

TRABAJO DE GRADO
Opción Seminario-Diplomado.

Ciclo de Vida de las Pruebas en una Tienda Online – Equipo QA Squad

Corporación Universitaria Uniremington.

Facultad de Ingeniería

Ingeniería de Sistemas

Luisa Fernanda Aristizabal Giraldo

Diego Alejandro Muñoz Robayo

Jorge Andres Ricardo Arcia

Tutor: Fabio Leon Grisales Morales

Seminario de Pruebas de Software

Agradecimientos

Agradecemos a la Corporación Universitaria Uniremington y a la Facultad de Ingeniería por el espacio de formación, y de manera especial a nuestro tutor, Fabio Leon Grisales Morales, por su acompañamiento constante durante el seminario de pruebas de software: sus explicaciones, retroalimentaciones y exigencia práctica hicieron posible que este proyecto se desarrollara con la calidad que aquí se presenta.

Tabla de Contenidos	
Resumen.....	5
Marco conceptual y contextual	6
Conceptos fundamentales del proceso de pruebas.....	6
Técnicas de diseño de casos de prueba	7
Gestión de defectos y métricas	7
Marco contextual	8
Desarrollo e implementación del aprendizaje.....	8
Aclaración de requisitos: preguntas al cliente	9
Planeación: plan de pruebas v1 y su refinamiento a v2	10
Análisis de pruebas: condiciones y técnicas	12
Diseño de los casos de prueba	14
Implementación y ejecución de las pruebas.....	14
Resultados de la ejecución	15
Gestión de defectos: Bugtracker	18
Seguimiento, control e informe de cierre.....	21
Conclusiones	23
Referencias.....	25

Ilustración 1 Cobertura del alcance v2: 38 casos ejecutados (31 aprobados, 7 fallidos) y 11 no aplicables.....	15
Ilustración 2 Resultado de la ejecución por historia de usuario: los fallos se concentran en login, carrito y compra.	17
Ilustración 3 Evidencia del caso CP-03-02 (aprobado): el sistema rechaza un correo ya registrado con mensaje claro.....	17
Ilustración 4 . Evidencia del caso CP-06-01 (fallido): se añade un producto al carrito sin haber iniciado sesión.....	18
Ilustración 5 Reportes por tipo: la mitad fueron hallazgos, lo que confirma el valor de las pruebas exploratorias.....	20
Ilustración 6 Reportes por naturaleza: la seguridad concentra 4 de los 8 defectos.....	20
Ilustración 7 Tiempos estimados vs. reales por actividad: 105 h reales frente a 118 h planificadas (-11 %).	21
Tabla 1 Reglas de negocio confirmadas por el cliente (respuestas 16 a 19).....	9
Tabla 2 Alcance de ejecución v2: historias de riesgo Alto y casos por analista (total 49 casos).	11
Tabla 3 Técnicas de diseño aplicadas y su justificación en el proyecto.	13
Tabla 4 Resultado de la ejecución por historia de usuario.....	16
Tabla 5 Reportes registrados en el Bugtracker durante la ejecución (14 y 15 de julio de 2026).	19

Resumen

Este trabajo de grado presenta la aplicación práctica del proceso completo de pruebas de software sobre una aplicación web de comercio electrónico (Tienda Online de ropa y accesorios), desarrollada como proyecto del seminario de pruebas de software. El cliente (rol asumido por el docente) requería verificar que las 14 historias de usuario del rol cliente cumplieran sus criterios de aceptación, con énfasis en el flujo crítico del negocio: registro, inicio de sesión, catálogo, carrito, compra con pago y confirmación por correo con PDF adjunto. El equipo QA Squad ejecutó todas las etapas del proceso de pruebas: formulación de 26 preguntas al cliente para aclarar requisitos; planeación con un plan de pruebas bajo IEEE 829 adaptado, refinado a una versión 2 mediante matriz de riesgos (Probabilidad $\times$ Impacto) que delimitó el alcance a las 7 historias de riesgo Alto; análisis de condiciones de prueba criterio por criterio; diseño de 49 casos de prueba aplicando técnicas de caja negra (particiones de equivalencia, valores límite, tablas de decisión, transición de estados) y basadas en la experiencia; implementación y ejecución en el ambiente de pruebas demo.prestashop.com; y cierre con informe final. Se ejecutaron 38 casos (77,6 % del alcance) con una tasa de aprobación del 81,6 %, y se reportaron 8 defectos en el Bugtracker (4 errores y 4 hallazgos, 6 de ellos de impacto severo, concentrados en seguridad y en el flujo de pago). El informe de cierre evaluó el alcance real contra el estimado, los riesgos materializados, la desviación de tiempos (105 h reales frente a 118 h planificadas) y las métricas de defectos, y concluyó con la recomendación de no aptitud para producción hasta corregir los defectos severos. El trabajo evidencia el dominio práctico de los conceptos del seminario: estrategia basada en riesgo, diseño estructurado de casos, gestión de defectos, seguimiento y control, y comunicación de resultados mediante métricas.

Palabras clave

Pruebas de software · Plan de pruebas · Diseño de casos de prueba · Gestión de defectos ·
Informe de cierre

Marco conceptual y contextual

Las pruebas de software son el proceso mediante el cual se evalúa un producto para verificar que cumple los requisitos acordados (verificación) y que satisface la necesidad del cliente (validación), aportando información objetiva sobre su calidad antes de la salida a producción. El seminario abordó este proceso como un ciclo de actividades: planeación, análisis, diseño, implementación, ejecución, y finalización o cierre, acompañadas de la actividad transversal de seguimiento y control. A continuación se presentan los conceptos aplicados en el trabajo y el contexto en el que se desarrolló.

Conceptos fundamentales del proceso de pruebas

El plan de pruebas es el documento rector del proceso: define objetivos, alcance, estrategia, criterios de entrada y salida, riesgos, estimación de tiempos, cronograma, roles y entregables; es un documento vivo que se refina a medida que el proyecto avanza. Este trabajo adoptó el estándar IEEE 829 adaptado con prácticas del ISTQB. La estrategia aplicada fue la de pruebas basadas en riesgo: cada historia de usuario se valoró con la fórmula $\text{Riesgo} = \text{Probabilidad} \times \text{Impacto}$ (escala 1 a 5) y el esfuerzo se dirigió a las historias de nivel Alto, aplicando el principio de que las pruebas exhaustivas son inviables.

Técnicas de diseño de casos de prueba

Se aplicaron técnicas dinámicas de caja negra: particiones de equivalencia (agrupar las entradas en clases válidas y no válidas y probar un representante de cada una), análisis de valores límite (probar en las fronteras de los rangos, donde se concentran los defectos), tablas de decisión (enumerar combinaciones de condiciones y su acción esperada) y transición de estados (verificar los cambios de estado permitidos y prohibidos del sistema). Se complementaron con técnicas basadas en la experiencia: pruebas exploratorias con sesiones documentadas, predicción de errores y listas de comprobación.

Gestión de defectos y métricas

Un defecto o bug se reporta cuando el resultado obtenido difiere del esperado. El seminario distinguió tres tipos de incidente: error (falla en un caso de prueba diseñado), hallazgo (defecto encontrado fuera del alcance diseñado) y sugerencia (oportunidad de mejora). Cada reporte registra impacto (severo, alto, medio, bajo), prioridad (urgente, alta, media, baja), naturaleza (funcional, seguridad, performance, hardware/software, entre otras), pasos para reproducirlo y estado dentro de su ciclo de vida (Nuevo → Asignado → Corregido → Re-test → Cerrado/Reabierto). De estos campos se derivan las métricas del proceso: cobertura, tasa de aprobación, densidad de defectos, desviación de tiempos y defectos reabiertos.

Marco contextual

El trabajo se desarrolló en el marco del seminario teórico-práctico de pruebas de software de la Corporación Universitaria Remington, en el que el docente asumió el rol de cliente/product owner de una Tienda Online de ropa y accesorios: una aplicación web con catálogo alimentado por proveedores, carrito, compra con vales de descuento, pasarela de pago y notificación por correo con PDF. El insumo fue el documento 'Idea original e historias de usuario', con 14 historias del rol cliente (UH-U-01 a UH-U-14). Las pruebas se ejecutaron sobre el ambiente web proporcionado: <https://demo.prestashop.com/#/en/front> . El equipo QA Squad estuvo conformado por tres analistas —Luisa Fernanda Aristizabal Giraldo, Diego Alejandro Muñoz Robayo y Jorge Andres Ricardo Arcia— con rotación del rol de líder en cada etapa (planeación, análisis/diseño y ejecución), tal como lo propuso la metodología del seminario.

Desarrollo e implementación del aprendizaje

El aprendizaje del seminario se implementó de principio a fin sobre el proyecto Tienda Online. Esta sección presenta los resultados de cada etapa del proceso de pruebas, con los productos generados (testware): preguntas de requerimientos, plan de pruebas (v1 y v2), análisis, diseño de casos, evidencias de ejecución, Bugtracker e informe de cierre.

Aclaración de requisitos: preguntas al cliente

Antes de planear, el equipo revisó la idea original y las 14 historias de usuario y formuló 26 preguntas al cliente, agrupadas por tema (alcance, historias de usuario y gestión del plan). Las respuestas del 30 de junio se convirtieron en requisitos y restricciones del plan: el alcance cubre solo las historias del rol cliente, las ambigüedades quedaron a criterio del equipo (documentadas como supuestos) y se confirmaron cuatro reglas de negocio clave.

Regla	Respuesta del cliente	Caso que la verifica
R16 – Cambio de precio con producto en el carrito	Se factura con el precio actualizado	CP-06-06 y CP-07-07
R17 – Timeout de sesión por inactividad	No existe: la sesión permanece activa	CP-04-05
R18 – Productos en el carrito sin comprar	Pasan a la lista de deseos a las 24 horas	CP-06-07
R19 – Productos agotados	No se deberían mostrar en el catálogo	CP-01-10

Tabla 1 Reglas de negocio confirmadas por el cliente (respuestas 16 a 19).

Planeación: plan de pruebas v1 y su refinamiento a v2

El plan de pruebas se construyó bajo IEEE 829 adaptado y se refinó con la retroalimentación del cliente del 7 de julio (control de versiones: la v1 se conserva en el repositorio). El refinamiento aplicó el modelo de la triple restricción: con alcance, tiempo y recursos fijados por el cliente (14 historias; entrega del ambiente el 10 de julio y cierre el 15; 3 analistas con jornada de 8 horas), la variable que el equipo gestionó fue la profundidad del alcance mediante el análisis de riesgos. La matriz de riesgos valoró riesgos de proyecto (con planes de mitigación) y de producto (uno por historia, mitigado con un tipo de prueba). Con el criterio $P \times I \geq 15$ se delimitó el alcance de ejecución a las 7 historias de riesgo Alto, que cubren el flujo crítico completo del negocio, más 2 casos transversales: 49 casos a ejecutar en 72 horas de ejecución disponibles. La estimación total del plan fue de 118 horas (48 de preparación y 70 de ejecución) frente a 120 de capacidad.

ID	Historia	Riesgo (P×I)	Prioridad	Casos	Analista
UH-U-01	Visualización de los productos	16 – Alto	P2 – Alta	10	Jorge R.
UH-U-03	Registro	16 – Alto	P1 – Crítica	6	Diego M.
UH-U-04	Inicio de sesión	15 – Alto	P1 – Crítica	6	Diego M.
UH-U-06	Carrito	15 – Alto	P1 – Crítica	8	Luisa A.
UH-U-07	Compra	15 – Alto	P1 – Crítica	7	Luisa A.
UH-U-10	Mail (correo con PDF)	16 – Alto	P1 – Crítica	5	Jorge R.
UH-U-14	Recuperación de contraseña	20 – Alto	P2 – Alta	5	Diego M.
TRANS	Transversales (E2E y seguridad de sesión)	–	Propuesta	2	Equipo

Tabla 2 Alcance de ejecución v2: historias de riesgo Alto y casos por analista (total 49 casos).

El plan definió además la estrategia (niveles de sistema e integración de sistemas; tipos funcional, integración, E2E, exploratorio, seguridad básica y asociadas al cambio), los criterios de entrada, salida, suspensión y reanudación, el cronograma día a día (aprovechando el fin de semana como ventana natural para la prueba de las 24 horas del carrito), la gestión de defectos con severidades S1 a S4, los roles del equipo y los supuestos ante los puntos no definidos por el cliente.

Análisis de pruebas: condiciones y técnicas

Con el alcance delimitado, el equipo tomó cada historia y derivó, criterio de aceptación por criterio, las condiciones de prueba y los títulos de los posibles casos, seleccionando para cada condición la técnica de diseño más adecuada y justificando la elección. Se tuvo presente que un criterio de aceptación genera como mínimo un caso de prueba, pero normalmente varios (positivos y negativos), y que las respuestas del cliente (R16 a R19) son reglas de negocio que deben quedar cubiertas.

Técnica	Dónde se aplicó en el proyecto
Particiones de equivalencia	Correo del registro (válido, vacío, mal formado, duplicado); credenciales del login; filtros del catálogo
Valores límite	Paginación con 9, 10 y 11 productos; stock 0 y 1; totales del carrito
Tabla de decisión	Vales de descuento: estado (válido/vencido/inexistente) combinado con primer o segundo vale
Transición de estados	Carrito → lista de deseos a las 24 h (R18); sesión iniciada/cerrada; transacción de pago (aprobada/rechazada/cancelada)
Basadas en la experiencia	Sesiones exploratorias (correos no-Gmail, intentos de login), predicción de errores e inyección básica, listas de comprobación del correo/PDF

Tabla 3 Técnicas de diseño aplicadas y su justificación en el proyecto.

Diseño de los casos de prueba

Los 49 casos del alcance se diseñaron en la plantilla de Excel del curso, con una hoja por analista. Cada caso registra: identificador (CP-XX-YY), nombre y objetivo, precondiciones, pasos numerados uno a uno, resultado esperado, campos de seguimiento (resultado obtenido, estado, observaciones), tiempo estimado, día programado y técnica aplicada. Se siguieron las buenas prácticas del seminario: diseñar pensando en que otra persona ejecutará el caso (sin omitir pasos) y escribir paso a paso, de modo que cada acción pueda verificarse de forma independiente. En la práctica de ejecución los analistas intercambiaron diseños, lo que validó la claridad de los casos.

Implementación y ejecución de las pruebas

La implementación dejó el ambiente a punto: URL de pruebas (<https://demo.prestashop.com/#/en/front>), usuarios y cuentas de correo de prueba, archivo de evidencias y Bugtracker en el repositorio del equipo (carpeta de ejecución). La ejecución se realizó el 14 y 15 de julio de 2026: cada analista siguió los pasos de sus casos, tomó pantallazos personalizados (resaltando el campo o botón verificado) y documentó cada caso en el archivo de evidencias con su identificador, tal como se diseñó.

Resultados de la ejecución

De los 49 casos del alcance se ejecutaron 38 (77,6 %): 31 aprobados y 7 fallidos; los 11 restantes quedaron como 'No aplica' por limitaciones del ambiente demo (sin acceso a base de datos para preparar stock y precios, sin vales de descuento configurables y sin envío del correo de recuperación de contraseña). La Figura 1 y la Tabla 4 resumen el resultado.

Cobertura del alcance v2

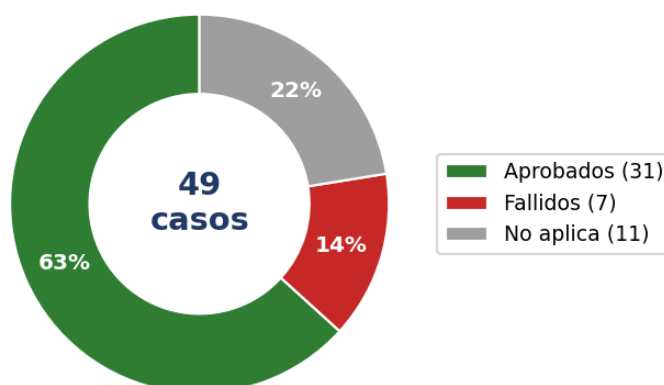


Ilustración 1 Cobertura del alcance v2: 38 casos ejecutados (31 aprobados, 7 fallidos) y 11 no aplicables.

Historia	Casos	Ejecutados	Aprobados	Fallidos	No aplica
UH-U-01	10	10	10	0	0
Catálogo					
UH-U-03	6	6	6	0	0
Registro					
UH-U-04	6	6	3	3	0
Login					
UH-U-06	8	6	4	2	2
Carrito					
UH-U-07	7	3	1	2	4
Compra					
UH-U-10	5	5	5	0	0
Mail					
UH-U-14	5	0	0	0	5
Contraseña					
Transversales	2	2	2	0	0
TOTAL	49	38	31	7	11

Tabla 4 Resultado de la ejecución por historia de usuario.

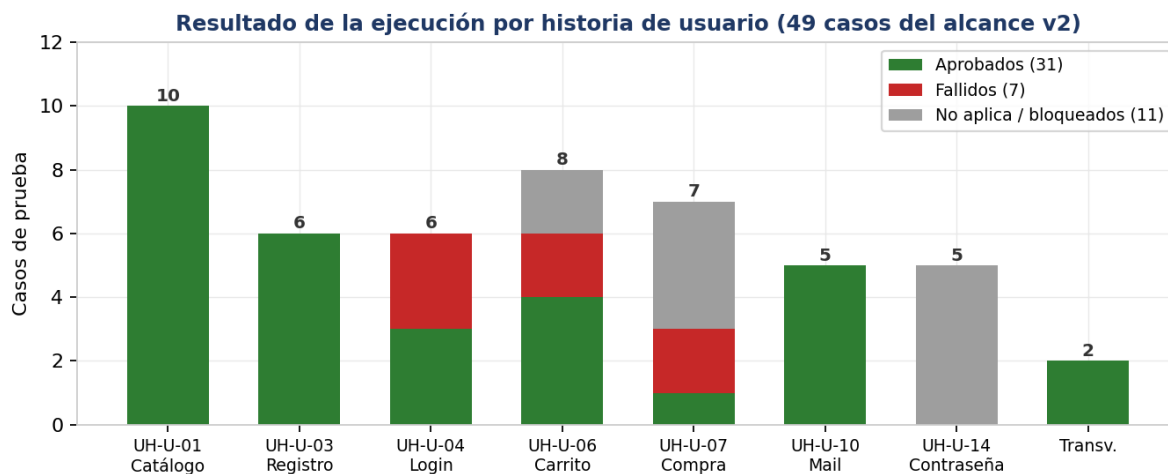


Ilustración 2 Resultado de la ejecución por historia de usuario: los fallos se concentran en login, carrito y compra.

Las evidencias completas (35 páginas de pantallazos, uno o varios por caso, con la descripción de lo verificado) quedaron consolidadas en el documento 'Evidencias de Ejecución'. Las Figuras 3 y 4 muestran dos ejemplos: una validación aprobada del registro y el defecto del carrito sin sesión.

PRESTASHOP START NOW EXPLORE BACK OFFICE

my store Clothes Accessories Art Search products...

Last name *

Only letters and the dot (.) character, followed by a space, are allowed.

Email *

The email is already used, please choose another one or sign in

Password *

Enter a password between 8 and 72 characters
The minimum score must be: Strong

Birthdate

Ilustración 3 Evidencia del caso CP-03-02 (aprobado): el sistema rechaza un correo ya registrado con mensaje claro.

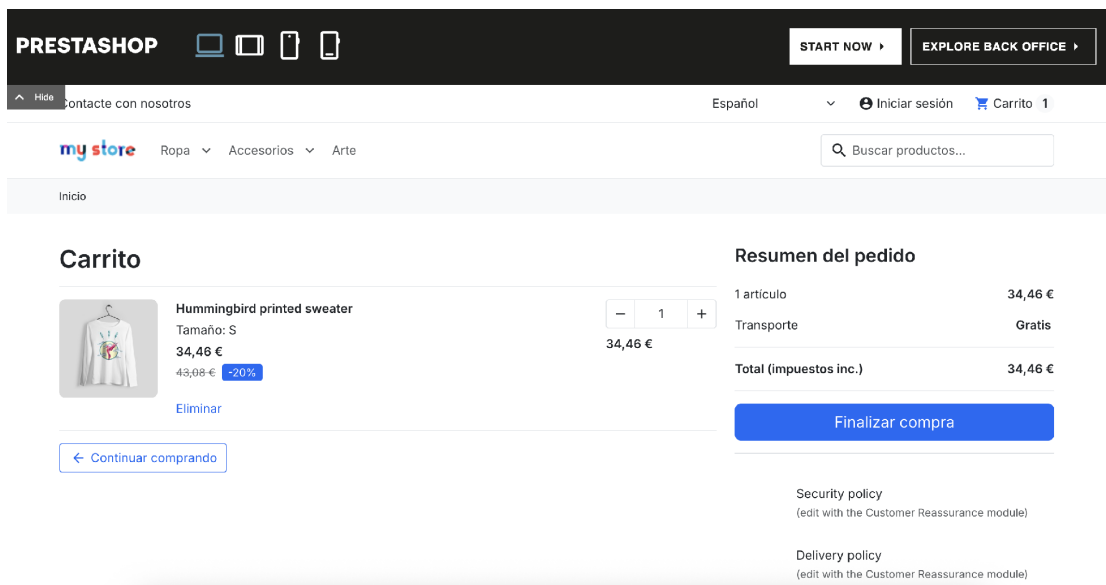


Ilustración 4 . Evidencia del caso CP-06-01 (fallido): se añade un producto al carrito sin haber iniciado sesión.

Gestión de defectos: Bugtracker

Cada ejecución fallida generó su reporte en el Bugtracker con impacto, prioridad, tipo, naturaleza, pasos exactos para reproducirlo, conclusión, responsable y fechas de creación y asignación, de modo que la plantilla 'hable por sí sola' y evite reprocesos con el equipo de desarrollo. Se registraron 8 reportes: 4 errores y 4 hallazgos; 6 de los 8 son de impacto severo y prioridad urgente, y la seguridad concentra la mitad de los reportes.

BUG	Caso	Tipo	Naturaleza	Impacto	Prioridad	Reportado por
1	CP-06-01	Error	Seguridad	Medio	Alta	Diego
2	CP-06-07	Error	Funcional	Severo	Urgente	Diego
3	CP-04-01	Hallazgo	Seguridad	Severo	Urgente	Luisa
4	Página caída	Hallazgo	Performan ce	Severo	Urgente	Diego
5	CP-07-05	Hallazgo	Hardware y/o Software	Severo	Urgente	Diego
6	CP-07-06	Error	Funcional	Medio	Alta	Luisa
7	CP-04-04	Error	Seguridad	Severo	Urgente	Luisa
8	CP-04-05	Hallazgo	Seguridad	Severo	Urgente	Luisa

Tabla 5 Reportes registrados en el Bugtracker durante la ejecución (14 y 15 de julio de 2026).

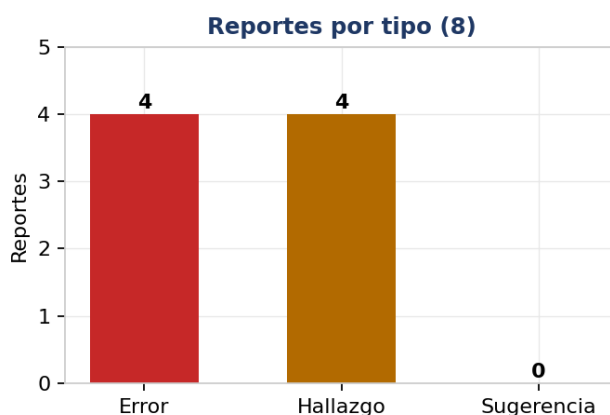


Ilustración 5 Reportes por tipo: la mitad fueron hallazgos, lo que confirma el valor de las pruebas exploratorias.

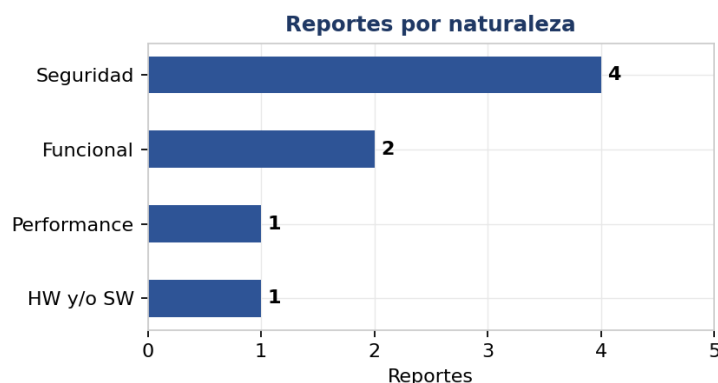


Ilustración 6 Reportes por naturaleza: la seguridad concentra 4 de los 8 defectos.

Los defectos más críticos comprometen la seguridad y el dinero del negocio: el sistema no conserva los usuarios registrados (BUG 3), las URLs privadas no redirigen al login (BUG 7), el carrito funciona sin sesión (BUG 1), una tarjeta rechazada permite completar la compra (BUG 5) y no existe opción de cancelar el pago en curso (BUG 6). Adicionalmente el ambiente presentó caídas aleatorias con pérdida de datos (BUG 4).

Seguimiento, control e informe de cierre

Durante la ejecución, la líder consolidó al cierre de cada día los casos ejecutados/aprobados/fallidos y los defectos por severidad, compartiendo el avance con el cliente. Al finalizar, el equipo elaboró el informe de cierre con los seis componentes del seminario: alcance (estimado vs. real, con las diferencias sustentadas), riesgos (cuáles se materializaron, qué ocasionaron y la efectividad de las contingencias), tiempos (estimados vs. reales con desviaciones sustentadas), métricas (con gráficas y análisis), lecciones aprendidas y oportunidades de mejora, cerrando con la recomendación del estado de salida.

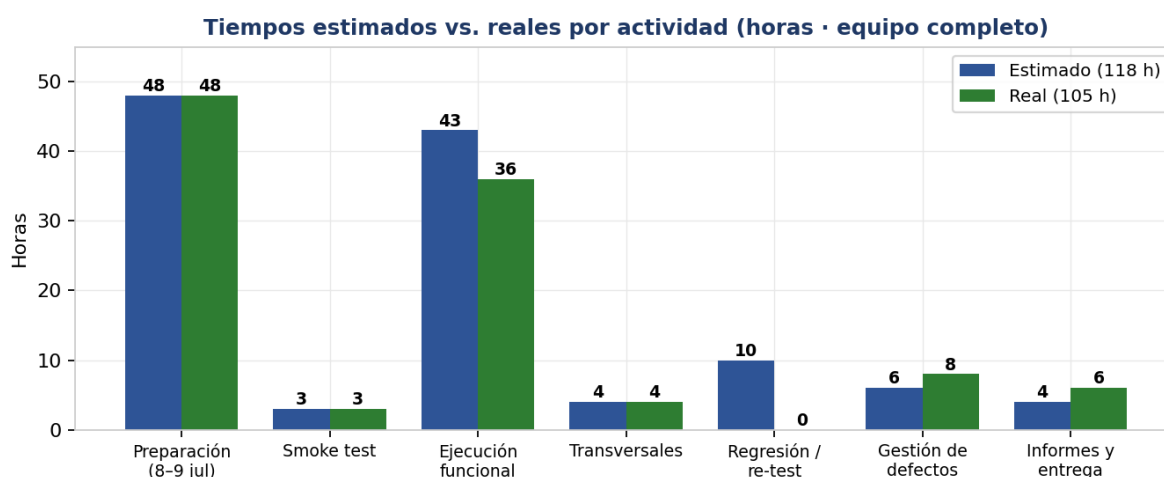


Ilustración 7 Tiempos estimados vs. reales por actividad: 105 h reales frente a 118 h planificadas (-11 %).

Entre los resultados del cierre se destacan: el criterio de salida del plan no se cumplió (quedan 6 defectos severos abiertos), por lo que la recomendación fue 'no apto para producción' hasta corregirlos y ejecutar el ciclo de confirmación y regresión; el riesgo más alto de la matriz (autenticación/seguridad) efectivamente se materializó y concentró los defectos, validando la estrategia basada en riesgo; y la desviación de tiempos fue favorable (-11 %) aunque refleja

también alcance no ejecutable en el ambiente demo. Como lecciones aprendidas quedaron el valor de diseñar pensando en quien ejecuta, el plan como documento vivo y la necesidad de verificar los prerequisites técnicos del ambiente; como oportunidades de mejora, la automatización de la regresión y el E2E, el acuerdo de niveles de servicio con desarrollo, un ambiente de pruebas con acceso a datos y el registro de horas reales desde el primer día.

Conclusiones

La puesta en práctica de la teoría del seminario sobre un proyecto real de principio a fin nos deja las siguientes conclusiones:

El proceso de pruebas es mucho más que ejecutar casos: las etapas de aclaración de requisitos, planeación y análisis determinaron la viabilidad del proyecto. Las 26 preguntas al cliente y la matriz de riesgos nos permitieron recortar el alcance con criterio técnico (7 historias de riesgo Alto) y aun así cubrir el flujo completo del negocio dentro de las 72 horas disponibles.

La estrategia basada en riesgo quedó validada con datos: las historias con mayor Probabilidad $\times$ Impacto (login, carrito, compra) fueron exactamente las que concentraron los 7 casos fallidos y la mayoría de los 8 defectos reportados. Priorizar por riesgo dirige el esfuerzo a donde están los problemas.

Las técnicas de diseño de caja negra demostraron su relación costo-beneficio: con particiones de equivalencia y valores límite, pocos casos dieron cobertura a infinitas entradas posibles, y el paso a paso en el diseño permitió que otro analista ejecutara los casos sin necesidad de preguntar.

La calidad del reporte de defectos es tan importante como encontrarlos: registrar impacto, prioridad, tipo, naturaleza y pasos exactos hace que el Bugtracker 'hable por sí solo', evita reprocesos con desarrollo y es la fuente de las métricas del cierre.

Las métricas convierten el trabajo de pruebas en información para decidir: la cobertura (77,6 %), la tasa de aprobación (81,6 %) y los 6 defectos severos abiertos sustentaron la recomendación de no aptitud para producción; sin métricas, esa recomendación sería solo una opinión.

El ambiente de pruebas es un prerequisite crítico: las limitaciones del demo público (sin base de datos, sin correos de recuperación, sin vales) impidieron ejecutar el 22,4 % del alcance.

Aprendimos que el smoke test debe validar las capacidades que el diseño requiere, no solo el flujo feliz.

Como profesionales en formación de Ingeniería de Sistemas, este seminario nos mostró un campo de aplicación directa: los roles de analista y líder de pruebas, el trabajo con estándares (IEEE 829, ISTQB, ISO/IEC 25010) y una disciplina que aplicaremos en nuestros proyectos, con la automatización de pruebas como el siguiente paso natural de profundización.

Referencias

- IEEE. (2008). IEEE Std 829-2008: IEEE Standard for Software and System Test Documentation. Institute of Electrical and Electronics Engineers.
<https://doi.org/10.1109/IEEESTD.2008.4578383>
- ISTQB. (2023). Certified Tester Foundation Level (CTFL) Syllabus v4.0. International Software Testing Qualifications Board. Recuperado el 22 de julio de 2026, de <https://www.istqb.org>
- ISO/IEC. (2011). ISO/IEC 25010:2011 – Systems and software engineering: Systems and software Quality Requirements and Evaluation (SQuaRE). International Organization for Standardization. <https://www.iso.org/standard/35733.html>
- Myers, G. J., Sandler, C., & Badgett, T. (2011). The Art of Software Testing (3.^a ed.). John Wiley & Sons.
- PrestaShop. (2026). PrestaShop Live Demo [Ambiente de pruebas]. Recuperado el 22 de julio de 2026, de <https://demo.prestashop.com/#/en/front>
- BrowserStack. (2024, 18 de septiembre). 60 Test Cases for Ecommerce Website. Recuperado el 22 de julio de 2026, de <https://www.browserstack.com/guide/test-cases-for-ecommerce-website>
- Tuskr. (s. f.). Writing Test Cases for E-commerce Websites: From Browsing to Buying. Recuperado el 22 de julio de 2026, de <https://tuskr.app/article/writing-test-cases-for-e-commerce-websites>