidad prometida al cliente. QA esta conformado por un conjunto de procesos que tiene como objetivo identificar problemas con antelación para que estos no lleguen a producción o al usuario final, es por esto mismo que se deja en evidencia la importancia del QA en el desarrollo de software puesto que garantizar la calidad de nuestro software es lo mas importante a la hora de desarrollar y emprender en un proyecto. Basándonos en metodología como SCRUM, DEVOPS y demás, podemos ver que el QA trabaja bajos los principios utilizados por estas metodologías, ahorrar tiempo y recursos, mejorar la productividad y colaboración de los equipos, y aumentar la satisfacción del cliente.

El QA es muy importante en el desarrollo de software porque nos permite identificar problemas o situaciones de forma temprana y de esta forma poder en cada ciclo del proceso ir garantizando la calidad del software a la par de avanzar en el desarrollo de este.

1. **¿Cuál es la diferencia entre pruebas manuales y pruebas automáticas?**

Las pruebas manuales son pruebas ejecutadas directamente por testers, tratando de simular las actividades que un usuario final o cliente podría realizar en el software, estas son muy comunes al principio del desarrollo puesto que se prueban las funciones esenciales del software, y al final del desarrollo cuando ya se ha creado algún tipo de interfaz para el usuario y asi simular su comportamiento.

Las pruebas automáticas son pruebas creadas tambien por los testers pero con ayuda de scripts, normalmente para pruebas que si fueran a realizarse de forma manual, serian muy largas, poco eficientes y certeras. Las pruebas automatizadas pueden ser ejecutadas sobre largas porciones de código una y otra vez, para asi garantizar la fiabilidad del software creado.

1. **¿Qué son los casos de prueba y cómo los crearías?**

Los casos de prueba son esencialmente un conjunto de casos o pasos a verificar para saber si el software desarrollado cumple con los criterios de calidad y funciona de la forma correcta y anticipada.

Para crear un caso de prueba primero hay que identificarlo con un ID, despues de esto tenemos que describir este caso de prueba, que función se va a verificar, etc, seguido de esto hay que saber si hay alguna condición previa para la ejecución de esta prueba, el siguiente paso seria definir este caso de prueba, o sea escribir los pasos a ejecutar de una forma comprensible, despues de todo esto ya podríamos ejecutar el caso de prueba y recopilar la información que nos haya arrojado este mismo.

1. **¿Cuál es el propósito de la prueba de regresión?**

Las pruebas de regresión tienen el propósito de garantizar, que despues de un cambio, corrección o nueva versión de un software, este siga funcionando de la forma prevista y no haya sucedido ninguna regresión.

**RESOLUCION DE PROBLEMAS**

Supongamos que eres responsable de probar la pasarela de pagos de NEU, y los usuarios han informado que, ocasionalmente, el proceso de pago falla sin dar ningún mensaje de error. El equipo de desarrollo no ha identificado la causa raíz aún. ¿Cuáles serían los pasos que seguirías para investigar y resolver este problema? .

**SOLUCION**

Para la solución de cualquier problema, no solo en el ámbito de desarrollo de software o QA, lo primero que debemos hacer o tener es una clara definición de cual es el problema, en este caso sabemos que es un problema relacion con la pasarela de pagos y para saber mas sobre este problema lo mas oportuno seria reproducir este error, hacer un tipo de prueba manual simulando el comportamiento de un usuario en esta pasarela de pagos y del mismo modo ejecutar pruebas automatizadas y regresión para corroborar la salud del sistema, despues de realizar todas estas pruebas de forma rigurosa se debería tener suficiente información para analizar y de esta forma identificar el problema en cuestión. Despues de haber identificado el problema el siguiente paso será informar y colaborar con el equipo de desarrollo para que todos tengan un mejor entendimiento del problema y asi el equipo pueda proponer soluciones para resolver este problema, ya sea eliminar o agregar nuevas características al sistema, todos estos cambios deben ser testeados con la misma rigurosidad cuando se estaba tratando de encontrar el problema para asi evitar errores o regresiones futuras, por ultimo despues de implementados estos cambios es importante monitorear y hacerle un seguimiento al sistema para saber que ese funcionando de la forma correcta y deseada y que el error en efecto ya no se presenta.

**PRACTICA**

Al correr el código en Visual Studio Code, pasándole una lista por parámetro podemos ver que la función no cumple con su objetivo de organizar la lista. Despues de analizar el código podemos evidenciar que se está utilizando el método de ordenamiento de burbuja el cual compara 2 elementos seguidos para saber cual es el mayor y asi intercambiarlos para este método ser efectivo tiene que ser repetido el numero de elementos que tenga para eso estan los 2 ciclos for. Al debuggear el código podemos ver que el error se encuentra en la línea 6 donde se intercambian los valores despues de hacer la comparación de mayor o menor, más específicamente aquí:

input\_list[j], input\_list[j] = input\_list[j + 1], input\_list[j]

Esta línea de código simplemente no realiza el intercambio entre los 2 elementos comparados y para que la función cumpla su cometido debería estar asi:

input\_list[j], input\_list[j + 1] = input\_list[j + 1], input\_list[j]

Aqui podemos ver que en la posición del índice [j], en caso de que el elemento en j sea mayor que el de [j + 1], se estaría cambiando por el elemento de la posición [j+1] y viceversa (o sea el elemento de [j + 1] se intercambiaría con el elemento de [j]).

De esta forma la función cumple con su cometido de forma correcta.