**Respuesta Prueba QA**

**Ejercicio 1**

Con la información Proporcionada no es posible realizar un plan de prueba que sea completo.

Para realizar un plan de pruebas que sea completo yo sugiero lo siguiente

* Realizar reuniones con el equipo de desarrollo o los propietarios del producto para obtener más información acerca de como se planea mejorar el proceso de inicio de sesión
* Cuáles son los datos de entrada y salida esperados.
* Los casos de uso de los usuarios.
* Cuáles son los posibles escenarios de error potenciales.

En base a la información obtenida yo sugeriría lo siguiente:

* **Objetivo:** Probar la funcionalidad básica del nuevo proceso de inicio de sesión.
* **Alcance:** El plan de pruebas inicial se centrará en los requisitos funcionales más básicos del nuevo proceso de inicio de sesión.
* **Estrategia**: El plan de pruebas inicial utilizará un enfoque de prueba de caja negra.
* **Casos de prueba:** El plan de pruebas inicial incluirá los siguientes casos de prueba:
  + Caso de prueba 1: Probar la capacidad de iniciar sesión con un usuario y una contraseña válidos.
  + Caso de prueba 2: Probar la capacidad de iniciar sesión con un usuario y una contraseña no válidos.
  + Caso de prueba 3: Probar la capacidad de iniciar sesión con un usuario que no existe.

**Sección B**

Si, eh trabajado con la metodología Agile y los artefactos y ceremonias contribuyen el día a día a mejorar la calidad de un producto son las siguientes:

**Daily Standup** (Reunión diaria): Los miembros del equipo informan sobre su progreso, planes y obstáculos. Esto promueve la comunicación, colaboración y la detección temprana de problemas, mejorando así la calidad del trabajo.

**Sprint Review** (Revisión de sprint): Al final de cada etapa, se analiza el trabajo hecho y se escucha la opinión del cliente o equipo de negocios. Esto permite hacer ajustes y mejorar constantemente el producto.

**Sprint Planning** (Planificación de sprint): Durante esta ceremonia, se planifican las tareas y se establecen los objetivos para el próximo sprint. Esto asegura que las características críticas se aborden y que se dedique tiempo suficiente a las pruebas.

**Definition of Done** (Definición de hecho): Este artefacto define los criterios que una historia de usuario debe cumplir antes de considerarse completa. Esto garantiza que las pruebas se realicen y que se cumplan los estándares de calidad.

**Ejercicio 2**

Feature: Validación de JSON Request

Scenario: JSON Request válido

Given un JSON request válido

When envío la solicitud a la API

Then la API debe responder con un código de estado 200

Scenario: JSON Request con fecha de vencimiento inválida

Given un JSON request con una fecha de vencimiento inválida

When envío la solicitud a la API

Then la API debe responder con un mensaje de error relacionado con la fecha de vencimiento

Scenario: JSON Request sin un campo obligatorio

Given un JSON request sin uno de los campos obligatorios

When envío la solicitud a la API

Then la API debe responder con un mensaje de error indicando el campo faltante

Scenario: JSON Request con tipo de tarjeta no válido

Given un JSON request con un tipo de tarjeta no válido

When envío la solicitud a la API

Then la API debe responder con un mensaje de error relacionado con el tipo de tarjeta

Scenario: JSON Request con monto negativo

Given un JSON request con un monto negativo

When envío la solicitud a la API

Then la API debe responder con un mensaje de error relacionado con el monto

Scenario: JSON Request con ID de tienda inexistente

Given un JSON request con un ID de tienda inexistente

When envío la solicitud a la API

Then la API debe responder con un mensaje de error relacionado con el ID de tienda

**Ejercicio 3**

Feature: Buscar un producto

Scenario: El usuario busca un producto

Given el usuario está en la página de inicio de Amazon

When el usuario ingresa el nombre de un producto en el campo de búsqueda

And el usuario hace clic en el botón de búsqueda

Then los resultados de la búsqueda se muestran en la página

Feature: Agregar un producto al carrito de compras

Scenario: El usuario agrega un producto al carrito de compras

Given el usuario está en la página de detalles de un producto

When el usuario hace clic en el botón "Agregar al carrito"

Then el producto se agrega al carrito de compras

Feature: Checkout y realizar una compra

Scenario: El usuario realiza una compra

Given el usuario tiene productos en su carrito de compras

When el usuario hace clic en el botón "Checkout"

And el usuario ingresa su información de envío y pago

And el usuario hace clic en el botón "Realizar pedido"

Then el pedido se realiza

**Ejercicio 4**

Este es un diagrama que muestra el proceso de bugtracking:

* Se identifica y luego se reporta bugs
* Asignar bugs a desarrolladores
* Los desarrolladores resuelven bugs
* El QA prueba bugs resueltos
* Bugs se cierran

Para mantener a mi equipo informado sobre los bugs, utilizábamos la herramienta Jira para registrarlos. En cada reporte, incluíamos detalles como la descripción del bug, los pasos para reproducirlo, una captura de pantalla o un video del problema, así como su nivel de prioridad y criticidad, que podía ser alta, media o baja en este caso.

Con respecto a los Bugs Bloqueantes, en una ocasión, un cliente deseaba efectuar una compra utilizando la tarjeta de crédito como medio de pago. Sin embargo, al intentar finalizar la transacción seleccionando esta opción, no fue posible completarla. En su lugar, se requirió el uso de una Terminal Transbank, destinada a resolver problemas relacionados con los medios de pago.

Se siguieron los siguientes pasos para abordar la situación:

En primer lugar, se notificó al cliente acerca del inconveniente y se proporcionaron las instrucciones necesarias.

Posteriormente, al cierre del local, se implementó una actualización que corrigió el bug que impedía la realización de la transacción con tarjeta de crédito.