



TRABAJO DE GRADO
Opción Seminario-Diplomado.

Custom Flow QA

Corporación Universitaria Remington.
Ingeniería de Sistemas:
Pruebas de Software:

Jhon Mauricio Alzate Garcia - Yeimer Ferney Henao Montoya.
Docente: Fabio León Grisales Morales
Opción de Trabajo de grado Seminario-Diplomado.
14/12/2025

Tabla de Contenidos

Resumen.....	3
Palabras clave.....	4
Marco conceptual y contextual	5
Desarrollo e implementación del aprendizaje.....	6
Tabla 1. Resultados de ejecución de casos de prueba.....	7
Figura 1. Distribución de defectos por impacto (Bugtracker)	8
Figura 2. Comparación entre casos de prueba planificados y ejecutados.....	9
Figura 3. Resultados de ejecución de casos de prueba	10
Tabla 2. Resumen del plan de pruebas realizado	12
Conclusiones	13

Resumen

El trabajo de grados muestra el proceso y los resultados de diseño y documentación completa del Plan de Pruebas Funcionales realizado por la empresa Customer Flow QA para una plataforma web de comercio electrónico enfocada en la venta de ropa y accesorios. El objetivo principal fue garantizar que cada módulo del sistema cumpliera correctamente con los requerimientos definidos, asegurando una experiencia estable, funcional y confiable para el usuario final.

Durante el desarrollo del proyecto, se realizó un análisis detallado de las historias de usuario, el alcance del sistema y los riesgos asociados. A partir de este estudio, se elaboró una estructura organizada de prueba que incluye: definición del alcance, análisis de riesgo, estrategias de pruebas, identificación de los tipos de pruebas necesarios y la construcción de una matriz de 58. Casos que cubren los procesos claves del sistema. Como visualización del catálogo, registro, inicio de sesión, manejo de carrito, validación del checkout, confirmación de compras, generador de PDF. Historial de compras y recuperación de contraseñas

Cada caso de prueba fue diseñado con un enfoque práctico, siguiendo la trazabilidad entre alcance e historia de usuario. Confirmaciones Caso de prueba, garantizando que cada funcionalidad crítica estuviera correctamente validada. Este trabajo no solo permitió evaluar la calidad del sistema, sino que también aplicó de manera directa lo aprendido en el seminario, integrando conocimiento sobre análisis funcional, metodología de aseguramiento de calidad, redacción técnica y documentación profesional.

En conjunto, este proyecto representa la base estructurada para asegurar la calidad del sistema, permitiendo detectar fallos antes de una puesta en producción y fortaleciendo la confiabilidad del software; además, demuestra la importancia de un plan de pruebas sólido como parte esencial en el desarrollo de aplicaciones web actuales..

Palabras clave

- Pruebas Funcionales
- Calidad de Software
- Casos de Pruebas
- Comercio Electronico
- Customer Flow QA

Marco conceptual y contextual

El trabajo que estamos presentando se desarrolla en el marco del seminario de pruebas de software, cuyo propósito es formar al estudiante en la aplicación real de técnicas, metodologías y buenas prácticas de aseguramiento de calidad (QA). Este informe técnico refleja la implementación de los conocimientos adquiridos aplicados a un proyecto práctico donde se elaboró un plan de pruebas funcionales completo para una plataforma de comercio electrónico. En conjunto con la empresa Customer Flow QA, organización dedicada al análisis, diseño y ejecución de pruebas orientadas a la mejora continua de productos digitales.

Desde el punto de vista conceptual, el trabajo se fundamenta en la calidad del mismo software entre ellos.

- Las Pruebas Funcionales
- Casos de pruebas
- Historias de usuario
- Análisis de riesgo
- Trazabilidad

En cuanto al contexto del trabajo, el proyecto se centra en una tienda virtual que ofrece ropa y accesorios que requiere un proceso de validación formal para garantizar que su funcionamiento sea adecuado antes de su despliegue. Este tipo de sistema incluye operaciones críticas como visualización de productos, registro de usuarios, inicio de sesión, carrito de compras, checkout, envío de correos y generación de comprobantes, las cuales deben ser verificadas con precisión para evitar fallos que afecten la experiencia del usuario y la confiabilidad del negocio.

La empresa Customer Flow QA permitió simular un entorno profesional de pruebas, donde se aplicaron las metodologías enseñadas en el diplomado. Este entorno facilitó la construcción de un plan de pruebas sólido, apoyado en una matriz estructurada de casos, análisis de riesgo, definición del alcance y documentación técnica clara, logrando así una aplicación práctica y completa de lo aprendido en el seminario.

Desarrollo e implementación del aprendizaje

El seminario "Pruebas de Software" nos brindó la oportunidad de poder aplicar de manera práctica los conocimientos adquiridos en cada una de las clases en un entorno académico simulado, utilizando una aplicación web de una tienda online.

Como objetivo principal fue evaluar la calidad del software mediante diferentes actividades que íbamos viendo en cada una de las clases basadas en aseguramiento de calidad, también siguiendo las buenas prácticas vistas durante el curso.

Como parte de la implementación se aplicó el ciclo de vida de una prueba de software, iniciando con trabajos de elaboración de plan de pruebas. En el documento se definió el alcance, los módulos a validar, los riesgos asociados, los criterios de entrada y salida, también como la estrategia general.

En esta etapa establecimos una base clara para una ejecución controlada de los procesos de QA

Después, se diseñaron los casos de pruebas que tomaron funcionalidades críticas del sistema, como registro de usuarios, inicio de sesión persistencia del carrito de compras, validaciones en el proceso de checkout, confirmación de compra, generación de comprobantes y gestión del perfil del usuario.

Cada caso se documentó de forma formal, incluyendo precondiciones, pasos de ejecución, resultados esperados y resultados también obtenidos, lo que ayudo y facilito la trazabilidad y el análisis posterior

La ejecución de las pruebas se ejecuta de forma manual con el entorno simulado, lo que nos permite identificar diferentes comportamientos incorrectos en el sistema. Los defectos controlados también se registraron en una bugtracker, clasificándolos según su severidad e impacto. Esta práctica evidenció la importancia de una correcta documentación para priorizar y corregir errores de manera eficiente.

Adicionalmente, aplicamos un **análisis de riesgos**, evaluando la probabilidad y también el impacto de posibles fallos relacionados con el manejo de sesiones, validaciones insuficientes y errores en procesos críticos del negocio. Este análisis nos ayudó a priorizar los esfuerzos de prueba sobre los componentes de mayor riesgo para el sistema.

Como resultado del proceso, se obtuvieron métricas de QA que nos permitieron evaluar el desempeño del plan de pruebas, tales como el número de casos que se ejecutaron, también la cobertura alcanzada, la cantidad de defectos detectados y la comparación entre el tiempo estimado y el tiempo real de ejecución. Estas métricas evidenciaron la

efectividad de los procesos de prueba y sirvieron como insumo para la toma de decisiones durante el cierre del mismo proyecto.

En conclusión, la implementación que realizamos de forma práctica del aprendizaje adquirido nos permite aplicar de forma integral los conceptos que vimos en el seminario, demostrando la relevancia del aseguramiento de calidad en el desarrollo de aplicaciones web. El ejercicio nos facilitó la comprensión del rol de QA dentro de un ciclo de desarrollo de software y fortaleció competencias clave como el análisis, la documentación, la identificación de riesgos y la mejora continua de los procesos de prueba.

Como parte del análisis de resultados, se tomaron las métricas que permitieron evaluar la efectividad del proceso de pruebas ejecutado. En la **Tabla 1** se presenta el resumen de los casos de prueba ejecutados y su resultado, mientras que en la **Figura 1** se evidencia la distribución de defectos según su nivel de impacto, información clave para priorizar correcciones y tomar decisiones sobre la calidad del sistema evaluado.

Métrica	Valor
Casos de prueba planificados	49
Casos de prueba ejecutados	49
Casos correctos	23
Casos con error	26
Cobertura de pruebas	100 %

Tabla 1. Resultados de ejecución de casos de prueba

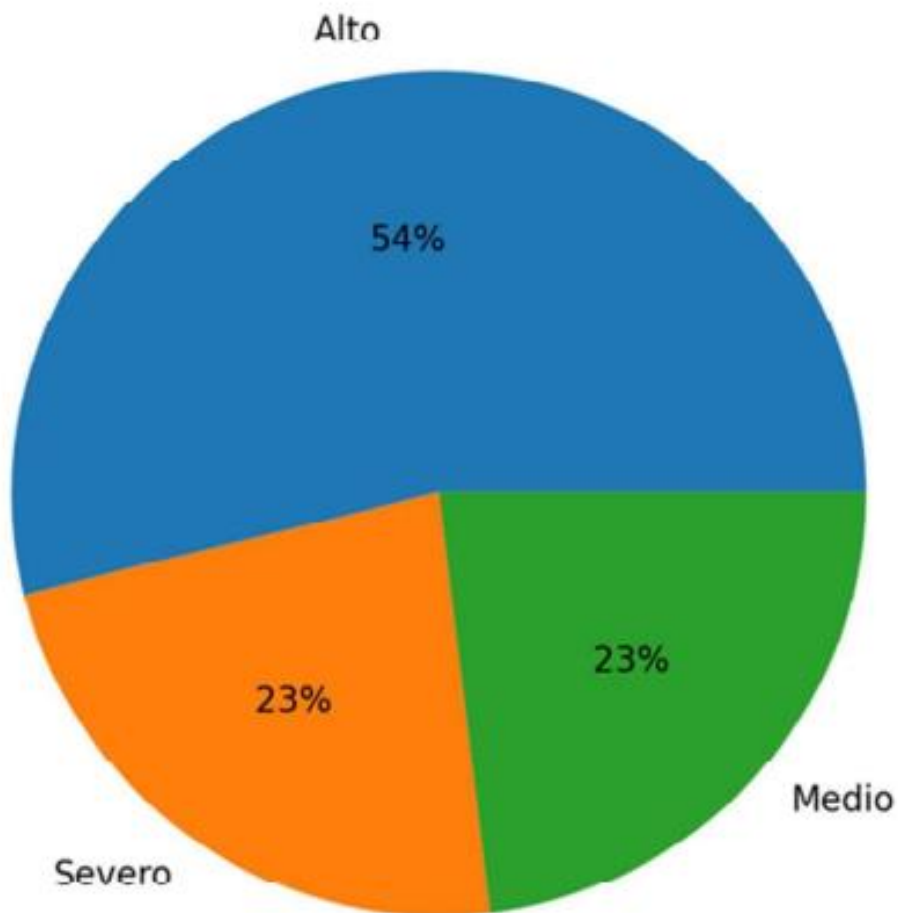


Figura 1. Distribución de defectos por impacto (Bugtracker)

La Figura 1 muestra la distribución de los defectos detectados durante la ejecución de las pruebas, evidenciando que una parte significativa corresponde a defectos de impacto alto y severo. Este resultado resalta la importancia de realizar pruebas tempranas y exhaustivas en funcionalidades críticas del sistema, como el manejo de sesiones, validaciones y procesos de compra.

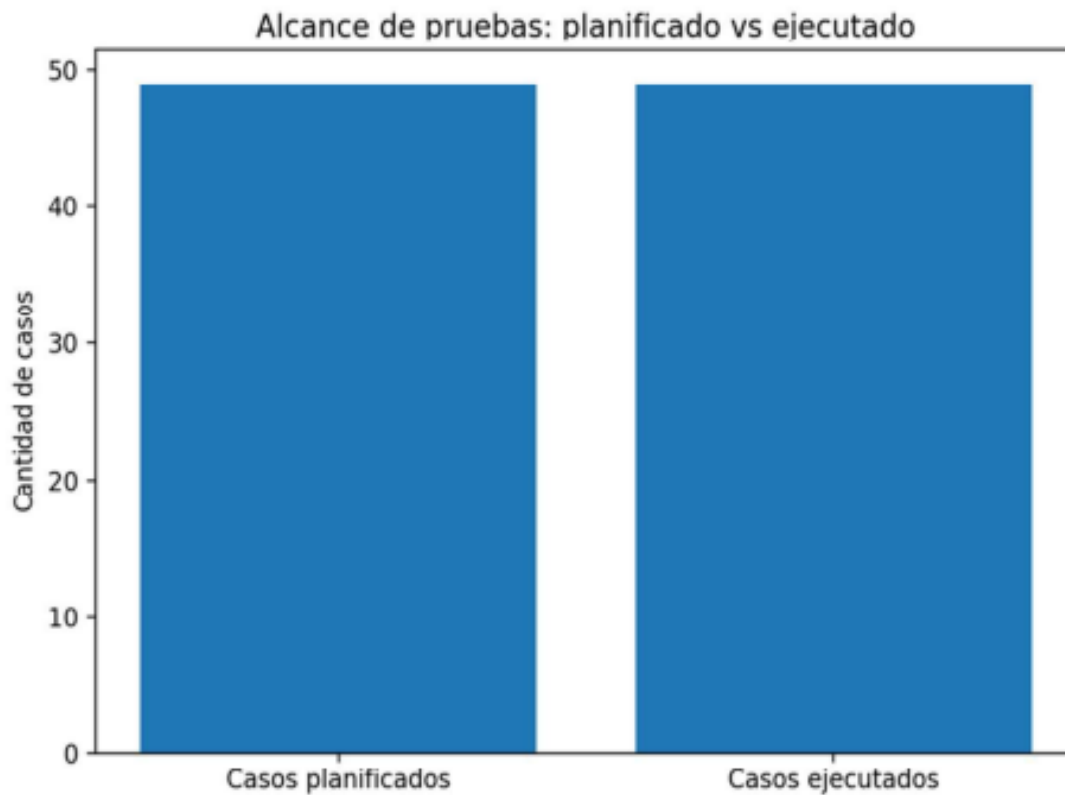


Figura 2. Comparación entre casos de prueba planificados y ejecutados

La Figura 3 muestra la distribución de los resultados obtenidos durante la ejecución de los casos de prueba, donde se identifican casos correctos y casos con error, información clave para evaluar la calidad del sistema y priorizar correcciones.

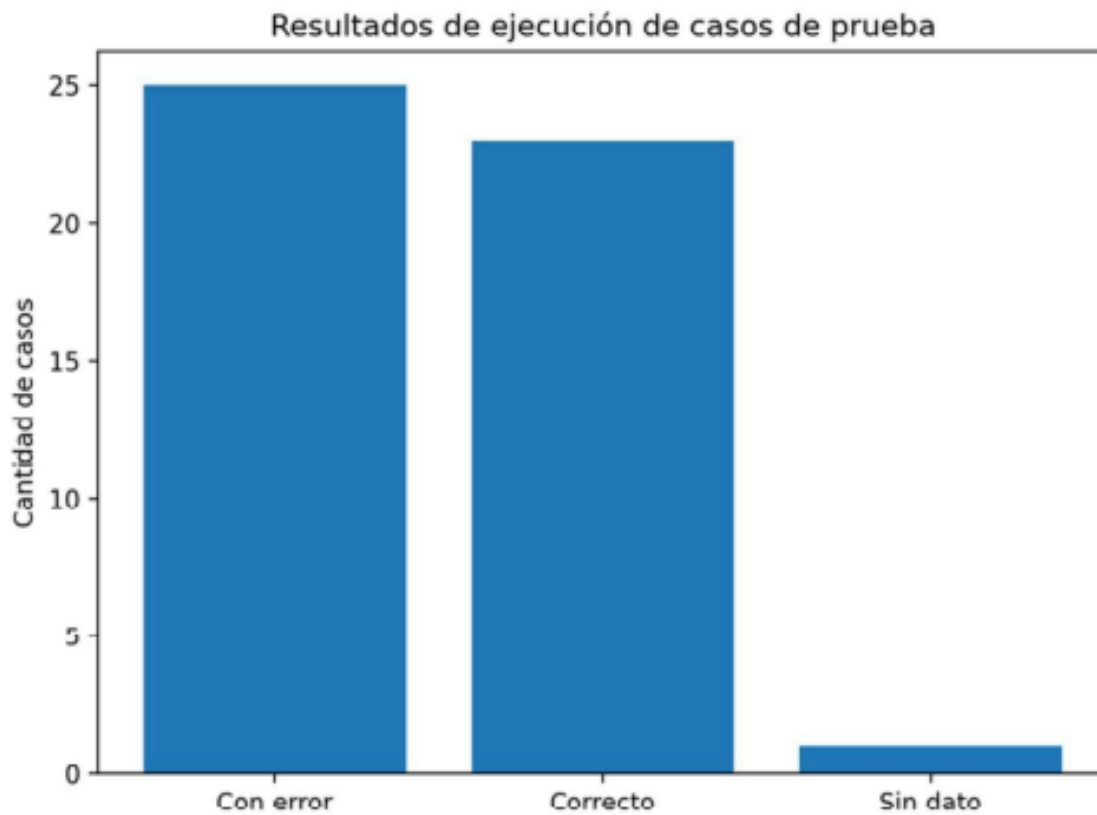


Figura 3. Resultados de ejecución de casos de prueba

La Figura 3 muestra la distribución de los resultados obtenidos durante la ejecución de los casos de prueba, donde se identifican casos correctos y casos con error, información clave para evaluar la calidad del sistema y priorizar correcciones.

Adicionalmente, los ejercicios que desarrollamos nos permitio evidenciar la similitud entre un ambiente academico simulado y escenarios de la vida real en el desarrollo de software, donde nos permitio asegurar que la calidad cumple el rol de prevenir fallos graves antes de salir a producción. La aplicación controlada de actividades como la planeación de pruebas, la ejecución controlada de casos, gestión de defectos y el analisis de riesgos se alinea con las buenas practicas de la industria del software, las cuales nos ayudan con la entrega de productos mucho mas confiables, estables y tambien orientados al usuario final. De esta manera, el seminario no solo nos fortalecio con conocimientos tecnicos en pruebas de software, si no que tambien nos ayudo a desarrollar una visión mas integral y tecnica dentro del rol de QA y el ciclo de vida del desarrollo de software.

Tabla 2. Resumen del plan de pruebas realizado

<i>Elementos evaluados</i>	<i>Resultado</i>
Tipo de pruebas	Pruebas funcionales manuales
Alcance del plan	12 módulos funcionales
Casos de prueba planificados	49
Casos de prueba ejecutados	49
Cobertura alcanzada	100%
Casos correctos	23
Casos con defectos	26
Total de defectos encontrados	26
Bugs críticos / severos	Aproximadamente 40 %
Herramienta de gestión de defectos	Bugtracker
Tiempo estimado del plan	60 horas
Tiempo real con desviación	105,6 horas
Fase con mayor consumo de tiempo	Diseño y Ejecución
Riesgos materializados	Pérdida de sesión y validaciones insuficientes
Resultado general	Plan de pruebas ejecutado completamente
Bugs Impacto severo	6
Bugs de impacto alto	14
Bugs de impacto medio	6

Conclusiones

- Poner en práctica lo del diplomado me dejó claro que un plan de pruebas bien hecho no es “probar por probar”, sino validar con criterios, alcance, casos, resultados y evidencia.
- La trazabilidad entre historias de usuario fue clave para no perdernos y asegurarnos de que todo lo importante quedara cubierto y no se desviara.
- Ejecutar los 49 casos de prueba nos permitió medir el sistema con datos reales y no con suposiciones. Además, el 100% de cobertura nos dejó un poco tranquilos porque no quedaron pruebas pendientes ni dudas sin resolver.
- Registrar los defectos en el Bugtracker (26 bugs) ayudó a clasificar el impacto, priorizar y justificar técnicamente por qué ciertas partes del sistema necesitan corrección prioritaria.
- El análisis de tiempos nos confirmó que donde más gasta tiempo es en ejecución y regresión, porque cada bug obliga a repetir pruebas y validar que no se rompa lo que ya funcionaba.
- En general, el ejercicio me demostró que QA no solo encuentra errores, sino que también mejora el proceso reduciendo los riesgos y da confianza antes de pasar a producción.

