**Repuestas**

Prueba de evaluación de conocimientos- Perfil básico Analista de Calidad de Software.

Nombre: \_\_\_\_Cethy Hernández Almonte

Fecha: \_\_25/04/2023

Empresa: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Básicos sobre análisis de calidad de software:**

Favor contestar con sus propias palabras:

1. **¿Qué entiendes por plan de pruebas?**

Este estables la forma en que se ejecutaran las pruebas, dando una descripción de como se probará un sistema o aplicación de software, agregando con esta información sobre el alcance de las pruebas, los recursos necesarios, los criterios de aceptación, los objetivos de las pruebas, etc.

1. **¿Cuál es la diferencia entre caso de pruebas y escenario de pruebas?**

Un caso de prueba, es una descripción detallada de una acción a tomar y validar el comportamiento de una aplicación o un sistema.

Un escenario de prueba, este compuesto por uno o varios casos de pruebas, es decir que se busca realizar una evaluación completa del comportamiento y funcionalidad del sistema o software

1. **Menciona el orden de ejecución de las tareas de QA que realizas dentro de un proyecto.**

1-Analisis y especificaciones

2-Diseño caso de prueba

3-Ejecucion de prueba manuales

4-Reporte de bugs

5-validación de los bugs corregidos

1. **Diferencia entre pruebas de caja blanca y pruebas de caja negra. Mencione algunos ejemplos.**

Pruebas de caja blanca: este realiza las pruebas en la estructura interna, es decir, código fuente, como pruebas unitarias o de integración y regresión.

Pruebas de caja negra: esta mayormente se usa para las pruebas funcionales, de aceptación del usuario y estrés

1. **Liste los tipos de pruebas que conozca.**

\*Prueba Unitarias

\*Pruebas caja blanca y caja negra

\*Prueba regresión

\*Prueba seguridad

\*Prueba funcionales

\*Prueba de aceptación usuario

1. **¿Cuál es la importancia de las pruebas automatizadas sobre las pruebas manuales?**

La importancia de las pruebas automatizadas, es que ahorra tiempo y recursos, es más rápida y permite pruebas mas frecuentes. Por otro lado, las pruebas manuales pueden detectar problemas que con las pruebas automatizada no podemos verificar, como errores de diseño y usabilidad.

1. **Asigne la tarea al área correspondiente:**

Area: QA, Desarrollo, Scrum, PO, Funcional.

|  |  |
| --- | --- |
| **Tarea** | **Área** |
| Pruebas unitarias | QA, Desarrollo |
| Pruebas de integración | QA, Desarrollo |
| Pruebas de sistema | QA, Desarrollo |
| Certificación de pruebas | QA |
| Recolección de data | Funcional |
| Gestión de las necesidades del equipo | Scrum |
| Elaboración de historias | PO, Funcional |
| Refinamiento de historias | PO, Funcional |
| Revisión de plan de pruebas | QA |

1. **Responsa Verdadero o falso**
2. Issue productivo, defecto, incidente y bug son sinónimos **\_F\_**
3. El QA es tan responsable como el desarrollador de los defectos encontrados en Producción. **\_\_V\_\_**
4. En in 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\_\_\_**
5. Para comenzar a probar un producto siempre es necesario definir los escenarios de prueba. **\_\_V\_\_\_**
6. **Responda Verdadero (V) o falso (F) según considere.**
   1. ¿Las pruebas de contabilidad deben realizarse siempre que haya un movimiento en una cuenta? **\_\_F\_\_\_\_**
   2. Siempre que haya un cambio en el desarrollo es necesario realizar pruebas de no afectación **\_\_F\_\_\_**
   3. La solicitud de pruebas (SQA) debe ser creada por desarrollo antes del pase a producción, no antes de iniciar las pruebas. **\_\_\_F\_\_\_**
   4. No es necesario que la solicitud de pruebas (SQA) tenga la guía de instalación y los fuentes, así como los dueños funcionales y técnicos. **\_\_F\_\_**
   5. Los defectos pueden ser reportados por teams sin necesidad de registrarlos en Jira. **\_\_V\_\_\_**
   6. Si tengo alguna situación que afecte mi trabajo debo reportarlo a mi supervisor inmediato y a mi equipo de proyecto. **\_V\_\_\_**
   7. Los defectos deben ser vinculados a la solicitud de pruebas. **\_\_V\_\_**

**Sobre SQL:**

1. **¿Cuáles son las diferencias entre los comandos "delete" y "truncate"?**
2. **¿Qué son ‘table’ y ‘field’?**
3. **¿Qué es ‘join’?**
4. **¿Cuál es la diferencia entre Primary Key y Unique Key?**
5. **¿Puedes decir que los valores NULL equivalen a cero?**
6. **¿Cuál es el uso principal para ‘stored procedure’?**

**Repuestas:**

1. "DELETE" elimina registros de una tabla uno por uno

"TRUNCATE" elimina todos los registros de una tabla de una sola vez

1. "Table" se refiere a una estructura de datos que contiene información organizada en filas y columnas.

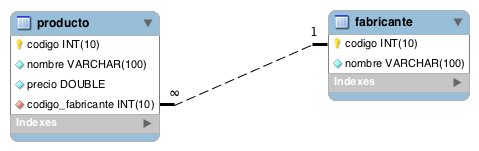
"field" se refiere a una columna individual en una tabla.

1. “JOIN” Permite unir filas de una tabla con filas de otra tabla utilizando una clave común.
2. “Primary Key” es una columna o conjunto de columnas que identifica de manera única cada registro en una table.

“Unique Key” permite valores únicos en una columna o conjunto de columnas.

1. No, los valores NULL no son iguales a cero. NULL significa "valor desconocido" o "falta de valor" y no se puede comparar con ningún otro valor, ni siquiera con cero.
2. El uso principal de los Stored Procedures es la ejecución de tareas complejas y frecuentes de forma eficiente y segura, ayudando a reducir la cantidad de consulta enviadas al servidor.

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



1. Crear base de datos.

**USE master;**

**GO**

**CREATE DATABASE CETHY\_db;**

**USE CETHY\_db;**

1. Crear tablas Insertar datos en cada tabla

**CREATE TABLE Fabricante (**

**codigo INT(10) PRIMARY KEY,**

**nombre VARCHAR(100)**

**);**

**CREATE TABLE Producto (**

**codigo INT(10) PRIMARY KEY,**

**nombre VARCHAR(100),**

**precio DOUBLE,**

**codigo\_fabricante INT(10),**

**FOREIGN KEY (código\_fabricante) REFERENCES Fabricante(codigo)**

**);**

**INSERT INTO Fabricante (codigo, nombre) VALUES (1, 'Fabricante A');**

**INSERT INTO Fabricante (codigo, nombre) VALUES (2, 'Fabricante B');**

**INSERT INTO Fabricante (codigo, nombre) VALUES (3, 'Fabricante C');**

**INSERT INTO Producto (codigo, nombre, precio, codigo\_fabricante)**

**VALUES (1, 'Producto A', 250.50, 1);**

**INSERT INTO Producto (codigo, nombre, precio, codigo\_fabricante)**

**VALUES (2, 'Producto B', 100.50, 1);**

**INSERT INTO Producto (codigo, nombre, precio, codigo\_fabricante)**

**VALUES (3, 'Producto C', 1000.75, 2);**

**INSERT INTO Producto (codigo, nombre, precio, codigo\_fabricante)**

**VALUES (4, 'Producto D', 120.80, 2);**

**INSERT INTO Producto (codigo, nombre, precio, codigo\_fabricante)**

**VALUES (5, 'Producto E', 600.65, 3);**

1. Listar nombres de todos los productos de la tabla “Productos”

**SELECT nombre FROM Producto;**

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

**SELECT nombre, precio FROM Producto;**

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

**SELECT nombre, ROUND(precio) FROM Producto;**

1. 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 LIMIT 1;**

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

**SELECT p.nombre AS producto, p.precio, f.nombre AS fabricante**

**FROM Producto p**

**INNER JOIN Fabricante f ON p.codigo\_fabricante = f.codigo;**

1. Calcula el numero total de fabricantes que hay en la tabla Fabricante.

**SELECT COUNT(\*) FROM Fabricante;**

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

**SELECT SUM(precio) FROM Producto WHERE codigo\_fabricante = 1;**

1. Crear un “Stored procedure”

**USE CETHY\_db**

**GO**

**CREATE PROCEDURE sp\_productos**

**AS**

**SELECT p.nombre AS producto, p.precio, f.nombre AS fabricante**

**FROM Producto p**

**INNER JOIN Fabricante f ON p.codigo\_fabricante = f.codigo;**

**GO**

LLAMAMOS EL PROCEDIMIENTO

**EXEC sp\_productos ;**

**Prueba practica:**

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

Realice las siguientes tareas:

1. Crear plan de prueba.
2. Diseñe los casos de pruebas
3. Definir alcance de las pruebas
4. Reporte de defecto.
5. Documenta algunos casos de pruebas.

**Repuestas:**

**1. Crear plan de prueba:**

**Requerimientos:**

Verificar el correcto funcionamiento de la página web <https://www.saucedemo.com/inventory.html> y sus funcionalidades.

Las pruebas se realizarán en la página web <https://www.saucedemo.com/inventory.html> en diferentes navegadores web, como en ordenadores de escritorio. Se evaluarán las siguientes funcionalidades:

- Inicio de sesión

- Visualización de productos

- Agregar productos al carrito de compras

- Realizar una compra

- Visualización del historial de compras

Método de prueba: Se realizarán pruebas manuales exploratorias y pruebas automatizadas utilizando Selenium WebDriver y Python.

Criterios de aceptación: Se considerará que la prueba ha sido exitosa si todas las funcionalidades se han probado sin errores y el comportamiento de la página web se mantiene estable y predecible durante la prueba.

2**. Diseño de casos de prueba**:

**Caso de prueba 1** - Inicio de sesión:

1. Ingresar a la página web <https://www.saucedemo.com/inventory.html>

2. Ingresar el nombre de usuario "standard\_user"

3. Ingresar la contraseña "secret\_sauce"

4. Presionar el botón "Login"

5. Verificar que se muestra la página de inventario

**Caso de prueba 2** - Visualización de productos:

1. Ingresar a la página web <https://www.saucedemo.com/inventory.html>

2. Verificar que se muestran los productos disponibles en la página

3. Verificar que cada producto muestra su nombre, descripción y precio

**Caso de prueba 3** - Agregar productos al carrito de compras:

1. Ingresar a la página web <https://www.saucedemo.com/inventory.html>

2. Hacer clic en el botón "Add to cart" de un producto

3. Verificar que el producto se ha agregado correctamente al carrito de compras

4. Repetir los pasos 2 y 3 para otro producto

5. Verificar que ambos productos se muestran en el carrito de compras

**Caso de prueba 4** - Realizar una compra:

1. Ingresar a la página web <https://www.saucedemo.com/inventory.html>

2. Hacer clic en el botón "Add to cart" de un producto

3. Hacer clic en el botón "Checkout"

4. Ingresar la información de envío y de pago

5. Hacer clic en el botón "Continue"

6. Verificar que se muestra la página de confirmación de la compra

**Caso de prueba 5** - Visualización del historial de compras:

1. Ingresar a la página web <https://www.saucedemo.com/inventory.html>

2. Hacer clic en el botón del carrito de compras

3. Hacer clic en el botón "Checkout"

4. Ingresar la información de envío y de pago

5. Hacer clic en el botón "Continue"

6. Hacer clic en el botón "Finish"

7. Ingresar a la página web <https://www.saucedemo.com/inventory.html>

8. Hacer clic en el botón del historial de compras

9. Verificar que se muestra la lista de compras realizadas

3. **Alcance de las pruebas:**

El alcance de las pruebas incluirá la evaluación de los siguientes aspectos de la página <https://www.saucedemo.com/inventory.html>. Se verificará que los usuarios puedan:

* Iniciar sesión en la página.
* Navegar por la página y buscar productos por categoría o nombre.
* Añadir productos al carrito de compra y verificar que se actualice correctamente.
* Acceder al carrito de compra y proceder con la compra de los productos seleccionados.
* Verificar que la información de los productos y precios se muestren correctamente.
* Verificación de que los usuarios pueden realizar el pago y finalizar la compra sin problemas.
* Verificación de que los mensajes de error se muestran correctamente cuando hay algún problema en la página.

Se probará la funcionalidad en diferentes navegadores, dispositivos y sistemas operativos, para garantizar que el sitio web sea accesible para todos los usuarios.

4. **Reporte de defecto:**

En caso de encontrar un defecto durante las pruebas, se informará de la siguiente manera:

* Se describirá el problema de manera clara y detallada.
* Indicar el paso a paso para reproducir el error
* Adjuntar capturas de pantalla o videos que ayuden a visualizar el problema.
* Asignar una prioridad al defecto según su impacto en la funcionalidad del sitio.
* Informar al equipo de desarrollo para que puedan solucionar el problema.

5. **Documentación de algunos casos de prueba:**

A continuación, se presentan algunos casos de prueba que se pueden utilizar en la evaluación del sitio:

1. **Verificar que se pueda iniciar sesión correctamente en el sitio**:

* Paso 1: Acceder a la página de [https://www.saucedemo.com](https://www.saucedemo.com/).
* Paso 2: Ingresar un usuario y contraseña válidos.
* Paso 3: Verificar que se redireccione correctamente a la página de inventario.

1. **Añadir un producto al carrito de compra:**

* Paso 1: Seleccionar un producto de la página de inventario.
* Paso 2: Verificar que se muestre la información del producto (nombre, precio, descripción, imagen).
* Paso 3: Hacer clic en el botón "Agregar al carrito".
* Paso 4: Verificar que se muestre una notificación indicando que el producto se agregó al carrito.
* Paso 5: Acceder al carrito de compra y verificar que el producto seleccionado se muestre correctamente.

1. **Realizar una compra:**

* Paso 1: Añadir al menos dos productos al carrito de compra.
* Paso 2: Acceder al carrito de compra y verificar que se muestren todos los productos seleccionados.
* Paso 3: Hacer clic en el botón "Checkout".
* Paso 4: Completar los campos de información de envío y facturación.
* Paso 5: Seleccionar un método de pago válido.
* Paso 6: Hacer clic en el botón "Finalizar compra".
* Paso 7: Verificar que se muestre una notificación indicando que la compra se realizó correctamente.