



PRUEBAS DE SOFTWARE NO FUNCIONAL
www.amazon.com

Presentado por:

RICARDO ENRIQUE HERRERA CASTAÑO


DINA ABISAG LOPERA

Pereira
CEAD Dosquebradas.

Medellín
CEAD - Medellín

Tutor
Manuel Antonio Sierra Rodríguez

Universidad Nacional Abierta y a Distancia UNAD
Escuela de Ciencias Básicas Tecnología e Ingeniería ECBTI
DIPLOMADO EN PROFUNDIZACIÓN EN PRUEBAS DE SOFTWARE
Tecnología en Desarrollo de Software
22/5/2020





Introducción

La evolución de los sistemas de información, los aplicativos, el software y todo aquel desarrollo en pro de un objetivo y un resultado específico, está obligado a brindar la confiabilidad y aceptabilidad que asegure su nivel de competencia y existencia.

Debido a esto se implementan ciertas facetas de planeación, diseño e implementación de pruebas. Con el fin de evaluar el estado del producto.

Aquí es donde verdaderamente se establece la calidad y continuidad de un producto que debe ser aceptado por un cliente o un ente que realiza una petición delimitada a unos requerimientos para dar una solución a un fin específico.

Objetivos

Analizar el comportamiento de uno o varios módulos de la plataforma www.amazon.com.

Diseñar un Plan de Pruebas acorde basado en pruebas funcionales y de Aceptación, delimitando alcances y límites.

Identificar, generar y diligenciar los formatos y herramientas adecuados para la ejecución de pruebas.

Dar a conocer de acuerdo con los hallazgos, recomendaciones o cambios sugeridos a dichos Módulos.

Documentar, almacenar y presentar las herramientas definidas.



Tabla de Contenidos

Pruebas de Software Aplicación www.amazon.com	5
1.1 Aspectos Preliminares de las Pruebas de Software	6
1.1.1 Objetivos de Prueba	6
1.1.2 Responsables	6
1.1.3 Métricas de Ejecución del Proyecto	7
1.2 Alcance y Limitaciones de la Prueba	7
1.2.1 Características que serán probadas	7
1.2.2 Características que no serán probadas.....	7
1.3 Pruebas de Humo	8
1.4 Metodología del Plan de Pruebas	9
1.5 Herramientas Utilizadas	10
1.6 Tipos de Pruebas Realizadas	10
1.6.1 Métricas de Ejecución de los Casos de Prueba	11
1.6.2 Métricas de Ejecución de los Defectos Encontrados.....	12
1.6.3 Parámetros	13
1.6.4 Severidad	13
1.7 Métricas de Ejecución de Defectos Proveedor.....	15
1.7.1 Severidad:	16
1.7.2 Naturaleza:	17
1.7.3 Estado	18
1.8 Evaluación del Producto y del Proveedor.....	19
Lecciones Aprendidas.....	21
Bibliografía.....	22

Índice de Tablas

Tabla 1 Planeación de Métricas por Módulos Funcionales.....	7
Tabla 2 Aspectos Revisados en la Prueba de Humo.....	8
Tabla 3 Métricas de Ejecución de los Casos de Prueba.....	11
Tabla 4 Descripción de los Defectos Encontrados	12
Tabla 5 Parámetros tenidos en Cuenta	13
Tabla 6 Severidad.....	14
Tabla 7 Métricas de Ejecución de Defectos	15
Tabla 8 Parámetros Tenidos en Cuenta.....	16
Tabla 9 Defectos de Acuerdo con la Severidad.....	16
Tabla 10 Defectos de Acuerdo con la Naturaleza	17
Tabla 11 Defectos de Acuerdo con el Estado	18
Tabla 12 Evaluación Final Funcionalidad Compras.....	19
Tabla 13 Evaluación Final Funcionalidad Autenticación.....	20

Índice de Figuras

Figura 1 Etapas del Plan de Pruebas.....	9
Figura 2 Resultado de las Pruebas por Módulo.....	11
Figura 3 Defectos por Severidad.....	14
Figura 4 Vista de Defectos por Severidad	17
Figura 5 Defectos de Acuerdo con la Naturaleza	18
Figura 6 Defectos de Acuerdo con el Estado	19



Pruebas de Software Aplicación www.amazon.com

Cuando hablamos de Pruebas de software hablamos del conjunto de mecanismos enfocados a la validación del funcionamiento de un determinado producto, en este caso Software. Para esto se utilizan diferentes tipos de pruebas de acuerdo con lo que se necesita verificar, dichas pruebas son elegidas con enfoques acordes a la planeación durante el desarrollo para establecer fallos y errores de código o estructura (Arreola Ramirez, 2018).

Estos tipos de pruebas de software están enfocados a validar diferentes aspectos del producto denominado software, algunas son implementadas durante y otras después de finalizado el desarrollo.

Existen varios tipos de pruebas, así como las funcionales utilizadas para validar el comportamiento del sistema de acuerdo con el cumplimiento de sus requerimientos. Existen las pruebas no funcionales enfocadas en el funcionamiento del sistema y su interacción con el hardware, por ejemplo. Así mismo contamos con pruebas denominada estructurales que aplican a la validación de la estructura del software (Panel Testing, 2015).

Pruebas funcionales: Este tipo de pruebas son utilizadas durante el proceso de desarrollo del software, establecen parámetros definidos donde se busca ir cumpliendo con metas haciendo así más compacto y de mejor calidad el desarrollo y en la búsqueda del cumplimiento de los objetivos.

Las pruebas funcionales están divididas por tipos, definidas según el diseño y los requerimientos del software, durante los ciclos de estas pruebas se presenta la interacción entre desarrolladores, analistas y evaluadores (ISTQB®, 2018).

Pruebas no funcionales: Es una forma de controlar la calidad de un producto de software, que permite verificar si funciona correctamente y en qué circunstancias puede fallar. Las finalidades son (Soto Morales, 2018):

“Mejora la experiencia de usuario.


Aumenta los ingresos en línea.

Utiliza métricas importantes para poner a punto tu sistema.

Determina los cuellos de botella como la configuración de base de datos.

Ayuda a establecer si una nueva versión está preparada para la producción o no.

Permite a los interesados conocer el rendimiento real en comparación con sus expectativas”.



1.1 Aspectos Preliminares de las Pruebas de Software

Este tipo de pruebas son utilizadas durante el proceso de desarrollo del software, establecen parámetros definidos donde se busca ir cumpliendo con metas haciendo así más compacto y de mejor calidad el desarrollo y en la búsqueda del cumplimiento de los objetivos.

Las pruebas funcionales están divididas por tipos, definidas según el diseño y los requerimientos del software, durante los ciclos de estas pruebas se presenta la interacción entre desarrolladores, analistas y evaluadores.

1.1.1 Objetivos de Prueba

Se establecen como objetivos de prueba los módulos correspondientes denominados Autenticación y Comprar.

Se busca realizar el análisis y definir el correcto funcionamiento de los módulos de acuerdo con las exigencias de la plataforma www.amazon.com y su modelo de negocio.

Ambos módulos son esenciales en el funcionamiento de la plataforma, son bases fundamentales, la seguridad de las transacciones realizadas por un cliente mediante su cuenta y asegurar la venta de los productos ofertados.

1.1.2 Responsables

Nombre de la Aplicación Para Evaluar:	www.amazon.com
GRUPO DE TRABAJO	
Líder	Ricardo Enrique Herrera Castaño
Probadores	Dina Abisag Lopera

1.1.3 Métricas de Ejecución del Proyecto

Tabla 1 Planeación de Métricas por Módulos Funcionales

Módulo	Fase	Fechas estimadas		Tiempo estimado de horas	Fechas reales		Tiempo real horas	% Avance
		Inicial	Final		Inicial	Final		
Autenticación	Análisis	20-03-2020	25-03-2020	7	21-03-2020	25-03-2020	6	20%
	Diseño	27-03-2020	31-03-2020	7	26-03-2020	31-03-2020	7	30%
	Ejecución	01-04-2020	02-04-2020	9	01-04-2020	01-04-2020	8	50%
Compras	Análisis	15-03-2020	19-03-2020	8	17-03-2020	21-03-2020	7	25%
	Diseño	20-03-2020	29-03/2020	12	22-03-2020	25-03-2020	6	35%
	Ejecución	30-03-2020	04-04-2020	4	28-03-2020	31-03-2020	3	40%

Fuente: Los autores.

1.2 Alcance y Limitaciones de la Prueba

1.2.1 Características que serán probadas

Modulo Comprar

- Interacción y diseño
- Búsqueda de articulo y filtros
- Agregar al Carrito

Modulo Autenticación

- Autenticación diseño
- Autenticación Usuario y Contraseña

1.2.2 Características que no serán probadas

Modulo Comprar

- No será probado El Código del aplicativo.
- No se Realizará verificación de autenticación.
- No será probado modulo diferente a comprar y carrito de compras
- No se realizará compra verídica real ni transacción electrónica.

- Las pruebas se realizarán contando con los requisitos básicos de conectividad y acceso a la red.

Modulo Autenticación

- No será probado el módulo de compra.
- No será probado el módulo de categorías
- No será probado el código fuente
- No se efectúa compra

1.3 Pruebas de Humo

Tipo de Prueba: Funcional

Nivel de Prueba: Pruebas de Aceptación

Es fundamental que los siguientes numerales se cumplan para iniciar las pruebas en una aplicación.

Tabla 2 Aspectos Revisados en la Prueba de Humo

DESCRIPCIÓN	CUMPLE	NO CUMPLE
1. El usuario puede acceder a la aplicación	SI	
2. La funcionalidad de la aplicación está disponible y accesible	SI	
3. El usuario puede realizar la navegación elemental sin que la aplicación se bloquee	SI	
4. El usuario puede navegar desde la aplicación	SI	
5. La funcionalidad requerida está disponible y su contenido es suficiente	SI	
6. El sistema es suficientemente estable y tiene suficiente funcionalidad para pasar a la fase de prueba	SI	

Fuente: los autores.

De acuerdo con la interacción, navegación y uso de la web www.amazon.com, se realizaron pruebas denominadas como de humo, evidenciando los siguientes resultados:

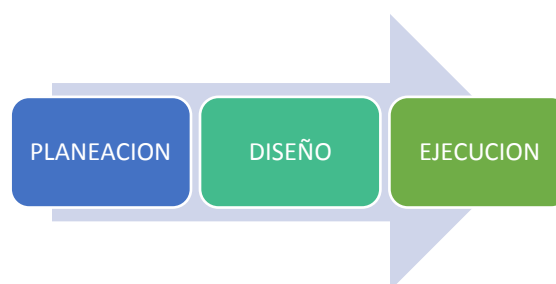
- La aplicación cuenta con un acceso inmediato escribiendo la dirección en un navegador web con acceso a internet.

- La funcionalidad de la página es correcta, adecuada y adaptada a la idea de negocio solicitada por el cliente.
- La página puede ser navegable sin bloqueos elementalmente.
- Cumple con los requisitos de navegación, cuenta con módulos específicos de acuerdo al uso de la página y su propósito inicial, contando con información relacionada y suficiente.

1.4 Metodología del Plan de Pruebas

El plan de pruebas incluye tres etapas secuenciales como se muestra en la figura 1.

Figura 1 Etapas del Plan de Pruebas



Fuente: Los autores.

Planeación

- ✓ Planteamiento del Proyecto
- ✓ Análisis y objetivos del plan de pruebas, restricciones y requerimientos
- ✓ Asignación de roles integrantes equipo de prueba

Diseño

- ✓ Diseño y priorización de pruebas.
- ✓ Diseño de plan de pruebas e inclusión de herramientas necesarias.

Ejecución

- ✓ Uso de Herramientas determinadas.
- ✓ Ejecución de pruebas diseñadas.
- ✓ Registro de versiones y hallazgos
- ✓ Documentación.

La metodología está enfocada en la correcta gestión seguimiento y control de proyectos, buscando siempre el aseguramiento de la calidad.

1.5 Herramientas Utilizadas

Las herramientas de pruebas de software son aplicaciones especializadas en análisis y validación de funcionamiento de determinados desarrollos.

Están clasificadas de acuerdo con su funcionalidad y propósito ya sea en desarrollos locales web o móviles, así mismo están disponibles de forma gratuita y de pago.

Estas aplicaciones se encargan de aplicar pruebas aleatorias, programables o automáticas sobre un proyecto o desarrollo establecido, se pueden automatizar para así ahorrar tiempo y ofrecer un testeo de calidad y generar los resultados deseados (Pandorafms, 2018) . En este ejercicio el Software de Automatización de Pruebas que se trabajó fue Selenium IDE.

Instrumentos de registro y Control de Prueba

- ✓ Plan de pruebas
- ✓ Estimación de los casos de prueba
- ✓ Propuesta de prueba
- ✓ Diseño de alto nivel
- ✓ Diseño de bajo Nivel
- ✓ Gestión de los casos de prueba
- ✓ Gestión de incidencias
- ✓ Informe de evaluación del producto

Software de Automatización de Pruebas

- ✓ Selenium IDE

Complemento con licenciamiento gratuito para el navegador Mozilla Firefox.

1.6 Tipos de Pruebas Realizadas

Pruebas Funcionales y de Aceptación.

De acuerdo con los requerimientos y solicitudes se realizaron pruebas de Tipo Funcional, donde se logre establecer la correcta funcionalidad de los módulos definidos.

Se establecieron parámetros y métricas para las pruebas determinadas en un ciclo, enfocadas en la aceptación del producto según los requerimientos previamente establecidos.

1.6.1 Métricas de Ejecución de los Casos de Prueba

Cada responsable realizó pruebas a un módulo específico y registró el numero casos exitosos, fallidos y los que no se lograron ejecutar en la tabla 3.

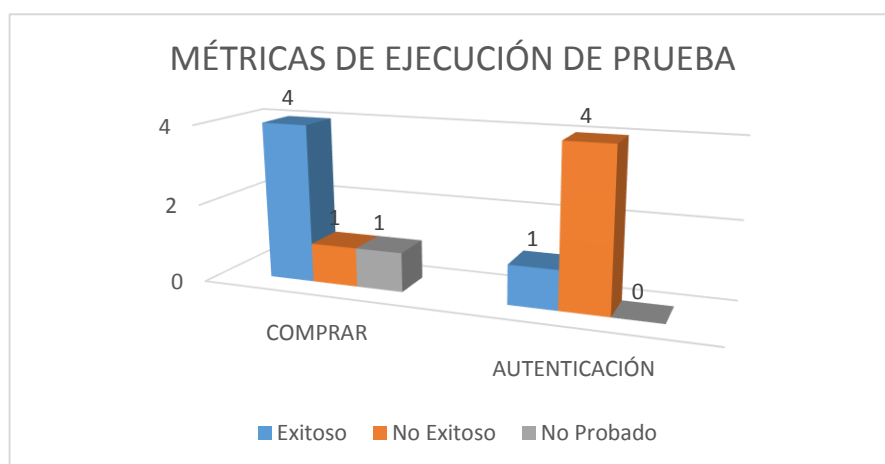
Tabla 3 Métricas de Ejecución de los Casos de Prueba

RESPONSABLE	FUNCIONALIDAD	Ciclo	Exitosos		No Exitoso		No se ejecuto		CP Totales
			#	%	#	#	#	%	
RICARDO ENRIQUE HERRERA CASTAÑO	Módulo comprar	1	4	66	1	17	1	17	6
DINA ABISAG LOPERA	Módulo autenticación	1	1	20	4	80	0	0	5

Fuente los autores.

La vista gráfica de los resultados de la prueba se puede observar en la figura 2.

Figura 2 Resultado de las Pruebas por Módulo



Fuente: los autores.

1.6.2 Métricas de Ejecución de los Defectos Encontrados

Tabla 4 Descripción de los Defectos Encontrados

Consecutivo	Fecha Detección	Responsable	Modulo / Funcionalidad	Ciclo	Caso de Prueba	Descripción Defecto	Naturaleza	Tipo Bug	Severidad	Prioridad	Estado
D001	15-04-2020	Ricardo Herrera	Compras	1	Com_Ep4_FILTERAR	Ausencia de filtros adecuados para clientes básicos o inexpertos potenciales	Ambiente	Mejora	Alta	Alta	Abierta
D001	30-04-2020	Dina Abisag Lopera	Autenticación	1	AUT_Cp01_AutenticaciónUsuario	Agregaría un captcha medida de seguridad el cual permita identificar un programa automático humano.	Funcionalidad	Sugerencia	Fatal	Urgente	Cerrado

Fuente: Los autores.

1.6.3 Parámetros

A continuación, se presentan los parámetros tenidos en cuenta para registrar la anterior matriz.

Tabla 5 Parámetros tenidos en Cuenta

Naturaleza	Tipo Bug	Estado	Prioridad	Severidad
Ambiente	Cambio/Mejora	Abierto	Alta	Alta
Datos	Consideración	Reabierto	Baja	Baja
Documentación	Defecto	No es Defecto	Normal	Media
Funcionalidad	Sugerencia	Cerrado	Urgente	Fatal
Hardware	Duplicado	Próxima Versión		
Ortografía				
Parametrización				
Presentación				
Rendimiento				
Rendimiento				
Seguridad				

Fuente: Los autores.

1.6.4 Severidad

El termino Severidad en las pruebas de Software está asociado con la criticidad y el impacto de algún error o defecto en el desarrollo y así mismo en la afectación del sistema enfocado a la funcionalidad.

La severidad esta segmentada de acuerdo con errores producidos y su impacto, existen clasificaciones como la Severidad enfocada a una acción de mejora, Severidades de tipo medio que sugieren acciones correctivas leves, así como también se encuentra la Severidad enfocada a una acción urgente denominada como critica (Peño Sánchez, 2015). En la tabla 6 se relacionan los defectos de acuerdo con la severidad.

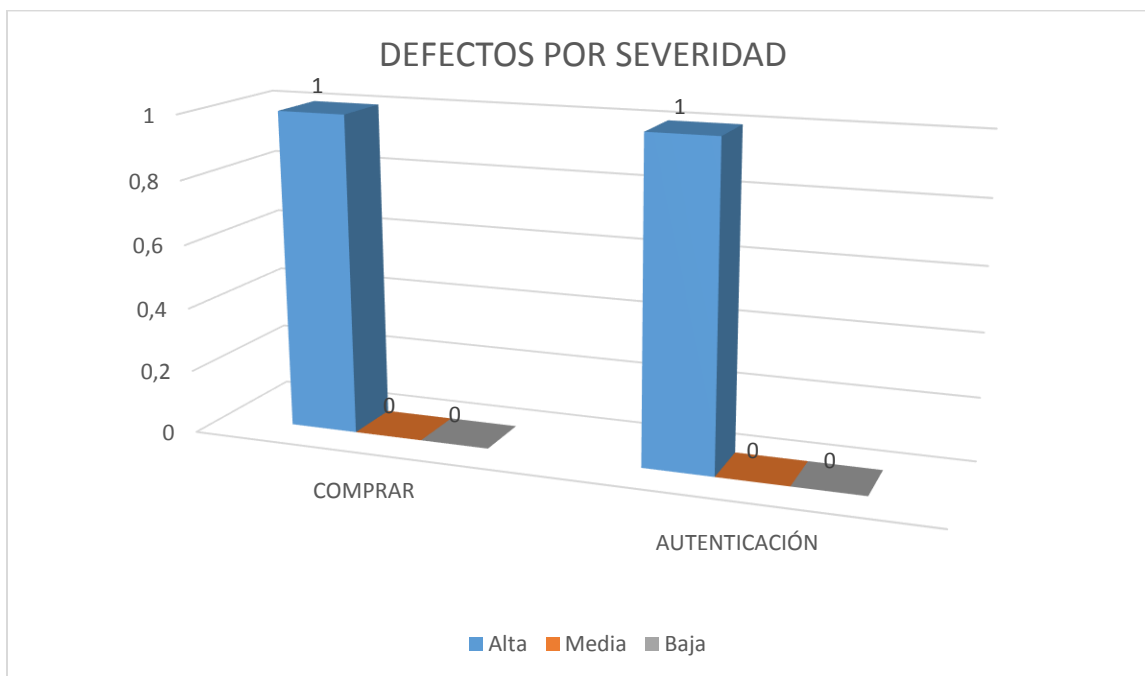
Tabla 6 Severidad

Responsable	Funcionalidad	Ciclo	Severidad		
			Alta	Media	Baja
Ricardo Herrera	Compras	1	1	0	0
Dina Abisag Lopera	Autenticación	1	1	0	0

Fuente: Los autores

En la figura 3, se representa la severidad de los módulos analizados: Compras y Autenticación.

Figura 3 Defectos por Severidad



Fuente: Los autores

1.7 Métricas de Ejecución de Defectos Proveedor

Hablamos de Métricas de Calidad de Software como los parámetros establecidos para monitorear el proceso de diseño y desarrollo, así como establecer a tiempo aspectos relevantes, de esa manera se puede establecer la calidad del producto. Están enfocadas en el análisis del código y las estructuras definidas para así lograr ofrecer una mayor comprensión del desarrollo para establecer y enfocar pruebas más acertadas y mejores soluciones. Están clasificadas por criterios como complejidad, calidad, competencia, desempeño o estilizadas, cada una con su objetivo específico (Barrientos, 2018). En la tabla 7 se muestran las métricas de ejecución de defectos.

Tabla 7 Métricas de Ejecución de Defectos

Consecutivo.	Fecha Detección	Responsable	Modulo / Funcionalidad	Ciclo	Caso de Prueba	Descripción Defecto	Naturaleza	Tipo Bug	Severidad	Prioridad	Estado
D001	22/03/2020	Ricardo Herrera	Compras	1	Com_Ep4_FILTRAR	Al momento de ejecutar el Script No se logran establecer los filtros nuevamente	Datos	Consideración	Baja	Baja	Reabierto
D001	30/04/2020	Dina Abisag Lopera	Autenticación		AUT_Cp01_AutenticaciónUsuario	Agregaría un captcha medida de seguridad el cual permita identificar un programa automático humano.	Funcionalidad	Sugerencia	Fatal	Urgente	Cerrado

Fuente: Los autores.

Parámetros: En la tabla 8, se presentan los parámetros para tener en cuenta para registrar la anterior matriz.

Tabla 8 Parámetros Tenidos en Cuenta.

Naturaleza	Tipo Bug	Estado	Prioridad	Severidad
Ambiente	Cambio/Mejora	Abierto	Alta	Alta
Datos	Consideración	Reabierto	Baja	Baja
Documentación	Defecto	No es Defecto	Normal	Media
Funcionalidad	Sugerencia	Cerrado	Urgente	Fatal
Hardware	Duplicado	Próxima Versión		
Ortografía				
Parametrización				
Presentación				
Rendimiento				
Rendimiento				
Seguridad				

Fuente: Los autores

1.7.1 Severidad:

En la siguiente tabla se relacionan los defectos de acuerdo con la severidad.

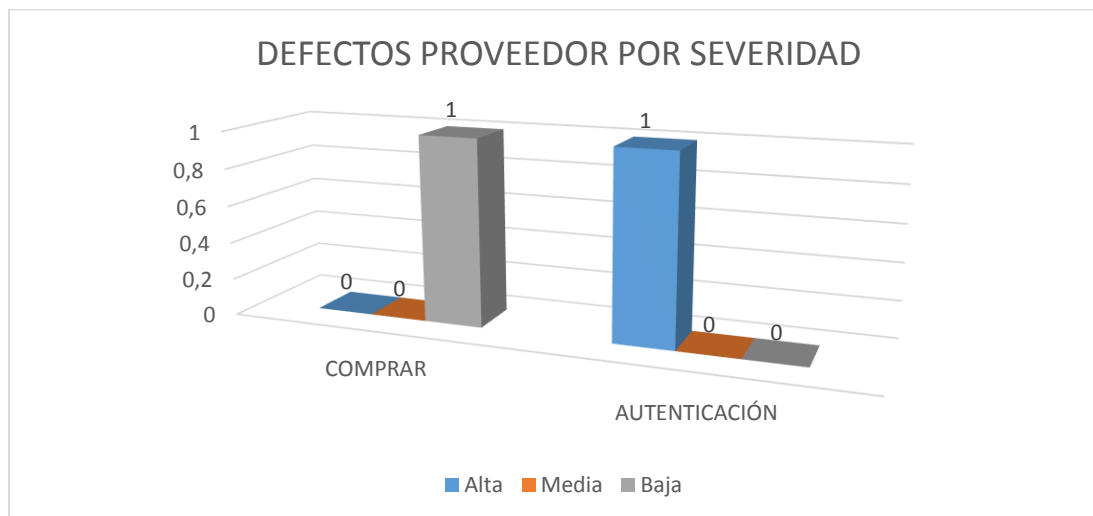
Tabla 9 Defectos de Acuerdo con la Severidad

Responsable	Funcionalidad	Ciclo	Severidad		
			Alta	Media	Baja
RICARDO E. HERRERA C.	COMPRAR	1	0	0	1
DINA ABISAG LOPERA	AUTENTICACIÓN	1	1	0	0

Fuente: Los autores

De la anterior tabla se han graficado los defectos por severidad obteniendo la figura 4.

Figura 4 Vista de Defectos por Severidad



Fuente: Los autores

1.7.2 Naturaleza:

En la siguiente tabla se relacionan los defectos de acuerdo con la naturaleza.

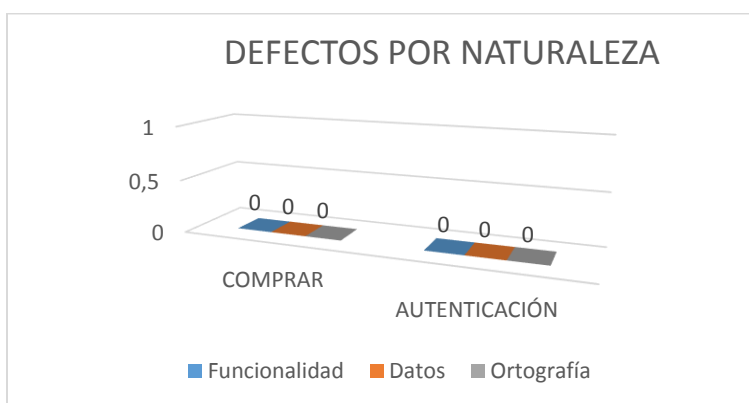
Tabla 10 Defectos de Acuerdo con la Naturaleza

Responsable	Funcionalidad	Ciclo	Naturaleza		
			Funcionalidad	Ortográfica	Se insertan los tipos de naturaleza que se hayan definido a los defectos encontrados.
RICARDO ENRIQUE HERRERA CASTAÑO	COMPRAR	1	COMPRAR	0	No Aplica
DINA ABISAG LOPERA	AUTENTICACIÓN	1	AUTENTICACIÓN	0	N/A

Fuente: Los autores

Siguiendo con la metodología, se ilustran los defectos por naturaleza, en este caso no se encontraron oportunidades y mejora en esta categoría (figura 5).

Figura 5 Defectos de Acuerdo con la Naturaleza



Fuente: Los autores

1.7.3 Estado

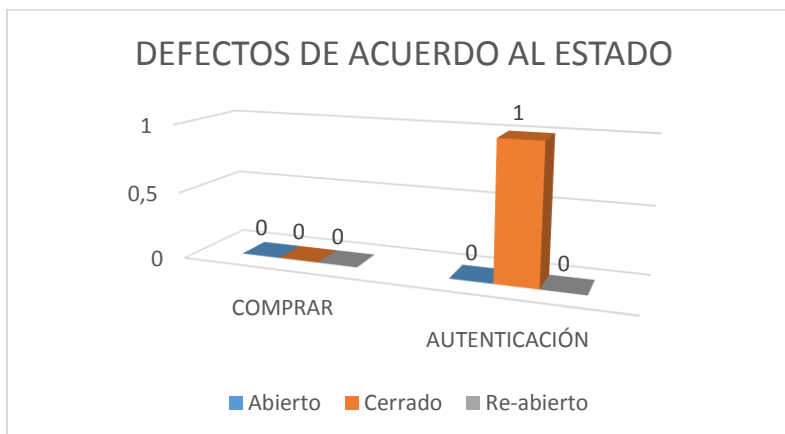
En la tabla 11 se relacionan los defectos de acuerdo con el estado.

Tabla 11 Defectos de Acuerdo con el Estado

Responsable	Funcionalidad	Ciclo	Estado		
			Abierto	Cerrado	Se insertan los estados que se hayan definido a los defectos encontrados.
RICARDO ENRIQUE HERRERA CASTAÑO	COMPRAR	1	0	0	No APLICA
DINA ABISAG LOPERA	AUTENTICACIÓN	1	0	1	N/A

Fuente: Autores.

Figura 6 Defectos de Acuerdo con el Estado



Fuente: Autores.

1.8 Evaluación del Producto y del Proveedor

La evaluación final de acuerdo con los análisis anteriores se muestra en las tablas 12 y 13.




Tabla 12 Evaluación Final Funcionalidad Compras

Funcionalidad	Casos de Prueba Ejecutados por Ciclos	Total Defectos Encontrados			Número de Ciclo	Defectos Reabiertos	Si se superan los 3 ciclos se castiga	Indicador Calidad Proveedor
		Severidad Alta	Severidad Media	Severidad Baja				
COMPRAS	6	1	0	0	1	0	1,00	0,33
							1,00	0,00
							1,00	0,00
							1,00	0,00
Total	6	1	0	0		0		0,33

Alta Calidad	≤ 0.2	
Media Calidad	>0.2 y <0.5	
Baja Calidad	>0.5	

Tabla 13 Evaluación Final Funcionalidad Autenticación


Funcionalidad	Casos de prueba ejecutados por ciclos.	Total, defectos encontrados			Número de ciclo	Defectos reabiertos	Indicador proveedor
		Severidad					
		Alta	Media	baja			
Autenticación	5	1	0	0	1	0	0.05
							0
							0
Total	5	1	0	0	1	0	0.05

Alta Calidad	≤ 0.2	
Media Calidad	> 0.2 y < 0.5	
Baja Calidad	> 0.5	



Lecciones Aprendidas

Se pueden determinar cómo lecciones aprendidas.

- Diseñar una prueba adecuada aplicable a determinado modulo.
 - Implementar con argumentos válidos dicha prueba ya diseñada.
 - Analizar y entender el desarrollo de la prueba.
 - Sistematizar pruebas mediante aplicativos o extensiones, caso Selenium.
 - Diligenciar y verificar formatos reglamentarios para implementar un seguimiento de pruebas y resultados
 - Identificar pruebas de alto, medio y bajo nivel
 - Implementar con argumentos válidos dicha prueba ya diseñada.
 - Observar y comprender la elaboración de las pruebas.
 - Plantear pruebas idóneas aplicables a un módulo.
 - Gestionar y revisar la documentación adecuada de pruebas y sus resultados
- 

Bibliografía

- Arreola Ramirez, O. A. (2018). *Cómo Ser un Tester: Introducción a Las Pruebas de Software*. (A. D. S. LLC, Ed.).
- Barrientos, D. (2018). Métricas de calidad de Software - Desafío Latam. Recuperado de <http://blog.desafiolatam.com/metricas-de-calidad-de-software/>
- ISTQB®, I. (2018). *Programa de estudio de nivel básico ISTQB*. Recuperado de <https://repository.unad.edu.co/handle/10596/26560>
- Pandorafms. (2018). Las mejores herramientas para pruebas en 2018: ¿Cuál usaremos? Recuperado de <https://pandorafms.com/blog/es/herramientas-para-pruebas/>
- Panel Testing. (2015). Software QA - ¿Cuáles son los tipos de pruebas software? Retrieved Recuperado de <https://www.panel.es/blog/software-qa-cuales-son-los-tipos-de-pruebas-software/>
- Peño Sánchez, J. M. (2015). Pruebas de Software. Fundamentos y Técnicas (+). Recuperado de <http://oa.upm.es/40012/>
- Soto Morales, V. M. (2018). Conoce qué son las pruebas no funcionales de software. Recuperado de <https://www.pragma.com.co/blog/conoce-que-son-las-pruebas-no-funcionales-de-software>