

TRABAJO DE GRADO

Opción Seminario-Diplomado.

Verificación del flujo crítico de compra de una tienda online en PrestaShop mediante un proceso completo de pruebas funcionales basado en riesgo, con automatización en Selenium y ejecución manual.

Corporación Universitaria Remington.

Facultad de Ingenierías

Ingeniería de Sistemas

Alejandro Sanchez Acevedo

Luis Eduardo Salazar Torreglosa

Wilker Julian Botello Medina

Tutor: Fabio Leon Grisales Morales

Seminario de Pruebas de Software

2026

Tabla de Contenidos

Resumen.....	5
Marco conceptual y contextual	7
1.1 Conceptos del seminario aplicados	7
1.2 Contexto del trabajo.....	10
Desarrollo e implementación del aprendizaje	12
2.1 Etapa de planeación	12
2.2 Etapa de análisis y diseño	16
2.3 Etapa de ejecución	23
2.4 Etapa de finalización y cierre.....	29
2.5 Comparación con otros ejercicios y buenas prácticas.....	35
Conclusiones	36
Referencias.....	38

Lista de tablas

Tabla 1 Matriz de riesgos de proyecto	13
Tabla 2 Matriz de riesgos de producto y selección del alcance (por historia)	14
Tabla 3 Historias dentro del alcance de la entrega.....	15
Tabla 4 Estimación del esfuerzo (horas).....	15
Tabla 5 Cronograma del proyecto.....	15
Tabla 6 Reglas de negocio confirmadas con el cliente	17
Tabla 7 Condiciones de prueba identificadas por historia de usuario.....	17
Tabla 8 Casos de prueba diseñados.....	20
Tabla 9 Matriz de trazabilidad (condición → caso).....	22
Tabla 10 Técnicas de diseño aplicadas	22
Tabla 11 Resultado de la ejecución de los 21 casos	23
Tabla 12 Hallazgos registrados en el bugtracker	26
Tabla 13 Comparación de tiempos por actividad (estimado vs. real).....	30
Tabla 14 Resumen de métricas del proceso	31
Tabla 15 Cumplimiento de los criterios de cierre	31
Tabla 16 Resultado de la gestión de riesgos	32
Tabla 17 Lecciones aprendidas y oportunidades de mejora (selección).....	34

Lista de figuras

Figura 1 Resultado de los casos de prueba (aprobados vs. hallazgos)	27
Figura 2 Hallazgos por naturaleza	28
Figura 3 Hallazgos por historia de usuario	28
Figura 4 Esfuerzo estimado vs. real por actividad.....	30
Figura 5 Tablero de indicadores del proceso de pruebas	33

Resumen

El presente trabajo de grado documenta la aplicación práctica de los conocimientos del Seminario de Pruebas de Software sobre un caso real: la verificación funcional de una tienda online de ropa y accesorios implementada en PrestaShop y entregada por el cliente. El ejercicio tuvo carácter teórico-práctico, en el que los fundamentos del seminario, planeación, análisis y diseño de casos, técnicas de caja negra, gestión de riesgos, métricas y criterios de cierre se ejecutaron sobre un producto operativo.

Ante una ventana de ejecución de aproximadamente dos días hábiles, el equipo Risk Tester adoptó un enfoque basado en riesgo: en lugar de cubrir superficialmente las catorce historias de usuario del producto, concentró el esfuerzo en el flujo crítico de compra, conformado por cuatro historias (Registro, Inicio de sesión, Carrito y Compra). Del análisis se derivaron veintidós condiciones de prueba, cubiertas mediante veintiún casos con trazabilidad completa entre condición, caso, ejecución y resultado.

La ejecución combinó automatización con Selenium sobre un entorno Kali Linux con navegador Chromium y pruebas manuales. De los veintiún casos, catorce resultaron satisfactorios (100 % de los técnicamente verificables) y siete se registraron como hallazgos, discrepancias entre la implementación entregada y las reglas de negocio confirmadas, clasificadas por naturaleza en seguridad, funcional y parametrización, sin identificarse defectos de código. El proceso se completó en treinta y una horas frente a cuarenta y tres estimadas (-27,91 %), dentro del plazo.

El trabajo recorrió el ciclo completo del proceso de pruebas: planeación, análisis y diseño, ejecución y cierre, este último formalizado en un informe que compara lo planeado con lo ejecutado, calcula métricas de tiempo, alcance, casos y hallazgos, y documenta lecciones aprendidas y oportunidades de mejora. Los resultados evidencian que el flujo crítico de compra opera conforme a lo esperado en todo lo verificable y que una metodología ordenada permite resultados confiables y trazables aún bajo restricción de tiempo. El principal aporte fue comprobar, en la práctica, el valor de priorizar por riesgo y de mantener la honestidad técnica en el reporte de resultados.

Palabras clave: Proceso de pruebas de software; Enfoque basado en riesgo; Pruebas funcionales de caja negra; Automatización con Selenium; Aseguramiento de la calidad.

Marco conceptual y contextual

La calidad de un producto de software puede entenderse como el grado en que este satisface las necesidades y expectativas para las que fue creado, cumpliendo sus requisitos funcionales y de negocio (Pressman y Maxim, 2015). Las pruebas de software son el conjunto de actividades orientadas a evaluar esa calidad y a detectar diferencias entre el comportamiento real del sistema y el esperado. Su propósito no se limita a encontrar errores, sino a prevenirlos mediante un proceso ordenado y sistemático que aporte valor al desarrollo (International Software Testing Qualifications Board [ISTQB], 2018). El Seminario de Pruebas de Software abordó este campo desde su planeación y gestión, y este trabajo aplica esos fundamentos sobre un caso real, por lo que constituye un ejercicio teórico-práctico: cada concepto estudiado en el seminario se ejecutó y verificó sobre un producto operativo entregado al cliente.

1.1 Conceptos del seminario aplicados

Proceso de pruebas.

Las pruebas no son una actividad aislada, sino un proceso con etapas definidas. El proceso de prueba fundamental descrito por el ISTQB (2018) comprende la planificación, el análisis, el diseño, la implementación, la ejecución y el cierre. Este trabajo organizó su ejecución en cuatro etapas coherentes con ese modelo: planeación, análisis y diseño, ejecución y finalización/cierre, cada una con entregables concretos y trazables entre sí.

Planeación y triple restricción.

Planear las pruebas implica definir objetivo, alcance, estrategia, estimación y cronograma bajo la tensión de tres variables: alcance, tiempo y recursos. Cuando una de ellas se vuelve rígida, otra debe ceder; en este proyecto, el tiempo fue la restricción dura y el alcance fue la variable que se ajustó.

Enfoque basado en riesgo.

Consiste en priorizar las funcionalidades según su valor de negocio y su probabilidad de falla, concentrando el esfuerzo donde un error tendría mayor impacto (ISTQB, 2018). Se apoya en una matriz de riesgos de proyecto y de producto y fue el eje de la estrategia adoptada.

Pruebas funcionales de caja negra.

Verifican que el sistema cumpla su comportamiento esperado a partir de sus entradas y salidas, sin conocer su estructura interna (Myers et al., 2012). Se aplicaron técnicas de partición de equivalencia y análisis de valores límite, con casos positivos y negativos.

Niveles y tipos de prueba.

Los niveles organizan la verificación según el alcance del elemento evaluado: integración, sistema y aceptación (ISTQB, 2018). En este trabajo predominaron las pruebas funcionales en esos tres niveles, con usabilidad ligera como complemento.

Historias de usuario y condiciones de prueba.

Una historia de usuario describe una funcionalidad desde la perspectiva del usuario y define sus criterios de aceptación. De su análisis se derivan las condiciones de prueba y, a partir de ellas, los casos de prueba. En este trabajo, 22 condiciones dieron origen a 21 casos.

Reglas de negocio.

Son las condiciones que el sistema debe cumplir según el cliente. Confirmarlas antes de diseñar convierte un supuesto en un criterio de aceptación verificable, y constituye la base contra la cual se contrastan los resultados.

Trazabilidad.

Es la relación demostrable entre condiciones, casos, ejecución y hallazgos (ISO/IEC/IEEE, 2013). Permite comprobar que todo lo definido se probó y facilita la auditoría del proceso.

Evidencia de prueba.

Es el registro objetivo que respalda el resultado de cada caso, capturas, registros o salidas del sistema. Sostiene la trazabilidad y hace auditable el proceso.

Automatización de pruebas.

Consiste en ejecutar casos mediante software en lugar de forma manual. En este proyecto se usó Selenium (The Selenium Project, s.f.) para los casos estables y ejecución manual para los de comportamiento variable, en un modelo mixto que optimizó la ventana de tiempo.

Defecto frente a hallazgo.

Un defecto es un error en el código; un hallazgo, en este trabajo, es una discrepancia entre la implementación entregada y una regla de negocio confirmada, o una funcionalidad no evidenciada. Esta distinción es esencial para reportar con honestidad técnica.

Métricas y criterios de cierre.

Las métricas cuantifican el proceso: cobertura, aprobación, desviación de tiempos, distribución de hallazgos y los criterios de entrada y salida determinan cuándo iniciar y cerrar el proceso, comparando lo obtenido con lo planeado (ISTQB, 2018).

1.2 Contexto del trabajo

El trabajo se enmarca en el dominio del comercio electrónico (e-commerce), entendido como la compra y venta de productos o servicios a través de internet. El producto evaluado es una tienda online de ropa y accesorios implementada en PrestaShop, una plataforma de código abierto ampliamente utilizada para crear y gestionar tiendas de comercio electrónico.

Comprender este dominio es relevante porque el objeto de las pruebas fue el flujo mediante el cual un cliente final puede registrarse, navegar y completar una compra.

El ejercicio se desarrolló sobre un caso real entregado por el cliente: la aplicación implementada en PrestaShop y accesible mediante una URL. A diferencia de un proyecto en el que el equipo recibe y despliega el código, aquí el cliente entregó la aplicación ya implementada y operativa; el equipo Risk Tester asumió la responsabilidad de planear, diseñar, ejecutar y cerrar las pruebas funcionales del producto, tratando el ejercicio como un encargo profesional.

El producto contemplaba catorce historias de usuario, además de funcionalidades de Administrador y Proveedor. Sin embargo, el proyecto se dio bajo una restricción de tiempo marcada por el cliente, con una ventana real de ejecución de aproximadamente dos días hábiles. Esta condición, sumada a la experiencia limitada del equipo, determinó la estrategia: acotar el alcance por riesgo a las cuatro historias del flujo crítico de compra, Registro, Inicio de sesión, Carrito y Compra, la cadena mínima que demuestra que la tienda puede vender. El resto de historias y los perfiles de Administrador y Proveedor quedaron fuera del alcance de esta entrega.

En conjunto, los conceptos presentados constituyen el marco teórico que el equipo aplicó de forma práctica en el proyecto. La siguiente sección describe cómo se implementaron a lo largo de las cuatro etapas del proceso, evidenciando el carácter teórico-práctico del ejercicio.

Desarrollo e implementación del aprendizaje

Esta sección presenta los resultados obtenidos en el proyecto y describe la ejecución práctica de lo aprendido en el seminario sobre un caso real: la verificación funcional de una tienda online de ropa y accesorios entregada por el cliente. El desarrollo se organiza según las cuatro etapas que recorrió el proceso de pruebas: planeación, análisis y diseño, ejecución, y finalización o cierre, coherentes con el proceso de prueba fundamental descrito por el ISTQB (2018). En cada etapa se describe qué se hizo, cómo se aplicaron los conceptos del marco conceptual y qué resultados y entregables se produjeron, acompañando la exposición con tablas y figuras.

El hilo conductor del desarrollo es el enfoque basado en riesgo, que determinó las decisiones desde la planeación hasta el cierre. Este enfoque no es una elección arbitraria: la literatura y los estándares de pruebas lo señalan como la estrategia recomendada para priorizar y focalizar el esfuerzo cuando los recursos o el tiempo son limitados (ISO/IEC/IEEE, 2013; ISTQB, 2018).

2.1 Etapa de planeación

La planeación definió el marco del proyecto bajo el modelo de triple restricción: alcance, tiempo y recursos. El cliente entregó la aplicación ya implementada en PrestaShop y fijó un plazo ajustado: el repositorio se recibiría el 10 de julio y el proyecto debía cerrarse el 15, con un fin de semana y un festivo (13 de julio) de por medio. Esto dejó una ventana real de ejecución de aproximadamente dos días hábiles (14 y 15 de julio). En consecuencia, el tiempo se asumió

como la restricción dura y el alcance como la variable que debía ceder. La decisión estratégica central fue acotar el alcance por riesgo, apoyándose en una matriz de riesgos de proyecto y de producto (ISTQB, 2018).

Tabla 1

Matriz de riesgos de proyecto

#	Riesgo	Prob.	Impacto	Nivel	Mitigación
RP1	Retraso en la entrega del repositorio por desarrollo (posterior al 10/07)	Media	Alto	Alto	Diseño de casos y datos adelantado (7–9 jul); confirmar la fecha con desarrollo
RP2	Fallas o demoras en el despliegue del código	Alta	Alto	Alto	Preparar el entorno con anticipación; documentar los pasos; reservar el 10/07
RP3	Ventana de ejecución muy corta ($\approx$ 2 días hábiles)	Alta	Alto	Alto	Alcance reducido a 4 historias del flujo crítico; ejecución en paralelo
RP4	Experiencia limitada del equipo frente a la app real	Media	Medio	Medio	Diseño previo detallado; foco en el flujo crítico; roles claros
RP5	Datos de prueba insuficientes (usuarios, productos)	Media	Medio	Medio	Preparar el conjunto de datos como criterio de entrada, antes del 10/07
RP6	Reglas de negocio ambiguas en las historias	Baja	Alto	Medio	Aclaradas y validadas con el cliente; documentadas como reglas confirmadas

Fuente: Plan de pruebas, sección 3.1.

La matriz de producto evaluó las 14 historias del usuario según su valor de negocio y su probabilidad de defecto, para decidir cuáles entraban en esta entrega.

Tabla 2*Matriz de riesgos de producto y selección del alcance (por historia)*

HU	Título	Valor	Estim.	Impacto	Prob.	Riesgo	Alcance
UH-U-01	Visualización de productos	100	13	Alto	Alta	Alto	Fuera
UH-U-02	Visualizar un producto	100	5	Alto	Media	Alto	Fuera
UH-U-03	Registro	100	2	Alto	Baja	Medio	Incluida
UH-U-04	Inicio de sesión	100	2	Alto	Baja	Medio	Incluida
UH-U-05	Suscripción	80	3	Medio	Baja	Bajo	Fuera
UH-U-06	Carrito	100	5	Alto	Media	Alto	Incluida
UH-U-07	Compra	100	8	Alto	Alta	Alto	Incluida
UH-U-08	Confirmación de compra	100	5	Alto	Media	Alto	Fuera
UH-U-09	Resumen	100	5	Alto	Media	Alto	Fuera
UH-U-10	Mail	100	8	Alto	Alta	Alto	Fuera
UH-U-11	Ver perfil	80	2	Medio	Baja	Bajo	Fuera
UH-U-12	Editar perfil	80	2	Medio	Baja	Bajo	Fuera
UH-U-13	Histórico	80	3	Medio	Baja	Bajo	Fuera
UH-U-14	Contraseña	90	5	Alto	Media	Alto	Fuera

Fuente: Plan de pruebas, sección 3.2. Criterio: valor 100/90 → impacto Alto; 80 → Medio. Estimación 8–13 → prob.

Alta; 5 → Media; 2–3 → Baja.

Cruzando el riesgo con la criticidad para el negocio y el tiempo disponible, se priorizó el flujo crítico de compra de extremo a extremo: las cuatro historias que conforman la cadena mínima de venta, incluyendo dos habilitadores de riesgo medio y difiriendo historias de riesgo alto que no bloquean la venta.

Tabla 3*Historias dentro del alcance de la entrega*

Prioridad	HU	Título	Riesgo	Rol en el flujo
1	UH-U-03	Registro	Medio	Habilitador: sin cuenta no hay compra
2	UH-U-04	Inicio de sesión	Medio	Habilitador: sin login no se puede comprar
3	UH-U-06	Carrito	Alto	Primer paso del acto de compra
4	UH-U-07	Compra	Alto	Núcleo del negocio: concreta la venta

Fuente: Plan de pruebas, sección 4.2.

La estimación combinó métricas (puntos Fibonacci de las historias) y juicio experto, distinguiendo diseño, preparación de datos, despliegue, ejecución, gestión de defectos y cierre.

Tabla 4*Estimación del esfuerzo (horas)*

Actividad	Detalle	Esfuerzo (h)
Diseño de casos	4 historias del alcance	13
Ejecución de casos	4 historias del alcance	13
Preparación de datos de prueba	Usuarios, productos y vales de ejemplo	4
Despliegue del entorno	Montar y verificar (10/07)	6
Gestión de defectos y re prueba	Registro, comunicación, re prueba	4
Cierre e informe	Consolidación y reporte	3
Total		43

*Fuente: Plan de pruebas, sección 6.***Tabla 5***Cronograma del proyecto*

Fase	Actividad	Fechas	Esfuerzo
1. Diseño	Diseño de casos y preparación de datos	7–9 jul	~17 h

Fase	Actividad	Fechas	Esfuerzo
2. Despliegue	Recepción del repositorio, montaje y verificación	10 jul	~6 h
—	Fin de semana y festivo (sin actividad)	11–13 jul	—
3. Ejecución	Ejecución de casos y registro de resultados	14–15 jul	~13 h
4. Cierre	Reprueba, consolidación e informe	15 jul	~7 h

Fuente: Plan de pruebas, sección 7.

Entregable de la etapa: Plan de pruebas (contexto, restricciones, matriz de riesgos, alcance, estrategia, estimación, cronograma y criterios de entrada/salida).

2.2 Etapa de análisis y diseño

A partir de las cuatro historias priorizadas y de las reglas de negocio confirmadas con el cliente, se realizó el análisis que identificó las condiciones de prueba de cada historia. Una dependencia relevante es que la historia de Carrito (UH-U-06) depende de la visualización de productos (UH-U-02), fuera del alcance; por ello los productos se trataron como dato de prueba preparado. En total se identificaron 22 condiciones de prueba, cada una asociada a su regla de negocio y a la técnica de diseño de caja negra correspondiente.

El análisis partió de las reglas de negocio aclaradas y confirmadas con el cliente, que constituyen los criterios de aceptación contra los cuales se contrastan los resultados. Estas reglas son la base para distinguir, en la ejecución, un comportamiento correcto de un hallazgo.

Tabla 6*Reglas de negocio confirmadas con el cliente*

Historia	Regla de negocio confirmada
Registro / Inicio de sesión	El orden lógico es registro y luego inicio de sesión; solo se validan los campos especificados.
Inicio de sesión	Se permiten hasta tres intentos fallidos.
Carrito	El carrito se conserva durante 24 horas y permite modificar la cantidad de los productos; añadir productos exige sesión iniciada.
Compra	Se verifican los datos personales y de envío especificados, y se aplican los vales de descuento según el monto de la compra y los puntos acumulados.

*Fuente: Plan de pruebas, sección 4.5.***Tabla 7***Condiciones de prueba identificadas por historia de usuario*

#	Condición de prueba	Regla aplicable	Técnica
3.1	El sistema acepta y almacena los datos cuando todos los campos son válidos	Los datos se almacenan en BD	Partición de equivalencia
3.2	El sistema rechaza el registro con correo de formato inválido	Validación de campos	Partición de equivalencia
3.3	El sistema no permite registrar un correo ya existente	Unicidad (necesaria para el login)	Partición de equivalencia
3.4	El sistema rechaza el registro con campos obligatorios vacíos	Validación de campos	Partición de equivalencia
3.5	El sistema respeta los límites de longitud de los campos	Validación de campos	Valores límite
4.1	El sistema permite el acceso con credenciales válidas registradas	Login correcto	Partición de equivalencia

#	Condición de prueba	Regla aplicable	Técnica
4.2	El sistema rechaza el acceso con credenciales incorrectas	Validación de acceso	Partición de equivalencia
4.3	Tras un login exitoso, conserva los datos y redirige a la página principal	Confirmación de la historia	Transición de estados
4.4	El sistema controla el número de intentos fallidos antes de bloquear	Regla: hasta 3 intentos	Valores límite
4.5	Al cerrar sesión, elimina los datos y no permite volver a zonas privadas	Confirmación de la historia	Transición de estados
4.6	Gestiona los estados: sin sesión → activa → cerrada	Ciclo de sesión	Transición de estados
6.1	Solo se pueden añadir productos al carrito con sesión iniciada	Confirmación + regla (exige login)	Transición de estados
6.2	El carrito muestra la información del producto (nombre, descripción, precio)	Confirmación de la historia	Partición de equivalencia
6.3	Se puede cancelar (quitar) un producto del carrito	Confirmación de la historia	Transición de estados
6.4	El carrito calcula y muestra el total a pagar	Confirmación de la historia	Valores límite / partición
6.5	Se puede modificar la cantidad y el total refleja el cambio	Regla: permite modificar cantidad	Valores límite
6.6	El carrito se conserva durante 24 horas	Regla: conservación 24 h	Transición de estados
7.1	Permite verificar los datos personales y de envío antes de comprar	Confirmación de la historia	Partición de equivalencia

#	Condición de prueba	Regla aplicable	Técnica
7.2	Valida el monto mínimo de compra para aplicar un vale	Confirmación + regla (activación por monto)	Valores límite
7.3	Aplica el vale solo con monto mínimo y puntos suficientes	Confirmación + regla (monto + puntos)	Tabla de decisión
7.4	Maneja correctamente los vales inválidos	Regla de vales	Partición de equivalencia
7.5	El total refleja correctamente el descuento aplicado	Confirmación (efecto del vale)	Valores límite

Fuente: Análisis de pruebas, secciones 1 a 4.

De estas 22 condiciones se derivaron 21 casos de prueba. La relación no es uno a uno: el caso 06 cubre las condiciones 4.1 y 4.3, por lo que las 6 condiciones de Inicio de sesión se verifican con 5 casos.

Tabla 8*Casos de prueba diseñados*

ID	Título	HU	Cond.	Téc.	Precondición
01	Registro con datos válidos	UH-U-03	3.1	PE	Formulario accesible; el correo no existe aún
02	Registro con correo de formato inválido	UH-U-03	3.2	PE	Formulario de registro accesible
03	Registro con correo ya existente	UH-U-03	3.3	PE	Existe un usuario con un correo conocido
04	Registro con campos obligatorios vacíos	UH-U-03	3.4	PE	Formulario de registro accesible
05	Límites de longitud de los campos	UH-U-03	3.5	VL	Formulario de registro accesible
06	Inicio de sesión con credenciales válidas	UH-U-04	4.1, 4.3	PE/TE	Usuario registrado; sin sesión iniciada
07	Inicio de sesión con credenciales inválidas	UH-U-04	4.2	PE	Pantalla de login accesible
08	Límite de 3 intentos fallidos	UH-U-04	4.4	VL	Usuario registrado; sin sesión iniciada
09	Cierre de sesión y protección de zonas privadas	UH-U-04	4.5	TE	Usuario con sesión iniciada
10	Ciclo completo de estados de sesión	UH-U-04	4.6	TE	Usuario registrado; sin sesión iniciada
11	Añadir productos solo con sesión iniciada	UH-U-06	6.1	TE	Productos cargados como dato de prueba

ID	Título	HU	Cond.	Téc.	Precondición
12	Visualización de la información del producto	UH-U-06	6.2	PE	Producto con datos; sesión iniciada
13	Cancelar un producto del carrito	UH-U-06	6.3	TE	Carrito con al menos un producto
14	Cálculo del total a pagar	UH-U-06	6.4	VL/PE	Productos con precios conocidos; sesión iniciada
15	Modificar la cantidad de un producto	UH-U-06	6.5	VL	Carrito con un producto; sesión iniciada
16	Conservación del carrito durante 24 horas	UH-U-06	6.6	TE	Carrito con productos; sesión iniciada
17	Verificación de datos personales y de envío	UH-U-07	7.1	PE	Sesión iniciada y carrito con productos
18	Monto mínimo para aplicar un vale	UH-U-07	7.2	VL	Carrito con productos y datos verificados; vale de prueba
19	Aplicación del vale según monto y puntos	UH-U-07	7.3	TD	Carrito y datos verificados; vale y puntos de prueba
20	Manejo de vales inválidos	UH-U-07	7.4	PE	Carrito y datos verificados; vales vencido/usado/inexistente
21	El total refleja el descuento aplicado	UH-U-07	7.5	VL	Carrito y datos verificados; vale válido disponible

Fuente: Casos de prueba y matriz de casos, hoja «Casos de prueba». PE = Partición de equivalencia; VL = Valores límite; TD = Tabla de decisión; TE = Transición de estados.

Tabla 9*Matriz de trazabilidad (condición → caso)*

HU	Condición	Caso	HU	Condición	Caso
UH-U-03	3.1	01	UH-U-06	6.1	11
UH-U-03	3.2	02	UH-U-06	6.2	12
UH-U-03	3.3	03	UH-U-06	6.3	13
UH-U-03	3.4	04	UH-U-06	6.4	14
UH-U-03	3.5	05	UH-U-06	6.5	15
UH-U-04	4.1	06	UH-U-06	6.6	16
UH-U-04	4.2	07	UH-U-07	7.1	17
UH-U-04	4.3	06	UH-U-07	7.2	18
UH-U-04	4.4	08	UH-U-07	7.3	19
UH-U-04	4.5	09	UH-U-07	7.4	20
UH-U-04	4.6	10	UH-U-07	7.5	21

Fuente: Casos de prueba y matriz de casos, hoja «Trazabilidad». 22 condiciones cubiertas por 21 casos; cobertura 100 %.

Tabla 10*Técnicas de diseño aplicadas*

Técnica	Propósito	Ejemplo de aplicación
Partición de equivalencia	Agrupar entradas que se comportan igual	Correos y contraseñas válidos e inválidos en registro y login
Valores límite	Probar los bordes de cada rango	3 intentos de login; cantidad en el carrito; monto para vales
Tabla de decisión	Combinar condiciones y resultados	Aplicación de vales en la compra (monto + puntos)
Transición de estados	Verificar cambios de estado válidos e inválidos	Sesión (inicio/cierre); carrito (agregar/modificar/cancelar/24 h)

Fuente: Plan de pruebas, sección 5.3.

Entregable de la etapa: Documento de análisis (22 condiciones con reglas y técnicas) y Casos de prueba con matriz de trazabilidad (21 casos, cobertura 100 %).

2.3 Etapa de ejecución

La ejecución se realizó sobre la aplicación entregada por el cliente, empleando un modelo mixto: los casos estables se automatizaron y los de comportamiento variable se ejecutaron manualmente. Para la automatización se empleó Selenium 4.31 (Python) sobre Kali Linux, con el navegador Chromium 149 controlado por chromedriver 149 en modo headless, generando evidencias por cada caso. En total se ejecutaron 12 casos automatizados y 9 manuales.

De los 21 casos ejecutados, 14 resultaron satisfactorios (PASA) y 7 se registraron como hallazgos.

Tabla 11

Resultado de la ejecución de los 21 casos

ID	Título	HU	Método	Estado	Observación
01	Registro con datos válidos	UH-U-03	Automatizado	PASA	Cuenta creada y sesión iniciada
02	Registro con correo inválido	UH-U-03	Automatizado	PASA	Rechaza correo sin formato válido
03	Registro con correo duplicado	UH-U-03	Manual	PASA	«The email is already used»

ID	Título	HU	Método	Estado	Observación
04	Registro con campos vacíos	UH-U-03	Automatizado	PASA	Exige campos obligatorios
05	Límites de longitud	UH-U-03	Automatizado	PASA	Límite real: 255 caracteres
06	Login con credenciales válidas	UH-U-04	Automatizado	PASA	Redirige y mantiene la sesión
07	Login con credenciales inválidas	UH-U-04	Automatizado	PASA	«Authentication failed»
08	Límite de 3 intentos	UH-U-04	Automatizado	Hallazgo	PrestaShop no implementa el bloqueo por 3 intentos
09	Cierre de sesión + zona privada	UH-U-04	Automatizado	PASA	Redirige a login al perder la sesión
10	Ciclo de estados de sesión	UH-U-04	Automatizado	PASA	Sin sesión → activa → cerrada
11	Añadir al carrito con sesión	UH-U-06	Automatizado	Hallazgo	PrestaShop permite carrito de invitado
12	Visualización de la información del producto	UH-U-06	Automatizado	PASA	Muestra nombre, descripción y precio
13	Cancelar un producto del carrito	UH-U-06	Automatizado	PASA	El producto se elimina del carrito
14	Cálculo del total a pagar	UH-U-06	Manual	PASA	1 und €22.94 → 3 und €68.83
15	Modificar la cantidad de un producto	UH-U-06	Manual	PASA	La cantidad válida actualiza el total

ID	Título	HU	Método	Estado	Observación
16	Conservación del carrito 24 h	UH-U-06	Manual	Hallazgo	No esperable 24 h; no persiste tras logout (obs.)
17	Datos personales y de envío	UH-U-07	Manual	PASA	Valida código postal (formato NNNNN)
18	Monto mínimo para aplicar un vale	UH-U-07	Manual	Hallazgo	La demo no tiene vales configurados
19	Aplicación del vale según monto y puntos	UH-U-07	Manual	Hallazgo	Sin campo de vale ni lógica de puntos
20	Manejo de vales inválidos	UH-U-07	Manual	Hallazgo	El checkout no expone campo de código promocional
21	El total refleja el descuento aplicado	UH-U-07	Manual	Hallazgo	No hay vales aplicables en el entorno

Fuente: Reporte de ejecución con evidencias, sección de resultados.

No se identificaron defectos de código. Aplicando la distinción defecto/hallazgo del marco, los 7 hallazgos son discrepancias entre la implementación entregada y las reglas de negocio confirmadas, o funcionalidades no evidenciadas. Cada uno se registró en el bugtracker con su naturaleza, impacto y prioridad.

Tabla 12*Hallazgos registrados en el bugtracker*

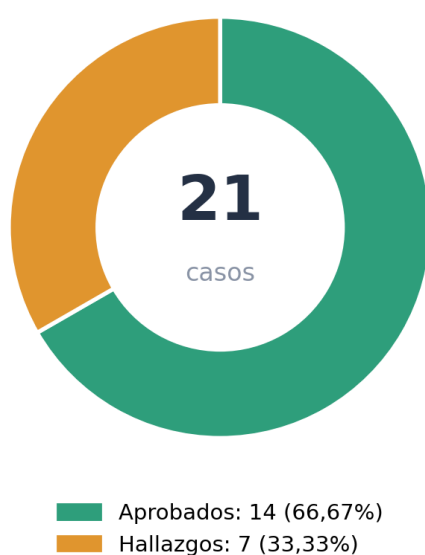
Caso	HU	Naturaleza	Impacto / Prioridad	Descripción del hallazgo
08	UH- U-04	Seguridad	Medio / Media	PrestaShop no implementa bloqueo ni verificación tras 3 intentos fallidos; permite intentos ilimitados. A validar con el cliente si la regla aplica sobre la aplicación real
11	UH- U-06	Funcional	Medio / Media	PrestaShop permite añadir al carrito sin sesión iniciada (carrito de invitado), contrario a la confirmación de UH-U-06. A validar si el comportamiento es aceptado
16	UH- U-06	Funcional	Bajo / Baja	No fue posible verificar la conservación 24 h por la duración del ciclo; además, el carrito no persiste tras cerrar sesión. A validar frente a la regla esperada
18	UH- U-07	Parametrización	Medio / Media	No fue posible verificar el monto mínimo para aplicar un vale: el entorno no tiene vales configurados
19	UH- U-07	Parametrización	Medio / Media	No fue posible verificar la aplicación del vale combinando monto y puntos: sin campo de vale ni lógica de puntos
20	UH- U-07	Parametrización	Medio / Media	No fue posible verificar el rechazo de vales inválidos: el checkout no expone campo de código promocional
21	UH- U-07	Parametrización	Medio / Media	No fue posible verificar que el total refleje el descuento del vale: no hay vales aplicables en el entorno

Fuente: Bugtracker, hoja «Bugs y otros».

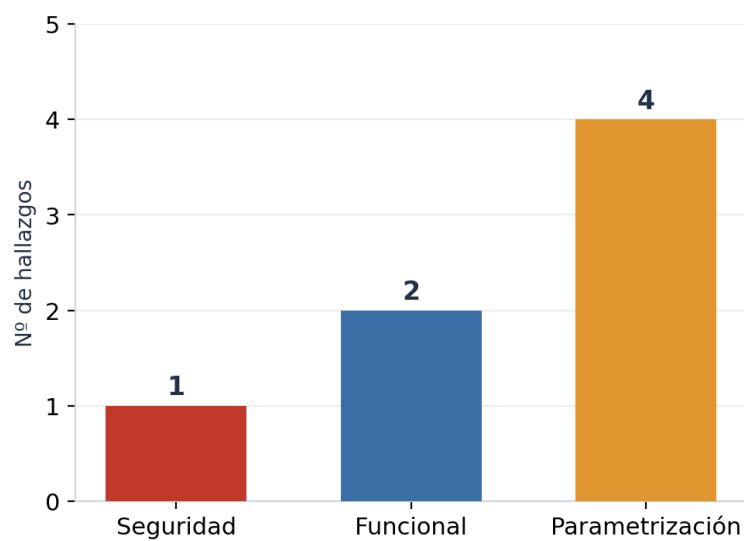
Los hallazgos se distribuyen por naturaleza en seguridad (1), funcional (2) y parametrización (4), y se concentran en Carrito y Compra, las historias de mayor riesgo previstas en la matriz.

Figura 1

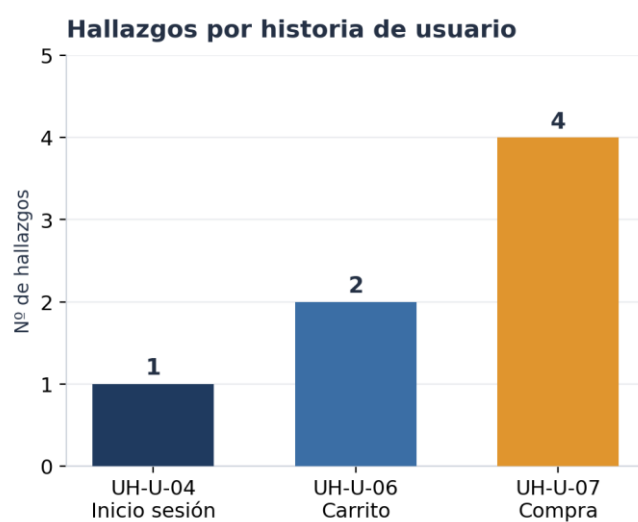
Resultado de los casos de prueba (aprobados vs. hallazgos)



Fuente: Reporte de ejecución.

Figura 2*Hallazgos por naturaleza*

Fuente: Bugtracker.

Figura 3*Hallazgos por historia de usuario*

Fuente: Bugtracker.

Durante la ejecución se presentaron situaciones reales documentadas como aprendizaje:

- Caso 05 (límites de longitud): al probar valores muy largos se descubrió el límite real del sistema, 255 caracteres en el campo de nombre, resultado característico de la técnica de valores límite.
- Caso 11 (añadir al carrito): el resultado fue «pasa con diferencia de comportamiento»: la funcionalidad opera, pero PrestaShop permite carrito de invitado, lo que difiere de la regla confirmada; por eso se registró como hallazgo.
- Caso 16 (conservación 24 h): se dejó como observación para validación con el cliente, ya que la regla no especifica el comportamiento ante el cierre de sesión, y no se clasificó como bug formal.
- Entorno técnico: se resolvió la incompatibilidad entre Google Chrome 150 y chromedriver 149 usando Chromium 149 emparejado; los ajustes quedaron documentados en la guía de automatización.

Entregable de la etapa: Reporte de ejecución con evidencias, Guía de automatización y Bugtracker (7 hallazgos clasificados).

2.4 Etapa de finalización y cierre

El cierre consolidó los resultados y comparó lo planeado con lo ejecutado, calculando las métricas del proyecto, evaluando los criterios de salida y documentando las lecciones aprendidas y las oportunidades de mejora. El proceso se completó en 31 horas frente a las 43 estimadas (-27,91 %), dentro del plazo y sin que ninguna actividad excediera su estimación.

Tabla 13*Comparación de tiempos por actividad (estimado vs. real)*

Actividad	Estimado (h)	Real (h)	Diferencia (h)	Desviación %
Diseño de casos	13	8	-5	-38,46 %
Preparación de datos	4	4	0	0,00 %
Entorno de automatización	6	1	-5	-83,33 %
Ejecución de casos	13	13	0	0,00 %
Gestión de defectos	4	2	-2	-50,00 %
Cierre e informe	3	3	0	0,00 %
Total	43	31	-12	-27,91 %

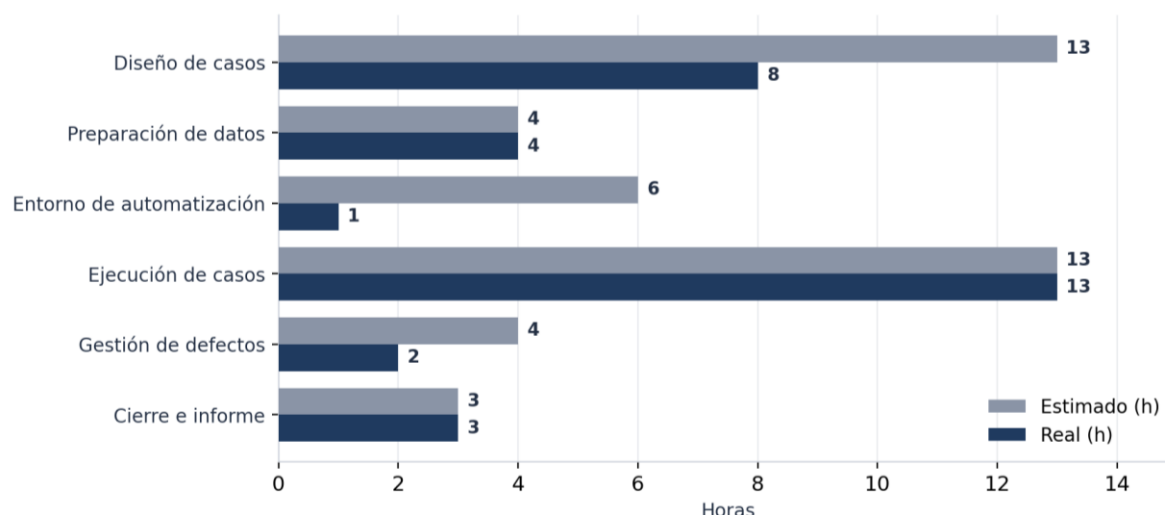
*Fuente: Informe de cierre, sección de tiempos.***Figura 4***Esfuerzo estimado vs. real por actividad**Fuente: Plan de pruebas y registro del equipo.*

Tabla 14*Resumen de métricas del proceso*

Métrica	Resultado
Cobertura del alcance (de la entrega)	4 de 4 historias (100 %)
Cobertura respecto al total del proyecto	4 de 14 historias (28,57 %)
Cobertura de condiciones de prueba	22 de 22 (100 %)
Porcentaje de ejecución de casos	21 de 21 (100 %)
Porcentaje de aprobación	14 de 21 (66,67 %)
Éxito sobre casos verificables	14 de 14 (100 %)
Tasa de hallazgos	7 de 21 (33,33 %)
Defectos de código	0 (0,00 %)
Desviación total de esfuerzo	-12 h (-27,91 %)

Fuente: Informe de cierre, sección de métricas. Cálculos con base en los datos del proyecto.

Tabla 15*Cumplimiento de los criterios de cierre*

Criterio de salida (plan)	Resultado obtenido	Cumple
Ejecución de todos los casos de las 4 historias	21 de 21 ejecutados (100 %)	Cumplido
Defectos críticos y altos corregidos y reprobados	0 defectos de código	Cumplido
Defectos medios/bajos documentados y aceptados	7 hallazgos documentados; aceptación pendiente del cliente	Cumplido parcialmente

Criterio de salida (plan)	Resultado obtenido	Cumple
Al menos 90 % de casos superados	66,67 % sobre el total; 100 % sobre lo verificable	Cumplido parcialmente
Resultados informados al cliente	Consolidados en el informe de cierre	Cumplido

Fuente: Informe de cierre, sección de criterios de cierre. Criterios de salida del Plan de pruebas, sección 5.4.

Como parte del cierre se evaluó qué ocurrió con los riesgos identificados en la planeación. Los riesgos de proyecto de mayor nivel no se materializaron —los de despliegue quedaron neutralizados porque el cliente entregó la aplicación ya implementada— y el riesgo de producto se materializó justo donde la matriz lo había previsto (Carrito y Compra).

Tabla 16

Resultado de la gestión de riesgos

Riesgo	Estado final	Observación
RP1 – Retraso en la entrega del repositorio	No materializado	El cliente entregó la aplicación ya implementada por URL
RP2 – Fallas en el despliegue del código	No materializado	No hubo despliegue de código; se neutralizó de origen
RP3 – Ventana de ejecución muy corta	Controlado	El alcance reducido y el diseño anticipado permitieron cerrar en plazo
RP4 – Experiencia limitada del equipo	Controlado	Compensado con diseño previo y foco en el flujo crítico
RP5 – Datos de prueba insuficientes	No materializado	Datos preparados como criterio de entrada

Riesgo	Estado final	Observación
RP6 – Reglas de negocio ambiguas	No materializado	Reglas aclaradas y validadas con el cliente antes de diseñar
Riesgo de producto (Carrito y Compra)	Materializado	6 de los 7 hallazgos se concentraron en estas historias, según lo previsto
Riesgo emergente – Incompatibilidad navegador/driver	Cerrado	Resuelto usando Chromium 149 emparejado con chromedriver 149

Fuente: Informe de cierre, sección de riesgos.

Figura 5

Tablero de indicadores del proceso de pruebas



Fuente: elaboración propia.

El cierre documentó las principales lecciones aprendidas por el equipo y las oportunidades de mejora derivadas del proceso, orientadas a futuros proyectos.

Tabla 17*Lecciones aprendidas y oportunidades de mejora (selección)*

Tipo	Aprendizaje / Oportunidad
Lección	El diseño anticipado de casos protege el tiempo de ejecución y permite cumplir plazos ajustados.
Lección	Confirmar la forma de entrega del producto evita sobre-estimar actividades como el despliegue.
Lección	Registrar los puntos abiertos como hallazgos trazables aporta más valor que descartarlos.
Lección	Verificar la compatibilidad navegador–driver es un requisito previo, no un detalle.
Oportunidad (proceso)	Verificar antes de ejecutar qué reglas de negocio están parametrizadas en el entorno.
Oportunidad (equipo)	Nivelar el conocimiento del equipo en automatización mediante la guía de replicación.
Oportunidad (individual)	Registrar los hallazgos en el bugtracker en el momento de identificarlos, no al final.

Fuente: Informe de cierre, secciones de lecciones aprendidas y oportunidades de mejora.

El informe de cierre documentó las lecciones aprendidas del equipo —entre ellas, que el diseño anticipado protege el tiempo de ejecución, que confirmar la forma de entrega del producto evita sobre-estimar el despliegue, y que registrar los puntos abiertos como hallazgos trazables aporta más valor que descartarlos— y las oportunidades de mejora en proceso, equipo e individuales. El flujo crítico de compra opera conforme a lo esperado en todo lo verificable; su cierre pleno depende de la validación de los hallazgos por parte del cliente.

Entregable de la etapa: Informe de cierre (comparación de tiempos, métricas, criterios de cierre, lecciones aprendidas y oportunidades de mejora).

2.5 Comparación con otros ejercicios y buenas prácticas

Los resultados del proyecto son coherentes con lo que la literatura y los estándares de pruebas de software plantean, lo que permite validar las decisiones tomadas más allá del caso particular.

En primer lugar, la decisión de acotar el alcance priorizando por riesgo ante una ventana de tiempo corta coincide con la aproximación recomendada por los estándares: la priorización basada en riesgo es descrita como el mecanismo para focalizar el esfuerzo de prueba cuando no es viable probarlo todo (ISO/IEC/IEEE, 2013; ISTQB, 2018). El proyecto aplicó exactamente ese principio al concentrar las pruebas en el flujo crítico de compra.

En segundo lugar, el diseño de casos mediante técnicas de caja negra, partición de equivalencia, valores límite, tabla de decisión y transición de estados, corresponde a las técnicas clásicas de diseño de pruebas funcionales descritas en la literatura de referencia (Myers et al., 2012). Su aplicación permitió, por ejemplo, descubrir el límite real de longitud de un campo (255 caracteres), resultado característico de la técnica de valores límite.

En tercer lugar, la distinción entre defecto y hallazgo y el registro trazable de los resultados reflejan la importancia que la ingeniería de software otorga a la calidad como proceso ordenado y verificable, y no solo a la detección de errores (Pressman y Maxim, 2015). Esta distinción fue clave para reportar los resultados con honestidad técnica.

En conjunto, el ejercicio no solo aplicó la teoría del seminario, sino que sus resultados se alinean con las buenas prácticas reconocidas en el área, lo que refuerza la validez del proceso seguido por el equipo.

Conclusiones

- El proyecto confirmó que las etapas del proceso de pruebas no son conceptos aislados, sino un flujo continuo en el que cada etapa alimenta a la siguiente. Aplicar esa secuencia sobre una aplicación real permitió comprobar que la teoría del seminario es directamente utilizable en un contexto cercano al profesional.
- Acotar el alcance a las cuatro historias del flujo crítico de compra, en lugar de cubrir superficialmente las catorce, resultó acertado: permitió probar con profundidad lo que realmente importaba dentro de una ventana de aproximadamente dos días hábiles. Los hallazgos se concentraron justo donde la matriz de riesgos los había anticipado (Carrito y Compra), lo que valida el enfoque.
- De los 21 casos diseñados y ejecutados (cobertura del 100 % de las 22 condiciones), 14 resultaron satisfactorios y 7 se registraron como hallazgos, sin identificarse defectos de código. El flujo crítico opera conforme a lo esperado en todo lo verificable; los 7 hallazgos son discrepancias con las reglas de negocio o funcionalidades no evidenciadas, pendientes de validación con el cliente.
- El proceso se completó en 31 horas frente a las 43 estimadas (-27,91 %), dentro del plazo y sin exceder ninguna actividad. Esto evidenció que un análisis y un diseño anticipados protegen el tiempo de ejecución, y que confirmar la forma de entrega del producto evita sobre-estimar tareas como el despliegue.

- Una de las lecciones más valiosas fue comprender que probar no consiste únicamente en encontrar errores, sino en documentar con rigor y honestidad lo observado. Mantener esta distinción permitió reportar resultados creíbles y auditables.
- Más allá de lo técnico, el proyecto fortaleció competencias de planeación, gestión del tiempo, trabajo en equipo y toma de decisiones bajo presión, y desarrolló un criterio más analítico para evaluar la calidad de un sistema.
- Aplicar una metodología de pruebas ordenada, priorizada por riesgo y sustentada en la honestidad técnica permite obtener resultados confiables incluso bajo restricciones de tiempo y con un producto entregado por un tercero. Con ello, el equipo Risk Tester culminó la documentación completa de un proceso de pruebas y consolidó, en la práctica, los aprendizajes del seminario.

Referencias

International Software Testing Qualifications Board. (2018). *Certified Tester Foundation Level syllabus* (Versión 2018). <https://www.istqb.org>

ISO/IEC/IEEE. (2013). *ISO/IEC/IEEE 29119-1:2013. Software and systems engineering — Software testing — Part 1: Concepts and definitions*. International Organization for Standardization.

Myers, G. J., Sandler, C., & Badgett, T. (2012). *The art of software testing* (3.^a ed.). John Wiley & Sons.

Pressman, R. S., & Maxim, B. R. (2015). *Software engineering: A practitioner's approach* (8.^a ed.). McGraw-Hill Education.

The Selenium Project. (s.f.). *Selenium documentation*. <https://www.selenium.dev/documentation/>