PRUEBA TÉCNICA - FUNCIONAL PARA EL ROL QA ENGINEER

Pruebas Manuales/Funcionales:

1. ¿Qué entiendes por el término pruebas funcionales?

las pruebas funcionales se basan en la validar el funcionamiento de la aplicación en base a los requisitos del clientes.

1. ¿Cuáles son los pasos importantes que se deben cubrir en las pruebas funcionales?

* **Revisión de Requisitos**: Requisitos funcionales (Criterios de aceptación en HUs).
* **Creación de Casos de Prueba**.
* **Ejecución de Casos de Prueba**:Manual o automatizado
* **Registro de Resultados**: Documentar los resultados de cada prueba,
* **Reporte de Defectos**.
* **Re-testeo**: Probar de nuevo despues de corregido el defecto .
* **Pruebas de regresión** : Evaluar la cobertura de pruebas y concluir la fase de pruebas.

1. ¿Cuál es la diferencia entre pruebas funcionales y no funcionales?

Las pruebas funcionales se aseguran que la aplicación cumpla con los requisitos funcionales(lo debe de hacer la aplicación) del cliente mientras que las pruebas no funcionales se aseguran de que el software haga lo que se supone que debe hacer de manera eficiente, segura y efectiva bajo diversas condiciones.

1. Define en tus propias palabras qué es un release.

Un relesea es la entre en producción de un MVP (Mínimo producto viable)

1. Escribe una lista de los posibles estados que puede tener un defecto (escribe una pequeña definición de cada uno).

* **Nuevo**: se registra por primera vez el defecto.
* **Asignado**: El defecto ha sido asignado a un desarrollador o equipo para su solución.
* **En Progreso**: Se esta trabajando en la solución del defecto.
* **Resuelto(QA Test)**: El defecto ha sido corregido por el desarrollador y pendiente de ser verificado.
* **Verificado(Done)**: El defecto ha sido revisado por el equipo de QA.
* **Cerrado**: El defecto ha sido solucionado satisfactoriamente.
* **Reabierto**: El defecto, que se creía resuelto, pero el bug se evidencia de nuevo.
* **Rechazado**: El defecto no es valido según los artefactos funcionales.

1. ¿Cuáles puntos consideras importantes a la hora de escribir un caso de prueba?

* **Título**: descripción corta de lo que se va a probar .
* **Descripción**: Explicación de la prueba.
* **Precondiciones**: Estado inicial de la prueba.
* **Pasos**: Detalle del paso a paso de las acciones a realizar.
* **Datos de Prueba**: Data requerida para le ejecución de la prueba
* **Resultado Esperado**: ya sea un caso éxito o alterno.

1. ¿Cuáles puntos consideras importantes a la hora de documentar un defecto?

* **Identificador**: Un código único para referenciar el defecto.
* **Título**: Descripción breve del defecto.
* **Descripción**: Explicación detallada del defecto.
* **Pasos para Reproducir**: Instrucciones para replicar el defecto.
* **Datos de Prueba**: Data con la que se pueda reproducir el defecto.
* **Resultado Actual**: comportamiento de la funcionalidad actual.
* **Resultado Esperado**: Comportamiento correcto esperado de la funcionalidad.
* **Severidad y Prioridad**: Impacto del defecto y urgencia de su resolución.
* **Entorno de Pruebas**: Ambiente en el cual se puede replicar el defecto.
* **Evidencia**: Capturas de pantalla, logs, etc..

1. ¿Qué diferencias hay entre probar la respuesta de un API y el esquema de respuesta de un API?

La diferencia es que al probar la respuesta del API, se valida la información de los campos entregados y al probar el esquema del API es la estructura de la respuesta, los tipos de campo según los requisitos funcionales.

**Análisis y Documentación de Pruebas:**

**En los casos de pruebas planteados se determino como artefacto principal los criterios de aceptación, el objetivo de las pruebas funcionales para el historial de pedidos en la aplicación de comercio electrónico es garantizar que los usuarios registrados puedan acceder y visualizar correctamente sus pedidos anteriores. Las pruebas cubrirán diversos escenarios que incluyen la visualización de la opción en el menú de navegación, la lista de pedidos, los detalles de un pedido específico, y la gestión de casos especiales como pedidos vacíos, información incompleta y como consideración especial, la respuesta de valores limites como eran expuestas al cliente tanto la cantidad máximo de pedidos o de productos.**

**Automation test:**

Para resolver y poder ejecutar los casos de pruebas, se modificó el tiempo de espera de carga de la pagina que estaba en 0,5 pero se cambio a 30.

Se debió actualizar el webdriver-manager==4.0.1 para tomar la ultima actualización del Chromediver.

y se modifico algunos locators para que la prueba corriera.

Se integro la ejecución con paralelismo para mejorar los tiempos de ejecución y para documentar el resultado de las pruebas se integraron los servicios de QASE.IO

Para ejecutar la prueba con ambiente virtual, ejecutar en consola

python3 -m venv venv;   
 source venv/bin/activate;  
 pip install -r requirements.txt;

Para correr las pruebas de manera paralela:

python behave\_parallel.py -t regression

Nota: se debe especificar la el tag a ejecutar (-t regression)

Secuencial test:

behave --no-capture --format plain --tags={@tag} -Ddriver=chrome

Credenciales qase:

<https://app.qase.io/run/DEMO/dashboard/1>

usuario: [simetrick@yopmail.com](mailto:simetrick@yopmail.com)

password: Mouse312#312