

Prueba de evaluación de conocimientos

Perfil básico Analista de Calidad de Software

Nombre: Rafael Arturo Valdez Rodriguez

Fecha: 04/04/2023

Empresa: Smart Testing

Básicos sobre análisis de calidad de software:

1. Favor contestar con sus propias palabras:
 - A. ¿Qué entiendes por plan de pruebas?
 - a. Es un documento donde se plasma los objetivos, la estrategia de trabajo y la elaboración paso por paso de las actividades de prueba.
 - B. ¿Cuál es la diferencia entre caso de pruebas y escenario de pruebas?
 - a. Un caso de prueba una única funcionalidad, mientras que el escenario de pruebas es un conjunto de casos de pruebas que forman un flujo de trabajo o workflow.
 - C. Menciona el orden de ejecución de las tareas de QA que realizas dentro de un proyecto.
 - a. Análisis de requerimientos
 - b. Planificación de pruebas
 - c. Diseño de pruebas
 - d. Implementación de ambientes de pruebas
 - e. Ejecución de pruebas
 - f. Reportes de pruebas
 - g. Cierre de prueba
 - D. Diferencia entre pruebas de caja blanca y pruebas de caja negra. Mencione algunos ejemplos.
 - a. Las pruebas de caja toman en cuenta el procesamiento de una entrada de datos. Se realizan pruebas al código interno de la aplicación. Algunas de las técnicas de prueba de caja blanca son: Cobertura de código, pruebas de flujo de data y pruebas de Branch.
 - b. Las pruebas de caja negra evalúan la salida dada una entrada en un sistema, sin tomar en cuenta el proceso interno. Algunas técnicas de pruebas de caja negra son: pruebas de partición equivalente, pruebas de valores limites y pruebas de transición de estados.
 - E. Liste los tipos de pruebas que conozca.
 - 1) Pruebas de aceptación
 - 2) Pruebas de accesibilidad
 - 3) Pruebas ad-hoc
 - 4) Pruebas alfa
 - 5) Pruebas automatizadas

- 6) Pruebas de caja Negra
- 7) Pruebas de valor límite
- 8) Pruebas de compatibilidad de navegadores
- 9) Pruebas de compatibilidad
- 10) Pruebas de cumplimiento
- 11) Pruebas de componentes
- 12) Pruebas de concurrencia
- 13) Pruebas de configuración
- 14) Pruebas cruzadas funcionales
- 15) Pruebas de datos
- 16) Pruebas de bases de datos
- 17) Pruebas de dependencias
- 18) Pruebas de E2E
- 19) Pruebas exploratorias
- 20) Pruebas funcionales
- 21) Pruebas de interfaz gráfica de usuario
- 22) Pruebas incrementales
- 23) Pruebas de instalación
- 24) Pruebas de integración
- 25) Pruebas de interoperabilidad
- 26) Pruebas de carga
- 27) Pruebas de localización
- 28) Pruebas de monkey
- 29) Pruebas negativas
- 30) Pruebas no funcionales
- 31) Pruebas de rendimiento
- 32) Pruebas piloto
- 33) Pruebas positivas
- 34) Pruebas de recuperación
- 35) Pruebas de regresión
- 36) Pruebas de confiabilidad
- 37) Pruebas de requisitos
- 38) Pruebas basadas en riesgos
- 39) Pruebas de escalabilidad
- 40) Pruebas de seguridad
- 41) Pruebas de humo
- 42) Pruebas de estabilidad
- 43) Pruebas de estrés
- 44) Pruebas del Sistema
- 45) Pruebas unitarias
- 46) Pruebas de usabilidad
- 47) Pruebas de aceptación de usuario
- 48) Pruebas de volumen
- 49) Pruebas de vulnerabilidad

- F. ¿Cuál es la importancia de las pruebas automatizadas sobre las pruebas manuales?
- Son dos tipos de pruebas diferentes. Lo ideal es usar ambas.
 - La diferencias que tienen las automatizadas contra las manuales, es que permite ejecutar un alto volumen de pruebas en muy corto tiempo.
 - La diferencia que tienen las manuales contra las automatizadas, es que este tipo de pruebas encuentra los defectos mas valiosos o críticos, porque tiene mucha flexibilidad para ejecutar casos de pruebas muy elaborados basado en un profundo conocimiento del sistema.

2. Asigne la tarea al área correspondiente según aplique: QA, Desarrollo, Scrum, PO, Funcional.

Tarea	Área
Pruebas unitarias	Desarrollo
Pruebas de integración	QA
Pruebas de sistema	QA
Certificación de pruebas	Funcional
Recolección de data	QA
Gestión de las necesidades del equipo	Scrum Master
Elaboración de historias	Product Owner (PO)
Refinamiento de historias	QA, Desarrollo, Scrum Master
Revisión de plan de pruebas	Funcional

3. Responda Verdadero o falso

- F** Issue productivo, defecto, incidente y bug son sinónimos
- F** El QA es tan responsable como el desarrollador de los defectos encontrados en Producción.
- F** En un ciclo "normal" de desarrollo de software las pruebas se realizan directamente en el ambiente de PRODUCCION, dado que como los tiempos son muy cortos, el cliente tiene mucho apuro, debe salir urgente a producción.
- F** Para comenzar a probar un producto siempre es necesario definir los escenarios de prueba.

QA en APAP

1. Responda correctamente:

- ¿Cuál (es) es (son) el (los) documento(s) encargados de llevar los lineamientos de control de actividades y responsabilidades del área de QA?

- i. Plan de aseguramiento de la calidad
 - ii. Procedimientos de control de la calidad
 - iii. Trazabilidad
 - iv. Informe de pruebas
- b. ¿Cuáles son las responsabilidades del Analista de Calidad en APAP?
 - i. Diseño de casos de prueba
 - ii. Ejecución de pruebas
 - iii. Reporte de defectos
 - iv. Participar en el Sprint Planning
 - v. Participar en el Grooming Session (Refinamiento)
 - vi. Participar en los Daily StandUp Meeting
- c. ¿Cuál es el documento que debemos exigir para las pruebas de performance y que debe estar aprobado por arquitectura?
 - i. Registro de pruebas
 - ii. Informe de prueba de rendimiento
 - iii. Plan de prueba de rendimiento
- d. ¿Cuáles son las fases de un SQA y que se espera en cada fase?
 - i. Nuevo: Cuando el desarrollador terminar su tarea y crea el SQA para que el área de QA empiece con las pruebas al desarrollo.
 - ii. En Proceso: Cuando las pruebas están en curso por parte del área de QA.
 - iii. Cerrado: Cuando el funcional certifica que las pruebas realizadas al desarrollo cumplen con lo requerido.
- e. ¿Qué debemos hacer en caso de identificar un hallazgo que nos impide continuar las pruebas, pero no es un defecto del proyecto en el que estamos?
 - i. Si es un defecto bloqueador se reporta y se informa al equipo SCRUM, para tratarse con la mayor prioridad.
- f. ¿Cuáles son los entregables de analista de aseguramiento?
 - i. Test Cases, Test Suite, Test Result.
- g. ¿Cómo se llama la biblioteca de QA donde esta toda la información sobre los procesos y manuales; donde está ubicada?
 - i. [QA Team - General - All Documents \(sharepoint.com\)](#)

2. Responda Verdadero (V) o falso (F) según considere.

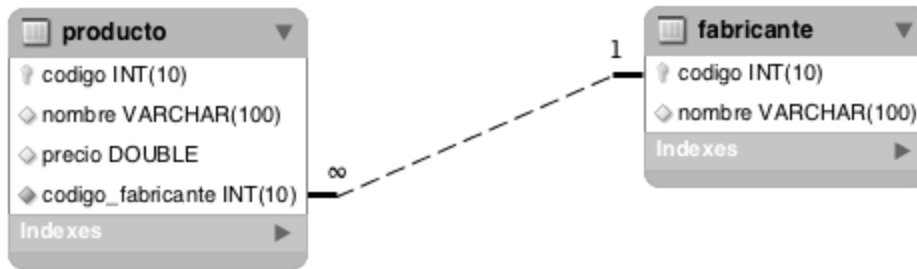
- a. V Las pruebas de contabilidad deben realizarse siempre que haya un movimiento en una cuenta
- b. V Siempre que haya un cambio en el desarrollo es necesario realizar pruebas de no afectación
- c. F La solicitud de pruebas (SQA) debe ser creada por desarrollo antes del pase a producción, no antes de iniciar las pruebas.
- d. F No es necesario que la solicitud de pruebas (SQA) tenga la guía de instalación y las fuentes, así como los dueños funcionales y técnicos.
- e. F Los defectos pueden ser reportados por teams sin necesidad de registrarlos en Jira.
- f. V Si tengo alguna situación que afecte mi trabajo debo reportarlo a mi supervisor inmediato y a mi equipo de proyecto.
- g. V Los defectos deben ser vinculados a la solicitud de pruebas.

- h. F Solo es necesario que el PO apruebe los cambios de cualquier sistema para ir a producción.

Sobre SQL:

1. ¿Cuáles son las diferencias entre los comandos "delete" y "truncate"?
 - a. DELETE se utiliza para eliminar filas de una tabla de forma selectiva
 - b. TRUNCATE se utiliza para eliminar todas las filas de una tabla de una sola vez.
 - c. TRUNCATE es más rápido que DELETE pero no permite la recuperación de datos.
2. ¿Qué son 'table' y 'field'?
 - a. Una 'table' es una colección de datos organizados en filas y columnas
 - b. Un 'field' representa un atributo de los objetos almacenados en la tabla.
3. ¿Qué es 'join'?
 - a. El comando 'join' es una operación que se utiliza para combinar dos o más tablas basándose en una columna común. El resultado es otra tabla que contiene las filas de la tabla original que cumplen con la condición. Hay 4 tipos de join: inner join, right join, left join y full outer join.
4. ¿Cuál es la diferencia entre Primary Key y Unique Key?
 - a. La 'Primary Key' se utiliza para identificar de forma única cada fila de una tabla, mientras que la 'Unique Key' se utiliza para garantizar que los valores de una o varias columnas de una tabla sean únicos. La 'Primary Key' no puede permitir valores nulos, mientras que la 'Unique Key' puede permitirlos.
5. ¿Puedes decir que los valores NULL equivalen a cero?
 - a. El valor NULL y cero no son equivalentes en SQL. El valor NULL se utiliza para representar la ausencia de un valor en una columna, mientras que cero es un valor numérico que puede ser almacenado en una columna.
6. ¿Cuál es el uso principal para 'stored procedure'?
 - a. Un 'Stored Procedure' es un conjunto de instrucciones SQL que se guarda en la base de datos y que puede ser invocado desde una aplicación o desde otra consulta SQL. Algunas de las características que ofrece los SP son: Modularidad y reutilización de Código, Mejora del rendimiento al estar precompilados, Seguridad y control de acceso, y Facilita la migración de datos.

Basado en el siguiente modelo entidad/relación realizar los siguientes ejercicios:



1. Crear base de datos.
 - a. CREATE DATABASE Almacen;
2. Crear tablas e Insertar datos en cada tabla
 - a. Crear tabla producto:

```

CREATE TABLE producto (
  codigo INT,
  nombre VARCHAR(100),
  precio DOUBLE,
  codigo_fabricante INT
);
  
```

- b. Crear tabla fabricante:

```

CREATE TABLE fabricante (
  codigo INT,
  nombre VARCHAR(100)
);
  
```

- c. Crear la relación entre las tablas:

```

ALTER TABLE producto
ADD CONSTRAINT fk_codigo_fabricante
FOREIGN KEY (codigo_fabricante)
REFERENCES fabricante(codigo);
  
```

- d. Insertar data en las tablas producto y fabricante:

```

INSERT INTO producto (codigo, nombre, precio, codigo_fabricante)
VALUES
(1, 'Producto 1', 19.99, 1),
(2, 'Producto 2', 29.99, 2),
(3, 'Producto 3', 39.99, 1);
  
```

```

INSERT INTO fabricante (codigo, nombre)
VALUES
(1, 'Fabricante Verde'),
(2, 'Fabricante Marron'),
(3, 'Fabricante Blanco');
  
```

3. Listar nombres de todos los productos de la tabla "Productos"

```

SELECT * FROM producto;
  
```

4. Lista los nombres y los precios de todos los productos de la tabla producto.

```
SELECT nombre, precio FROM producto;
```

5. Lista los nombres y los precios de todos los productos de la tabla producto, redondeando el valor del precio.

```
SELECT nombre, ROUND(precio) AS precio_redondeo FROM producto;
```

6. Lista el nombre y el precio del producto más barato. (Utilice solamente las cláusulas ORDER BY y LIMIT)

```
SELECT nombre, precio FROM producto ORDER BY precio ASC LIMIT 1;
```

7. Devuelve una lista con el nombre del producto, precio y nombre de fabricante de todos los productos de la base de datos.

```
SELECT a.nombre AS nombre_producto, a.precio, b.nombre AS nombre_fabricante  
FROM producto a  
JOIN fabricante b ON a.codigo_fabricante = b.codigo;
```

8. Calcula el número total de fabricantes que hay en la tabla Fabricante.

```
SELECT COUNT(*) AS total_fabricantes  
FROM fabricante;
```

9. Calcula la suma de todos los productos de uno de los fabricantes

```
SELECT SUM(a.precio) AS total_productos  
FROM producto a  
JOIN fabricante b ON a.codigo_fabricante = b.codigo  
WHERE b.nombre = ' Fabricante Verde';
```

10. Crear un “Stored procedure”

```
CREATE PROCEDURE calcular_precio_total_por_fabricante(IN fabricante VARCHAR(50))  
BEGIN  
    SELECT b.nombre AS fabricante, SUM(a.precio) AS precio_total  
    FROM producto a  
    JOIN fabricante b ON a.codigo_fabricante = b.codigo  
    WHERE b.nombre = fabricante  
    GROUP BY b.nombre;  
END;
```

Prueba practica:

De la siguiente página: <https://www.saucedemo.com/inventory.html>

Realice las siguientes tareas:

1. Crear plan de prueba.
 - a. Ver documento Plan de Pruebas
2. Diseñe los casos de pruebas
 - a. Ver documento Casos de Pruebas
3. Definir alcance de las pruebas
 - a. Está incluido en el Plan de Prueba.
4. Reporte de defecto.
 - a. Ver documento Defectos
5. Documenta algunos casos de pruebas.
 - a. Documentación Casos de pruebas.