

# PRUEBAS DE CAJA NEGRA BLANCA

**Sistema de Gestión de Clientes y Productos de Elena  
(SGCP-E)**

Jesús Huerta Aguilar  
Javier de la Luz Ruiz  
Ernesto Flores Cesareo



## CONTENIDO

<b>PRUEBAS DE CAJA NEGRA .....</b>	<b>3</b>
<b>CAJA NEGRA 1: REGISTRAR PRODUCTO.....</b>	<b>4</b>
<b>CASO DE PRUEBA.....</b>	<b>4</b>
<b>EJECUTAR CASO DE PRUEBA.....</b>	<b>7</b>
<b>COMPARAR RESULTADOS .....</b>	<b>8</b>
<b>REGISTRAR RESULTADOS .....</b>	<b>8</b>
<b>CAJA NEGRA 2: REGISTRAR CLIENTE .....</b>	<b>10</b>
<b>CASO DE PRUEBA.....</b>	<b>10</b>
<b>EJECUTAR CASO DE PRUEBA.....</b>	<b>14</b>
<b>COMPARAR RESULTADOS .....</b>	<b>14</b>
<b>REGISTRAR RESULTADOS .....</b>	<b>15</b>
<b>PRUEBAS DE CAJA BLANCA .....</b>	<b>16</b>
<b>Asd.....</b>	<b>16</b>

## PRUEBAS DE CAJA NEGRA

Las pruebas de caja negra son un método de pruebas de software en el cual la funcionalidad interna del sistema bajo prueba no se considera ni se examina. En lugar de ello, se centra en verificar cómo funciona el software al proporcionarle diferentes entradas y comprobar si las salidas son las esperadas.

El término "caja negra" se utiliza porque para estas pruebas el sistema es como una caja negra; no se puede ver lo que está sucediendo en su interior. Se examina desde una perspectiva externa, verificando el comportamiento y los resultados en lugar de los procesos internos.

En las pruebas de caja negra, los requisitos funcionales son de suma importancia, ya que se basan en ellos para diseñar los casos de prueba.

Las pruebas de caja negra tienen las siguientes características principales:

1. **Independencia del código:** Estas pruebas no requieren ningún conocimiento del código subyacente. Los casos de prueba se diseñan desde la perspectiva del usuario y de los requisitos funcionales, no desde la perspectiva del desarrollador.
2. **Se centra en los datos de entrada y salida:** El principal enfoque de las pruebas de caja negra es verificar si para un conjunto de entradas, la salida es la esperada. Se presta menos atención a cómo el sistema procesa la entrada para producir la salida.
3. **Usabilidad:** Las pruebas de caja negra pueden ayudar a identificar problemas de usabilidad, ya que se realizan desde la perspectiva del usuario.
4. **Pruebas a nivel de sistema:** Este tipo de pruebas se realiza a menudo a nivel de sistema, pero también se puede utilizar en niveles de integración y de unidad.

Los tipos de pruebas de caja negra incluyen:

1. **Pruebas Funcionales:** Verifica si el software cumple con los requisitos funcionales especificados.
2. **Pruebas de Regresión:** Verifica si las modificaciones en el código han introducido nuevos errores.
3. **Pruebas de Usabilidad:** Se centra en la facilidad de uso del software.
4. **Pruebas de Compatibilidad:** Verifica si el software puede funcionar en diferentes entornos, como diferentes sistemas operativos o navegadores.
5. **Pruebas de Rendimiento:** Verifica la eficacia del software bajo carga, como la velocidad de respuesta y la utilización de recursos.

Las pruebas de caja negra son un componente vital en el proceso de pruebas de software, ya que ayudan a asegurar que el software funcione correctamente desde la perspectiva del usuario, sin importar la complejidad del código subyacente.

## CAJA NEGRA 1: REGISTRAR PRODUCTO

### CASO DE PRUEBA

- Descripción del caso de prueba:

El caso de uso "Registrar Producto" es una función crítica que permite al administrador registrar un nuevo producto en la base de datos del sistema. Esta funcionalidad implica una serie de pasos, como ingresar información del producto, verificar los campos ingresados y finalmente, guardar la información en la base de datos.

En nuestras pruebas de caja negra, nos enfocaremos en asegurar que este proceso funcione de manera correcta y eficiente. Al ser una prueba de caja negra, no consideraremos los detalles de cómo se implementa la funcionalidad en el código, sino que nos centraremos en los resultados que produce el sistema al proporcionarle diferentes entradas.

Evaluaremos casos como los flujos normal y alternativo. En el flujo normal, todas las condiciones son ideales y el administrador proporciona todos los detalles necesarios para registrar un producto. En el flujo alternativo, se presentan situaciones donde el administrador puede dejar algún campo vacío, lo que debería desencadenar un mensaje de error por parte del sistema.

Para cada uno de estos casos de prueba, compararemos los resultados obtenidos con los resultados esperados, según lo definido en el caso de uso. Si hay discrepancias, se reportarán como fallas y se realizarán las correcciones necesarias en el sistema.

Con esta metodología, esperamos asegurar que la funcionalidad de "Registrar Producto" cumple con las expectativas y que proporciona una experiencia de usuario consistente y sin errores para el administrador.

- Requisito Funcional Asociado:

El requisito funcional vinculado a este caso de prueba es "Registrar Producto". Esta característica es una parte esencial de la funcionalidad general del sistema, ya que permite al administrador introducir nuevos productos en la base de datos, facilitando así la gestión eficiente de la tienda en línea y la disponibilidad de la gama completa de productos para los clientes.

Este requisito funcional, al ser fundamental en la operación cotidiana del sistema, será sometido a pruebas rigurosas con el objetivo de detectar y rectificar cualquier posible fallo, limitación o inconveniente. Este proceso incluirá la exploración de escenarios diversos y límites de operación para asegurar que el sistema es robusto y está listo para enfrentarse a situaciones reales de uso.

En estas pruebas, examinaremos múltiples aspectos como la facilidad de uso, el manejo de errores, la capacidad de procesar la entrada de datos y la eficacia en el

almacenamiento y visualización de la información del producto. El objetivo es asegurar que el sistema no solo cumpla con las expectativas de funcionamiento sino que también ofrezca una experiencia de usuario satisfactoria.

Además, estos exámenes nos permitirán evaluar el desempeño del sistema en escenarios de carga alta y baja, identificar y corregir problemas de usabilidad, y garantizar que el sistema es intuitivo y fácil de usar. Al final, las pruebas nos ayudarán a asegurar que el sistema se comporte según lo esperado, sea resistente a los errores y cumpla de manera efectiva con su propósito de "Registrar Producto".

- Entradas y Condiciones Previas:

Como precondiciones para esta prueba, es necesario que el administrador haya iniciado sesión en el sistema y que este último tenga acceso a la base de datos de productos. Estas condiciones garantizan que el administrador tiene los derechos y capacidades para llevar a cabo la tarea de registrar un producto y que el sistema está en condiciones de procesar y almacenar correctamente esta información.

Entrada de datos: Para este caso de prueba específico, intentaremos ingresar un dato no compatible en el campo DURACIÓN, que está diseñado para almacenar valores numéricos de tipo INT (entero). Esta situación puede ocurrir en situaciones reales cuando un usuario comete un error al ingresar la información. Al tratar de ingresar un valor de texto "cincuenta" en lugar de un valor numérico, podemos evaluar cómo el sistema maneja este tipo de error.

Datos de Entrada:

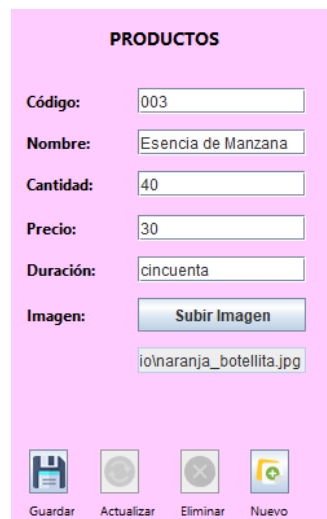
- A. Código: 003
- B. Nombre: Esencia de Manzana
- C. Cantidad: 40
- D. Precio: 30
- E. Duración: "cincuenta " (Nota: El precio debe ser un INT. En esta prueba ingresaremos un String.)
- F. Imagen: C:/ruta/de/la/imagen.png

- Pasos para Ejecutar el Caso de Prueba:

Para llevar a cabo el caso de prueba, se deben seguir estos pasos detallados:

1. Asegúrate de que se cumplen las precondiciones: el administrador ha iniciado sesión en el sistema y el sistema tiene acceso a la base de datos de productos.
2. Navega a la opción de "Productos" en el sistema, que es donde se puede añadir un nuevo producto.

3. El sistema mostrará una pantalla con los campos para ingresar la información del producto.
4. Comienza a llenar los campos con los datos proporcionados, que son: Código: 003, Nombre: Esencia de Manzana, Cantidad: 40, Precio: 30, Imagen: C:/ruta/de/la/imagen.png.
5. Cuando llegues al campo "Duración", intenta ingresar el string "cincuenta" en lugar del valor numérico esperado.
6. Tras ingresar el string en el campo "Duración", intenta continuar con el flujo normal del proceso. Esto incluirá intentar guardar la información pulsando el botón correspondiente.



The screenshot shows a web form titled 'PRODUCTOS' with the following fields and values:

- Código:** 003
- Nombre:** Esencia de Manzana
- Cantidad:** 40
- Precio:** 30
- Duración:** cincuenta
- Imagen:** Subir Imagen (button), ioInaranja\_botellita.jpg (text)

At the bottom, there are four icons with labels: Guardar (floppy disk), Actualizar (refresh), Eliminar (trash), and Nuevo (plus sign).

Ilustración 1: Campos llenados con las entradas de prueba

Al seguir estos pasos, estamos creando una situación en la que el sistema debe manejar una entrada de datos no válida (un string en lugar de un int). La forma en que el sistema maneje esta situación nos permitirá evaluar su robustez y su capacidad para manejar errores de manera eficaz, brindando al mismo tiempo una buena experiencia al usuario.

- Resultados Esperados:

En específico, esperamos que el sistema muestre una notificación o mensaje de error claro al usuario, indicándole que el tipo de entrada proporcionado en la sección de "Duración" no es correcto. Este mensaje de error debería ser lo suficientemente explícito para que el usuario entienda el error cometido y cómo corregirlo.

Además, el sistema debe evitar que la entrada incorrecta llegue a la base de datos. El manejo de errores efectivo es crítico para mantener la integridad de las tablas SQL y para prevenir problemas de interpretación o procesamiento de los datos en el futuro.

Finalmente, esperamos que el sistema permita al usuario corregir el error e intentar registrar el producto nuevamente. Esto debería suceder sin tener que reiniciar todo el proceso de registro, lo que podría causar frustración en el usuario.

En resumen, nuestros resultados esperados son: la detección y prevención de la entrada de datos incorrecta, la notificación clara y útil al usuario sobre el error, la protección de la integridad de la base de datos, y la oportunidad de corregir el error y reintentar el registro del producto sin inconvenientes innecesarios.

### EJECUTAR CASO DE PRUEBA

- Resultados Obtenidos:

Al ejecutar el caso de prueba e intentar registrar el producto con un string en el campo "Duración", se observó que el sistema no logró manejar la entrada de datos incorrecta de manera adecuada. Al presionar el botón "Registrar", el sistema se detuvo y se produjo un error, que se mostró en la consola con el mensaje:

**'Exception in thread "AWT-EventQueue-0"**  
**java.lang.NumberFormatException: For input string: "cincuenta"'**

Este error indica que el sistema intentó convertir el string "cincuenta" a un número (como se esperaba en el campo "Duración"), pero falló debido a que "cincuenta" no es una representación válida de un número en el formato esperado.

Además, se constató que el producto no se registró en la base de datos, lo cual es positivo ya que evitó la entrada de información incorrecta. Sin embargo, la respuesta del sistema ante este error no fue la más óptima, ya que el usuario tuvo que cerrar y volver a abrir el programa para solucionar el problema, lo que no es una experiencia de usuario deseable.

Por lo tanto, a pesar de que el sistema logró evitar el registro incorrecto del producto, los resultados obtenidos demuestran que el manejo de errores del sistema necesita mejoras para ofrecer una mejor experiencia al usuario y para manejar eficazmente las situaciones en las que se proporciona una entrada de datos incorrecta o no válida.

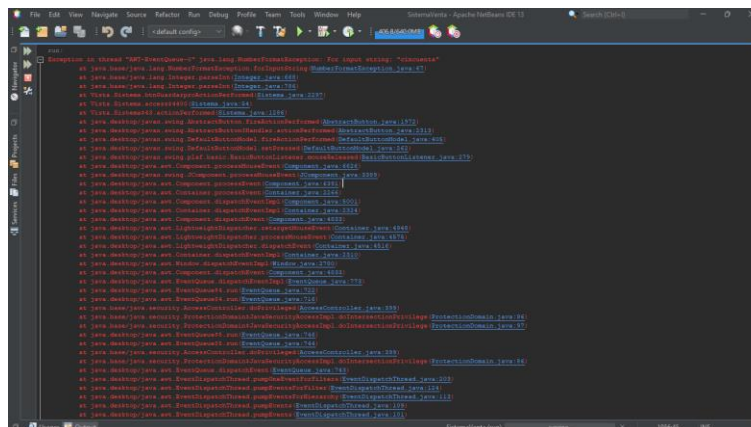


Ilustración 2: ejecución de prueba

### COMPARAR RESULTADOS

- Estado (aprobado / fallido):

### FALLIDO

La razón para determinar el estado como "Fallido" se debe a que, aunque el sistema logró evitar el registro incorrecto del producto en la base de datos (lo cual es correcto), el manejo de errores no cumplió con las expectativas establecidas. En lugar de informar al usuario de manera clara y precisa sobre el tipo de error y cómo corregirlo, el sistema simplemente se detuvo y mostró un mensaje de error técnico en la consola.

Además, para resolver el problema, fue necesario reiniciar el programa, lo cual puede ser molesto e ineficiente para el usuario. Este tipo de manejo de errores puede generar frustración y una mala experiencia del usuario, además de ser una mala práctica en términos de diseño y desarrollo de software.

Por lo tanto, aunque parte del sistema funcionó correctamente (es decir, no permitió que la entrada incorrecta se almacenara en la base de datos), el caso de prueba se considera fallido debido a su manejo deficiente del error de entrada de datos.

### REGISTRAR RESULTADOS

Elemento del Caso de Prueba	Detalle
Descripción del Caso de Prueba	Verificar cómo el sistema maneja la entrada incorrecta (String en lugar de INT) en el campo de precio durante el registro de un producto.
Requisito Funcional Asociado	Registrar Producto
Entradas y Condiciones Previas	A. Código: 003 B. Nombre: Esencia de Manzana C. Cantidad: 40 D. Precio: 30 E. Duración: "cincuenta" F. Imagen: C:/ruta/de/la/imagen.png
Pasos para Ejecutar el Caso de Prueba	Se siguieron los pasos indicados para ingresar los datos y se intentó registrar el producto.
Resultados Esperados	Se esperaba que el sistema mostrara un mensaje de error claro al usuario y evitara el registro del producto hasta que se corrigiera el error.



<b>Resultados Obtenidos</b>	El sistema se detuvo, mostró un mensaje de error técnico en la consola, y no permitió el registro del producto. Fue necesario reiniciar el programa para solucionar el problema.
<b>Estado</b>	Fallido
<b>Observaciones Adicionales</b>	El sistema logró evitar el registro incorrecto, pero el manejo del error no fue el adecuado. Se recomienda mejorar el manejo de errores y la retroalimentación al usuario en situaciones de este tipo.

## CAJA NEGRA 2: REGISTRAR CLIENTE

### CASO DE PRUEBA

- Descripción del caso de prueba:

Las pruebas de caja negra son un método esencial para validar si un sistema de software está funcionando según las expectativas y se alinea con los requerimientos especificados. En nuestro caso de estudio, analizaremos el proceso de "Registrar Cliente" en un sistema de aromaterapia. Nuestro enfoque de prueba de caja negra nos permitirá confirmar si el sistema está cumpliendo con las funciones y comportamientos descritos en este caso de uso.

El caso de uso "Registrar Cliente" describe el proceso por el cual un administrador registra un nuevo usuario en el sistema. Como pruebas de caja negra, examinaremos la funcionalidad de este proceso desde la perspectiva del administrador, quien es el usuario final en este escenario.

Comenzaremos identificando las entradas y salidas de cada paso en el flujo normal y alternativo del caso de uso, y luego desarrollaremos casos de prueba correspondientes. Por ejemplo, consideraremos qué sucede cuando el administrador introduce los datos correctos del cliente (flujo normal) y qué sucede cuando deja algún campo vacío (flujo alternativo). Aseguraremos que el sistema proporcione las respuestas adecuadas y esperadas en cada situación.

Nuestro objetivo es asegurar que el sistema se comporte correctamente de acuerdo a las precondiciones, postcondiciones y a cada paso del flujo de trabajo, y que pueda manejar tanto los casos exitosos como los excepcionales o de error. Como resultado de nuestras pruebas, esperamos confirmar que el sistema está cumpliendo con sus requerimientos funcionales y que proporciona una experiencia de usuario efectiva y libre de errores.

- Requisito Funcional Asociado:

El requisito funcional que se someterá a revisión a través del enfoque de pruebas de caja negra es "Registrar Cliente". Este requisito es un componente crítico del sistema de aromaterapia, ya que facilita la incorporación de nuevos usuarios al sistema. Su correcta implementación es fundamental para el funcionamiento general del sistema y la satisfacción del usuario, en este caso, el administrador.

Nuestro objetivo al realizar las pruebas en "Registrar Cliente" es identificar y rectificar cualquier problema potencial que pueda surgir durante su uso. Esto incluirá la búsqueda y resolución de errores, la identificación y eliminación de limitaciones innecesarias, la mejora de la usabilidad y la eficiencia, y la confirmación de que todas las funciones se realizan correctamente y según lo especificado.

Al seguir este enfoque, podremos garantizar que el sistema de aromaterapia es robusto, fiable y fácil de usar, cumpliendo con todas las expectativas y necesidades de los administradores en el proceso de registro de nuevos clientes.

De esta manera, el requisito funcional "Registrar Cliente" se someterá a un riguroso proceso de prueba y validación, asegurando su correcto funcionamiento en todos los escenarios posibles. Este esfuerzo mejorará la calidad del software, reducirá el riesgo de fallas en el sistema y proporcionará una mejor experiencia para los administradores que utilizan el sistema.

- Entradas y Condiciones Previas:

Las pruebas de software requieren un escenario específico y condiciones establecidas previamente para garantizar su efectividad. En nuestro caso de prueba de "Registrar Cliente", debemos asegurarnos de que se cumplan las siguientes condiciones previas:

1. **Inicio de sesión del administrador:** Antes de poder registrar a un nuevo cliente, el administrador debe haber iniciado sesión en el sistema de aromaterapia. Esta es una condición esencial, ya que solo un administrador autenticado tiene los privilegios necesarios para acceder a la funcionalidad de registro de clientes. El inicio de sesión exitoso del administrador será un prerrequisito clave para nuestro caso de prueba.
2. **Acceso a la base de datos:** Para poder registrar un nuevo cliente y guardar esa información para uso futuro, el sistema de aromaterapia debe tener acceso a su base de datos. Esto significa que la base de datos debe estar en línea y accesible, y el sistema debe tener los permisos correctos para crear nuevas entradas.

Una vez que se han establecido estas condiciones previas, podemos proceder con nuestro caso de prueba. Este caso de prueba tiene un enfoque específico: verificar la respuesta del sistema cuando solo se llena un campo de entrada y se dejan vacíos los demás. A continuación se detalla la entrada de datos:

- A. NOMBRE: Pablo
- B. TELEFONO:
- C. DIRECCION:
- D. CORREO:
- E. CUMPLEAÑOS:
- F. HOBBIES:
- G. MOLESTIAS:

El objetivo de esta prueba es validar cómo el sistema maneja los formularios incompletos durante el proceso de registro de clientes. Se espera que el sistema verifique la validez de la entrada y no permita la creación de un nuevo cliente a menos que todos los campos necesarios estén completos.



*Ilustración 3: Campos llenados con las entradas de prueba*

- Pasos para Ejecutar el Caso de Prueba:

Los pasos a seguir para ejecutar este caso de prueba están diseñados para validar cómo el sistema maneja las entradas incompletas en el formulario de registro de clientes. De acuerdo con el flujo normal del caso de uso "Registrar Cliente", los pasos para esta prueba son los siguientes:

1. **Iniciar sesión:** El administrador inicia sesión en el sistema de aromaterapia con sus credenciales válidas. Este paso garantiza que tenemos un usuario con los derechos adecuados para acceder a la funcionalidad de registro de clientes.
2. **Navegación al formulario de registro de clientes:** Una vez que el administrador ha iniciado sesión correctamente, navega hasta la sección "Cliente" del sistema. Aquí, se le presentará la opción de registrar un nuevo cliente.
3. **Ingresar datos del cliente:** En este paso, el administrador debe llenar el formulario de registro de clientes con los datos del nuevo cliente. Para esta prueba específica, el administrador solo proporcionará un valor para el campo "NOMBRE", dejando todos los demás campos vacíos. En este caso, el nombre ingresado será "Pablo".

Al realizar estos pasos, estamos imitando una situación en la que el administrador no ha proporcionado toda la información necesaria para completar el registro de un nuevo cliente. El objetivo es confirmar que el sistema valida correctamente las

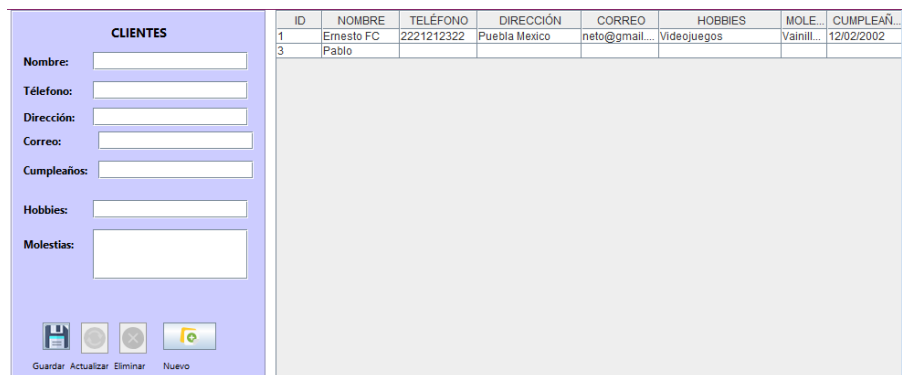
entradas y solicita al administrador que complete todos los campos obligatorios antes de proceder con el registro.

- Resultados Esperados:

En nuestro caso de prueba, donde el administrador ha llenado solo el campo "NOMBRE" y ha dejado todos los demás campos vacíos, nuestros resultados esperados son los siguientes:

- Validación del formulario:** El sistema debe validar el formulario de registro de clientes para garantizar que todos los campos obligatorios estén completos. Dado que solo hemos proporcionado un valor para el campo "NOMBRE", esperamos que el sistema identifique que faltan datos en los otros campos.
- Notificación al usuario:** Como resultado de la validación del formulario, el sistema debe notificar al administrador de que hay campos vacíos que deben llenarse. Esta notificación debería ser clara y específica sobre qué campos necesitan completarse.
- Prevención de registro incorrecto:** El sistema debe evitar que se registre un nuevo cliente con datos incompletos. Esto significa que, a pesar de que el administrador ha intentado registrar un cliente con solo el nombre proporcionado, no debería haber ninguna nueva entrada en la base de datos para un cliente llamado "Pablo".
- Invitación para corregir errores:** Después de notificar al administrador sobre los campos vacíos, el sistema debería invitar al administrador a corregir estos errores ingresando los datos faltantes. Este paso asegura que los datos recopilados para el registro de clientes sean completos y precisos.

Estos resultados esperados garantizan que el sistema mantiene la integridad de los datos y proporciona una experiencia de usuario efectiva, previniendo la entrada de información incorrecta en las tablas SQL y orientando al administrador a través del proceso de registro de clientes.



ID	NOMBRE	TELÉFONO	DIRECCIÓN	CORREO	HOBBIES	MOLE.	CUMPLEAÑOS
1	Ernesto FC	2221212322	Puebla Mexico	neto@gmail...	Videojuegos	Vainill...	12/02/2002
3	Pablo						

Ilustración 4: Tabla con apartado incompleto

### EJECUTAR CASO DE PRUEBA

- Resultados Obtenidos:

Los resultados obtenidos durante la ejecución del caso de prueba proporcionan información vital sobre el comportamiento real del sistema en respuesta a una serie de acciones definidas. En este caso, al seguir los pasos descritos previamente, los resultados obtenidos fueron los siguientes:

1. **Ejecución exitosa del sistema:** A pesar de que solo se proporcionó el nombre del cliente y se dejaron todos los demás campos vacíos, el sistema ejecutó el registro del nuevo cliente sin problemas. Esto sugiere que la validación del formulario no funcionó como se esperaba.
2. **Visualización del nuevo cliente:** Después de registrar al nuevo cliente, el sistema mostró la nueva entrada en la tabla de clientes. Sin embargo, como se anticipó, solo el campo "Nombre" estaba lleno, mientras que todos los demás campos permanecían vacíos.
3. **Falta de notificación:** El sistema no notificó al administrador de los campos vacíos durante el proceso de registro, a pesar de que era una acción esperada en el caso de campos obligatorios incompletos.
4. **Registro incorrecto en la base de datos:** A pesar de la falta de información completa, el sistema agregó el registro incompleto del nuevo cliente a la base de datos. Esto es un error, ya que el sistema debería haber evitado el registro debido a la información incompleta.
5. **Corrección del error:** Para resolver este problema, se siguió el flujo del caso de uso "Eliminar Cliente". Esta acción eliminó el registro incorrecto del cliente tanto de la tabla de clientes como de la base de datos.

Estos resultados obtenidos destacan varios problemas con el sistema de registro de clientes, específicamente en lo que respecta a la validación del formulario y la prevención de registros incorrectos. Los problemas identificados deben ser corregidos para garantizar que el sistema funcione correctamente y que se mantenga la integridad de los datos.

### COMPARAR RESULTADOS

- Estado (aprobado / fallido):

#### FALLIDO

Dado que los resultados obtenidos durante la prueba no coinciden con los resultados esperados, y el sistema permitió el registro de un cliente con información incompleta sin proporcionar ninguna notificación, el estado de esta prueba es **Fallido**. Este resultado indica que se identificó un defecto en el sistema, específicamente en la validación del formulario de registro y la prevención del

registro de datos incompletos, que debe ser corregido para asegurar el correcto funcionamiento del sistema y la integridad de los datos.

#### REGISTRAR RESULTADOS

Elemento del Caso de Prueba	Detalle
<b>Descripción del Caso de Prueba</b>	Validar el comportamiento del sistema al intentar registrar un nuevo cliente con información incompleta.
<b>Requisito Funcional Asociado</b>	Registrar Cliente
<b>Entradas y Condiciones Previas</b>	El administrador ha iniciado sesión correctamente y el sistema tiene acceso a la base de datos. Solo el campo "NOMBRE" está lleno, los demás campos están vacíos.  A. NOMBRE: Pablo B. TELEFONO: C. DIRECCION: D. CORREO: E. CUMPLEAÑOS: F. HOBBIES: G. MOLESTIAS:
<b>Pasos para Ejecutar el Caso de Prueba</b>	El administrador sigue el flujo normal hasta el paso 3, luego ingresa solo el dato de "NOMBRE" y deja los demás campos vacíos.
<b>Resultados Esperados</b>	El sistema debería validar el formulario, notificar al administrador de los campos vacíos, prevenir el registro incorrecto y solicitar al administrador que ingrese la información completa del nuevo cliente.
<b>Resultados Obtenidos</b>	El sistema permitió el registro del cliente con solo el nombre, sin notificar al administrador sobre los campos vacíos. Este registro incorrecto fue almacenado en la base de datos. Se corrigió el problema eliminando el registro incorrecto del cliente siguiendo el caso de uso "Eliminar Cliente".
<b>Estado</b>	Fallido

## PRUEBAS DE CAJA BLANCA

Las pruebas de caja blanca, también conocidas como pruebas de caja clara o pruebas de estructura lógica interna, son un método de pruebas de software en el que el diseño interno o el código de un programa son examinados para probar secciones específicas del software.

Este método de prueba está basado en el conocimiento detallado del funcionamiento interno del código que se está probando. El término "caja blanca" se refiere a la transparencia del sistema de software: el probador puede ver "dentro de la caja" y, por lo tanto, se concentra en cómo funciona el software.

Las pruebas de caja blanca son distintas de las pruebas de caja negra, que sólo se preocupan por el funcionamiento externo del software, es decir, las pruebas de caja negra no consideran cómo el sistema produce el resultado.

Aquí están algunos puntos clave acerca de las pruebas de caja blanca:

1. **Cobertura de código:** Las pruebas de caja blanca se llevan a cabo con el objetivo de cubrir todas las posibles rutas de código. Esto ayuda a asegurar que todas las líneas de código y todas las rutas posibles de ejecución se hayan probado.
2. **Diseño de pruebas:** Las pruebas de caja blanca se basan en el diseño y la implementación del código fuente. Los probadores deben tener un conocimiento detallado del código y utilizar este conocimiento para diseñar casos de prueba.
3. **Tipos de pruebas de caja blanca:** Algunos de los tipos de pruebas de caja blanca incluyen las pruebas de control de flujo, las pruebas de ruta básica, las pruebas de condición, las pruebas de ciclo y las pruebas de lazo.
4. **Ventajas y desventajas:** Las ventajas de las pruebas de caja blanca incluyen la detección temprana de errores y la optimización del código. Las desventajas incluyen la posibilidad de que las pruebas se vuelvan demasiado complejas si el software es muy grande o si el código cambia frecuentemente.

En conclusión, las pruebas de caja blanca son un tipo de prueba de software que busca asegurar que el código interno funcione correctamente. Son una parte importante del proceso de garantía de calidad del software. Sin embargo, deben complementarse con otros tipos de pruebas, como las pruebas de caja negra, para proporcionar una evaluación exhaustiva de un sistema de software.

Asd