

Instituto Tecnológico de Costa Rica

Centro Académico de Alajuela

IC-5701: Compiladores e Intérpretes



**Informe del Proyecto:**

*Ejecución de pruebas, informe de errores y problemas encontrados*

**Subgrupo:**

William Alfaro Quirós – Carné: 2022437996

Jesus Cordero Díaz - Carné: 2020081049

Anthony Rojas Fuentes - Carné: 2018027141

**Profesor:**

Mag. Andrés Víquez Víquez

**Fecha de entrega:**

24/11/2024

II Semestre, 2024

## Índice

Descripción general del proceso y del ambiente de pruebas.....	2
Informe de cobertura.....	4
Cantidad de casos de prueba exitosos.....	5
Cantidad de casos de prueba fallidos.....	6
Clasificación de los errores encontrados con base en su prioridad.....	8
Observaciones y comentarios.....	9
Análisis de problemas encontrados.....	10

## **Descripción general del proceso y del ambiente de pruebas**

### **Proceso de pruebas**

En el proceso de pruebas seguimos una serie de pasos bien organizados para garantizar que todas las funcionalidades principales del sistema OpenSourcePOS fueran verificadas adecuadamente, a continuación describimos los principales pasos:

### **Planeación de los casos de prueba**

Se comenzó identificando las áreas clave del sistema que requerían validación, centrándose en módulos esenciales como la gestión de clientes, proveedores e inventarios. Además, se evaluaron específicamente las validaciones en formularios para asegurar el cumplimiento de las reglas de negocio. A partir de este análisis, se diseñaron casos de prueba detallados que incluyen una descripción precisa de los pasos a seguir, las precondiciones necesarias y los resultados esperados en cada escenario.

### **Automatización de las pruebas**

En el proceso de automatización, se logró automatizar el 60% de los casos de prueba utilizando Java en su versión JDK 23 y Selenium WebDriver. El enfoque principal estuvo en las funciones críticas del sistema, especialmente aquellas relacionadas con la validación de campos obligatorios y las operaciones CRUD (Crear, Leer, Actualizar, Eliminar). Este esfuerzo permitió optimizar los tiempos y garantizar una cobertura efectiva en las áreas más sensibles del sistema.

### **Ejecución de pruebas**

Las pruebas automatizadas se ejecutaron en dos entornos diferentes: Windows 11 y la distribución de Linux Ubuntu 24.04. Para garantizar la compatibilidad y el rendimiento del sistema, se utilizó Google Chrome en su versión 131.0.6778.86, junto con el driver correspondiente de Selenium. Este enfoque permitió verificar el comportamiento del sistema en diferentes plataformas y entornos.

### **Registro y reporte de resultados**

Se documentaron los resultados de cada prueba, identificando tanto las pruebas exitosas como las que presentaron fallos. Los errores detectados se clasificaron en niveles de prioridad: críticos, altos, medios y bajos. Toda la información se consolidó en un informe exhaustivo que facilitó el análisis del estado del sistema y la priorización de las correcciones necesarias.

### **Ambiente de pruebas**

Configuramos un entorno de pruebas que nos permitiera replicar un ambiente realista y controlado para garantizar que los resultados fueran consistentes. A continuación, se detallan las características principales del entorno:

#### **Sistemas operativos utilizados:**

- **Windows 11:** Para evaluar la compatibilidad en uno de los sistemas más comunes en el ámbito empresarial.
- **Ubuntu 22.04:** Para probar el sistema en un entorno basado en Linux, típico de servidores y desarrolladores.

#### **Herramientas y tecnologías empleadas:**

- **Lenguaje de programación:** Java, en su versión JDK 23, fue nuestra elección para desarrollar las pruebas automatizadas.
- **Selenium WebDriver:** Lo utilizamos para automatizar las pruebas de interfaz de usuario, aprovechando sus capacidades para interactuar con el navegador.
- **Google Chrome:** Probamos exclusivamente en este navegador, asegurándonos de utilizar la versión 131.0.6778.86 con su respectivo driver para Selenium.

## Informe de cobertura

En el marco del proyecto, se diseñaron un total de 75 casos de prueba para cubrir las principales funcionalidades del sistema OpenSourcePOS. De estos, se lograron automatizar 45 casos de prueba, lo que representa una cobertura del 60%.

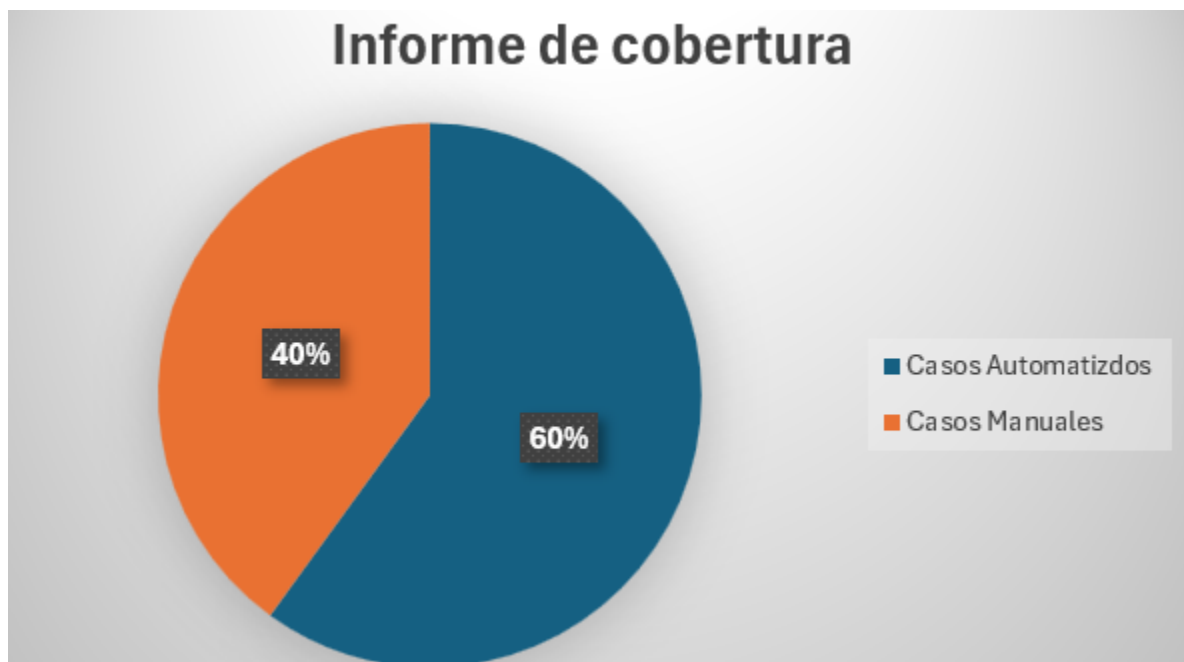
### Detalles de la cobertura

- **Casos de prueba automatizados:** 45.
- **Casos de prueba manuales:** 30.

### Distribución de los casos automatizados

Los casos de prueba automatizados abarcaron las funcionalidades más críticas del sistema, incluyendo:

- Validaciones de formularios de entrada.
- Verificación de campos obligatorios y sus mensajes de error.
- Operaciones CRUD (Crear, Leer, Actualizar, Eliminar) en los módulos de clientes, proveedores e inventarios.
- Pruebas de funcionalidad para acciones específicas como envío de mensajes SMS, exportación de datos y actualizaciones de stock.



### Cantidad de casos de prueba exitosos

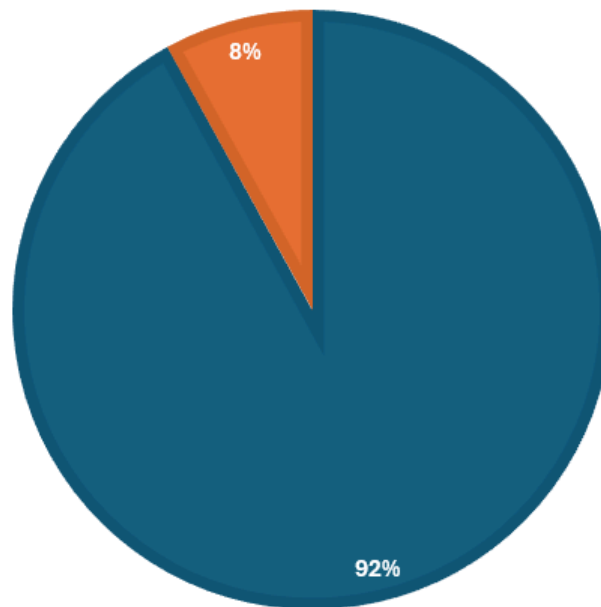
Entre los 45 casos de prueba automatizados y los 30 casos manuales, 69 resultaron exitosos, lo que representa un 92% de éxito en las pruebas realizadas. Estos casos validaron correctamente las funcionalidades clave del sistema, como:

- La gestión de clientes, proveedores e inventarios.
- Las validaciones de formularios en escenarios con datos válidos.
- Operaciones CRUD en diferentes módulos del sistema.

Los resultados reflejan un buen nivel de estabilidad en el sistema para las funcionalidades probadas, con margen para mejorar en las áreas identificadas con fallos.

### ANALISIS DE CASOS

■ casos de prueba exitosos ■ Casos de prueba fallidos



### **Cantidad de casos de prueba fallidos**

De los 45 casos de prueba automatizados ejecutados, 6 casos fallaron, lo que representa un 8% de fallos en la ejecución total. A continuación, se detallan los principales problemas identificados:

#### **Formato inválido en correos electrónicos (Caso de Prueba No. 001):**

- **Descripción:** El formulario "New Customer" permite enviar correos electrónicos con un formato incorrecto, sin mostrar el mensaje de error correspondiente.
- **Impacto:** Afecta la integridad de los datos.

#### **Formato inválido en números de teléfono (Caso de Prueba No. 002):**

- **Descripción:** El formulario "New Customer" acepta números de teléfono en un formato no válido, sin restringir su envío.
- **Impacto:** Puede generar inconsistencias en los datos de contacto.

#### **Error en el envío de mensajes SMS (Caso de Prueba No. 013):**

- **Descripción:** El formulario "Send SMS" no permite enviar mensajes aunque todos los campos estén correctamente completados.
- **Impacto:** Limita una funcionalidad clave del sistema.

#### **Campos de precios con caracteres no numéricos (Caso de Prueba No. 015):**

- **Descripción:** Los campos "Wholesale Price" y "Retail Price" en el formulario "New Item" aceptan caracteres no numéricos, permitiendo el envío de datos incorrectos.
- **Impacto:** Afecta la calidad de los datos relacionados con precios.

#### **Cantidad adicional de un mismo producto (Caso de Prueba No. 053):**

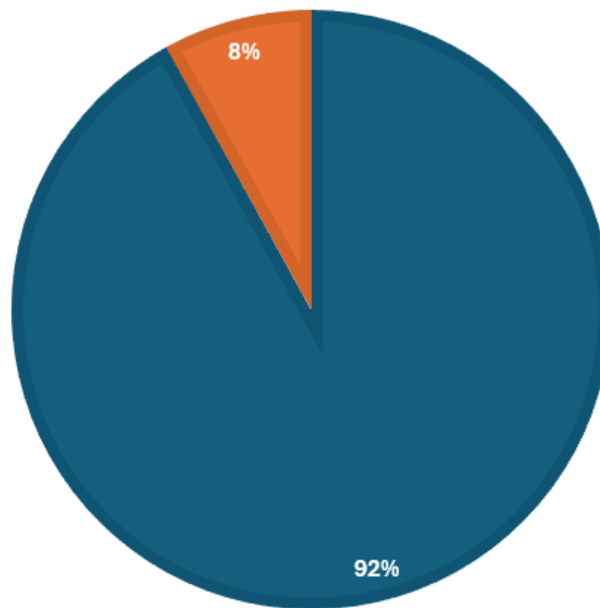
- **Descripción:** El sistema muestra un mensaje de error al intentar agregar más cantidad de un producto ya existente en el carrito de ventas, impidiendo que las cantidades se sumen correctamente.
- **Impacto:** Afecta la gestión de inventarios y el flujo de ventas al no permitir agregar cantidades adicionales del mismo producto.

#### **Error tipográfico en los campos "Opened By" y "Closed By" (Caso de Prueba No. 067):**

- **Descripción:** Al crear un nuevo Cashup, el sistema guarda correctamente el registro, pero muestra un error tipográfico en el campo "Closed By", donde aparece "Jhon Doe" en lugar de "John Doe".
- **Impacto:** Compromete la precisión de la información, lo que puede causar problemas en la identificación de los responsables de los registros.

## ANALISIS DE CASOS

■ casos de prueba exitosos    ■ Casos de prueba fallidos





## **Clasificación de los errores encontrados con base en su prioridad**

Durante la ejecución de los casos de prueba, se identificaron 6 errores clasificados según su prioridad en alta y media, dependiendo del impacto en el sistema y la urgencia de su resolución.

### **Errores de prioridad alta**

Estos errores tienen un impacto significativo en la funcionalidad del sistema y deben ser atendidos con urgencia:

- Error en el envío de mensajes SMS (Caso de Prueba No. 013):
- Campos de precios con caracteres no numéricos (Caso de Prueba No. 015):
- Cantidad adicional de un mismo producto (Caso de Prueba No. 053):
- Error tipográfico en los campos "Opened By" y "Closed By" (Caso de Prueba No. 067):

### **Errores de prioridad media**

Estos errores afectan la calidad de los datos y la experiencia del usuario, pero no son críticos para la operación del sistema:

- Formato inválido en correos electrónicos (Caso de Prueba No. 001):
- Formato inválido en números de teléfono (Caso de Prueba No. 002):

## **Observaciones y comentarios**

El sistema tuvo un buen desempeño durante las pruebas, con un 92% de casos exitosos. Sin embargo, se identificaron algunos problemas críticos, principalmente en la validación de datos (como correos electrónicos, números de teléfono y precios) y en el envío de mensajes SMS, que afecta una funcionalidad clave. La automatización permitió cubrir el 60% de los casos de prueba, enfocándose en las funcionalidades más importantes. Las pruebas se realizaron en dos entornos operativos, Windows 11 y Ubuntu, y no se encontraron problemas relacionados con el ambiente de pruebas. Esto aseguró que el sistema tiene buena compatibilidad en ambos entornos.

## **Análisis de problemas encontrados**

Durante el desarrollo del proyecto enfrentamos varios desafíos que inicialmente complicaron el avance, pero logramos superarlos gracias a la colaboración en equipo. Uno de los principales problemas fue el retraso en la conexión a la base de datos, lo que nos tomó cinco días adicionales debido a pruebas y configuraciones necesarias para implementar los tests del segundo entregable. También tuvimos dificultades al decidir la herramienta de automatización, ya que inicialmente consideramos Cypress, pero los errores asociados al uso de TypeScript nos llevaron a optar por Selenium WebDriver con Java, un entorno más familiar para el equipo. Además, la coordinación fue un reto, ya que al inicio no coincidimos en horarios, lo que solucionamos asignando tareas claras y organizando revisiones por pares. En cuanto a los resultados, completar los tests automatizados nos llevó 15 días en lugar de los 10 previstos debido a los retrasos iniciales, y logramos cubrir el 60% de los casos de prueba con automatización, frente al 76% esperado. Sin embargo, la implementación de revisiones ayudó a mejorar el flujo de trabajo y mantener la calidad del proyecto.