

# **Sistema de Administración de Farmacias**

## **SAF**

### **Informe Final de Verificación**

### **Versión 1.0**

### **Historia de revisiones**

Fecha	Versión	Descripción	Autor
23/11/2014	1.0	Creación inicial del documento	Juan Pablo Sierra

# Contenido

<b>1.RESUMEN DE LA VERIFICACIÓN .....</b>	<b>3</b>
1.1.PLANIFICADO VS. REALIZADO .....	3
1.2.USO DE RECURSOS HUMANOS .....	4
1.3.PRUEBAS DEL SISTEMA .....	5
1.3.1.Resumen de las pruebas .....	5
1.4.VERSIONES DEL SISTEMA.....	5
1.5.ANÁLISIS DE POLÍTICAS LLEVADAS A CABO .....	7
1.6.FUNCIONALIDADES .....	10
1.6.1.Gravedad de los errores .....	12
1.7.CIERRE DE LA VERIFICACIÓN .....	14
1.7.1.Errores conocidos.....	14
1.7.2.Nuevos errores detectados.....	14
<b>2.EVALUACIÓN .....</b>	<b>16</b>

## Resumen de la verificación

En la presente sección se incluye información respecto al funcionamiento del proceso de verificación desde el comienzo del proyecto.

### ***Planificado vs. Realizado***

Inicialmente, en la primer versión del documento de planificación de la verificación, se hacía referencia a un amplio número de pruebas distintas, siendo éstas las siguientes:

- ⑩ Prueba de integridad de la información
- ⑩ Prueba de funcionalidad
- ⑩ Prueba de ciclo de negocio
- ⑩ Prueba de interfaz de usuario
- ⑩ Prueba de carga
- ⑩ Prueba de esfuerzo
- ⑩ Prueba de volumen
- ⑩ Prueba de seguridad y control de acceso
- ⑩ Prueba de fallas y seguridad
- ⑩ Prueba de configuración
- ⑩ Prueba de instalación
- ⑩ Prueba de documentos

Eventualmente, se consideró, por distintas razones, que esa cantidad de pruebas no era factible considerarlas, tal como se refleja en la última versión del plan de verificación.

Del grupo de 12 tipos de prueba propuestos, solamente la mitad pudo llevarse a cabo. Considerando el entorno de ejecución algunos se descartaron dado que el énfasis no estaría en esos aspectos, y se optó por priorizar aquellos que el cliente habría marcado como importantes, además de también asegurar que el sistema funcione.

El principal obstáculo para la completitud de las pruebas inicialmente propuestas fue la falta de recursos humanos disponibles para llevar a cabo las tareas. En particular, muchas de las pruebas resultaban más aplicables para un producto con cierta completitud, pero en la etapa en la que se contaba con dicho producto se carecía de disponibilidad para ejecutarlas.

Adicionalmente, y a pesar de que se había considerado inicialmente en la planificación, resultaba insuficiente basar en los casos de uso los casos de prueba que regirían la verificación. El mayor número de errores no se encontraba en los aspectos funcionales, sino de diseño y usabilidad para lo cual se compuso un documento de estándares de interfaz e interacción, sirviendo de *checklist*. Esto si bien no se creó tan tempranamente como hubiera sido deseable, permitió una mejoría relevante en la calidad. De todas maneras, se

utilizaron planillas para la verificación que incluían las combinaciones deseables a verificar y los casos bordes relevantes.

Por otra parte, si bien se estimó que sería deseable contar con la mayor disponibilidad de recursos durante las etapas que lo exigieran, la realidad marcó que esas personas estaban ocupadas en otras tareas o con cuestiones ajenas al proyecto, lo que dificultó llevar a cabo el proceso de verificación tal como se había planificado. En promedio, los recursos invirtieron menos de la mitad del tiempo que se había estimado para tal fin. Este fenómeno, sumado a que los ciclos de liberación fueron en muchos casos restrictivos en términos de plazos para la evaluación del sistema, determinó que aplicando los criterios establecidos anteriormente, sólo se llevaran a cabo los tipos de verificación más importantes. En general, la verificación del sistema no vio disminuida su calidad, aunque en ocasiones fue necesario postergar algunas funcionalidades.

No se realizaron, al menos en forma extensiva, pruebas de verificación unitaria. Se estima que la carga en el equipo de desarrollo tuvo una decisiva influencia en la realización o no de la mismas.

## Uso de recursos humanos

En el presente gráfico se detalla el promedio de horas dedicadas por el equipo de verificación desglosadas por semana.

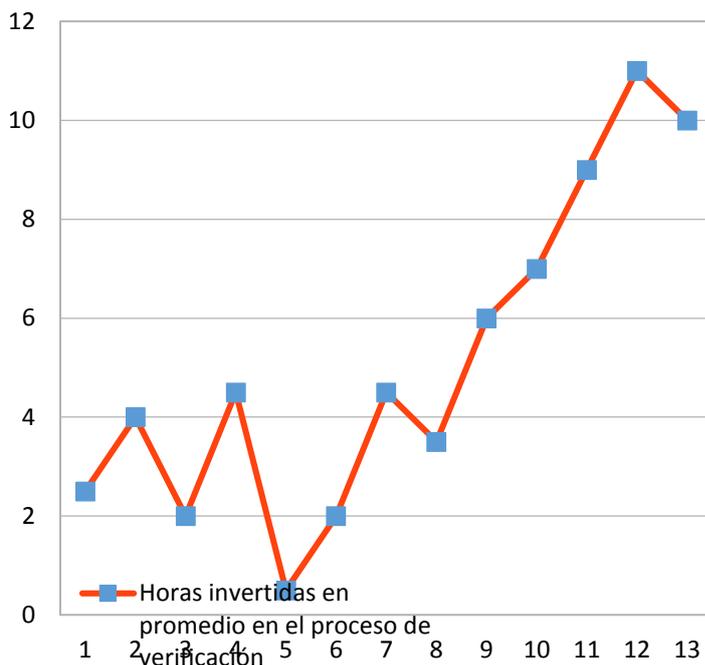


Gráfico 1: Evolución de la dedicación al proceso por parte de los verificadores

Se observa una creciente intervención de parte del equipo de verificación, aunque claramente insuficiente previo a la semana 10. La mayor dedicación de recursos se observó ya para el final del proyecto, aunque también en esa etapa la exigencia para el equipo era aún mayor. Algunos hitos pueden observarse, como el de la semana cinco y seis, que responden a problemas externos al proyecto.

## Pruebas del sistema

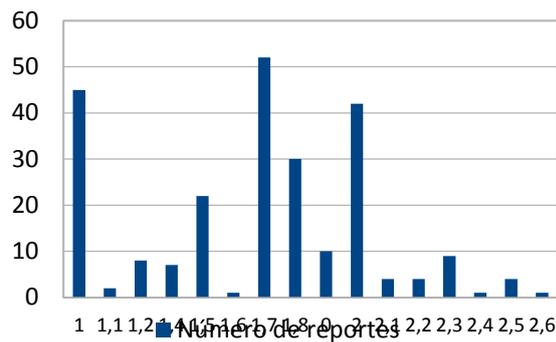
### Resumen de las pruebas

Durante el transcurso del proyecto se realizaron 22 versiones distintas del producto que fueron verificadas en distintas ocasiones y que entre éstas acumulan un total de 265 incidencias. Son 8 las personas a las que se han adjudicado errores y al final del proyecto son 15 los que resta corregir. El número de personas distintas que han reportado errores son 6.

### Versiones del sistema

A continuación se presenta una tabla con la lista de versiones distintas que fueron liberadas, acompañadas del número de incidencias encontradas y el período durante el cual se evaluaron las funcionalidades. Se incluyen versionse que han pertenecido a ramificaciones dentro del control de revisiones, por lo cual es posible que su período de verificación se encuentre compartido con otras versiones.

Versión	Cantidad de incidencias	Período de verificación
1.0	45	1/10 – 5/10
1.1	2	5/10 – 5/10
1.2	8	6/10 – 10/10
1.3	- no fue verificada -	
1.4	7	12/10 – 14/10
1.5	22	14/10 – 16/10
1.6	1	18/10
1.7	52	21/10 – 27/10
1.8	30	30/10 – 1/11
1.8.1	10	1/11
2.0	42	4/11 – 7/11
2.1	4	8/11
2.2	4	10/11
2.3	9	11/11
2.4	1	13/11
2.5	4	14/11
2.6	1	16/11
Branch susp. factura	3	8/11
Branch compras aut.	10	13/11
Sin versión especificada	10	2/10 – 13/11



de reportes por cada versión del producto

Gráfico 2: Cantidad

Se observa que las últimas versiones acumulan menos reportes, pero esto se puede explicar por los ciclos de liberaciones más cortos. Notar que en muchas de las liberaciones que se detallan, no se ha podido dedicar más de un día para su verificación dado que ya se contaba con una nueva versión con las correcciones.

En la siguiente gráfica se muestra el número de reportes por semana del proyecto.



Reportes por semana. El número de reportes es estable a partir de la semana 10

Gráfico 3:

En el gráfico anterior se observa una disminuida variabilidad, aunque a la baja, del número de reportes. Se observa un gran número de reportes inicial, comparable al momento de mayor actividad del proyecto, incluso teniendo muy pocas funcionalidades implementadas. Este período inicial fue el primer contacto entre verificadores y desarrolladores, y facilitado por la falta de estándares, se comprobó que ambas visiones eran considerablemente distintas.

La tendencia parecería indicar que posiblemente exista un número alto de incidencias aún presentes en el sistema. Si bien esto podría parecer tener sentido, habría que considerar que la detección de incidencias está fuertemente ligada a la actividad del proceso de desarrollo. Culminadas las tareas de desarrollo, sería esperable que el número de reportes comenzara a descender.

## Análisis de políticas llevadas a cabo

Durante el transcurso del proyecto se hizo un importante énfasis en mejorar cualidades importantes del producto. La usabilidad, un aspecto clave para el cliente, fue lo central durante la creación del mismo.

El principal eje de la política aplicada fue la compilación de una serie de criterios y estándares de usabilidad, que incluía además criterios de diseño e interacción con el usuario. No limitándose solamente a la usabilidad, también se consideraban aspectos de consistencia y navegación a lo largo de la aplicación. El mismo se facilitó a los desarrolladores en la semana 8.

A continuación se observa la evolución del número de errores de usabilidad y la relación respecto al total de incidencias halladas.

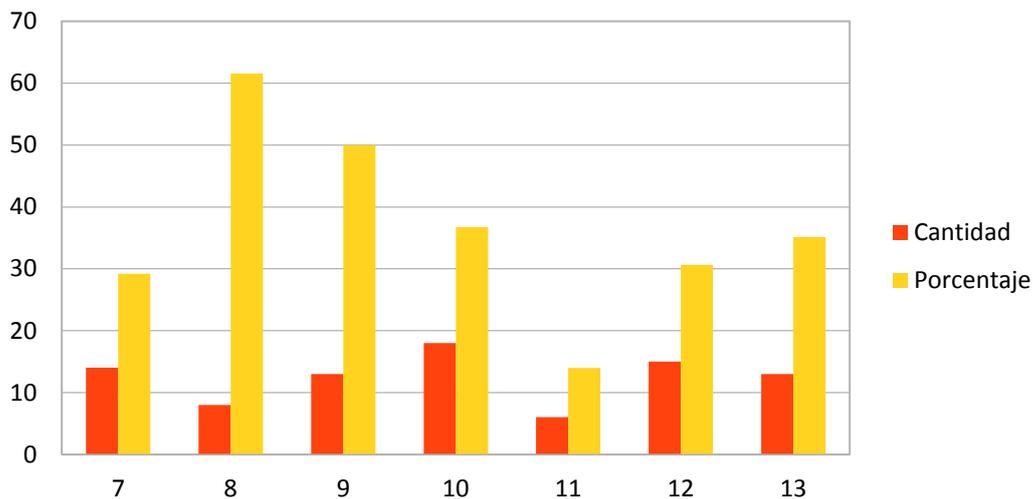


Gráfico 4: Evolución del número de errores de usabilidad y su peso relativo

Se observa una importante mejoría respecto de las primeras semanas, especialmente a partir del momento en el que se compone el documento de estándares. Las últimas dos semanas experimentan un crecimiento sostenido de este tipo de errores. La explicación para esto fue la inclusión de nuevos desarrolladores dentro del equipo que no estaban al tanto de las pautas de diseño.

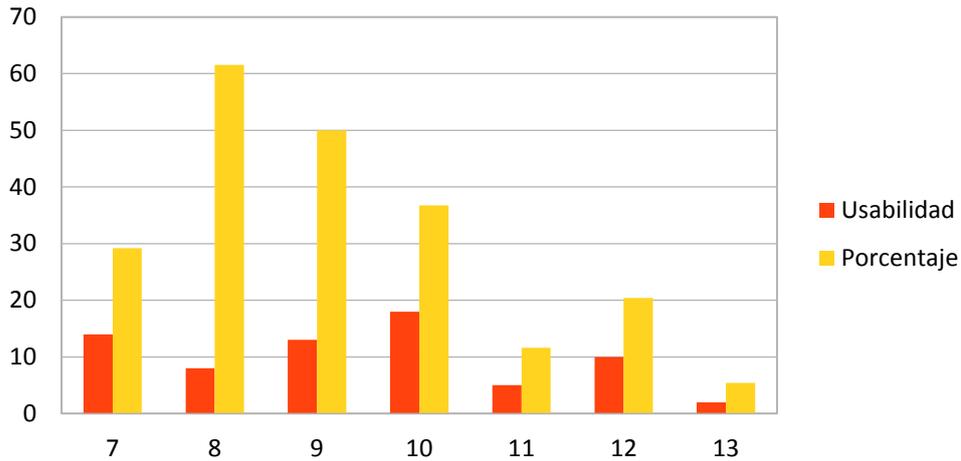


Gráfico 5:

*Estudio de usabilidad, sólo desarrolladores originales*

En la gráfica anterior, que solamente considera los desarrolladores originales, se observa un sostenido descenso en las cifras de reportes de usabilidad, por lo que se evidencia claramente que las políticas llevadas a cabo generaron el efecto deseado.

Otro aspecto importante sobre el cual se invirtió mucha atención fue en el caso de la consistencia. En la siguiente tabla se observa el mismo estudio pero realizado para los reportes de consistencia.

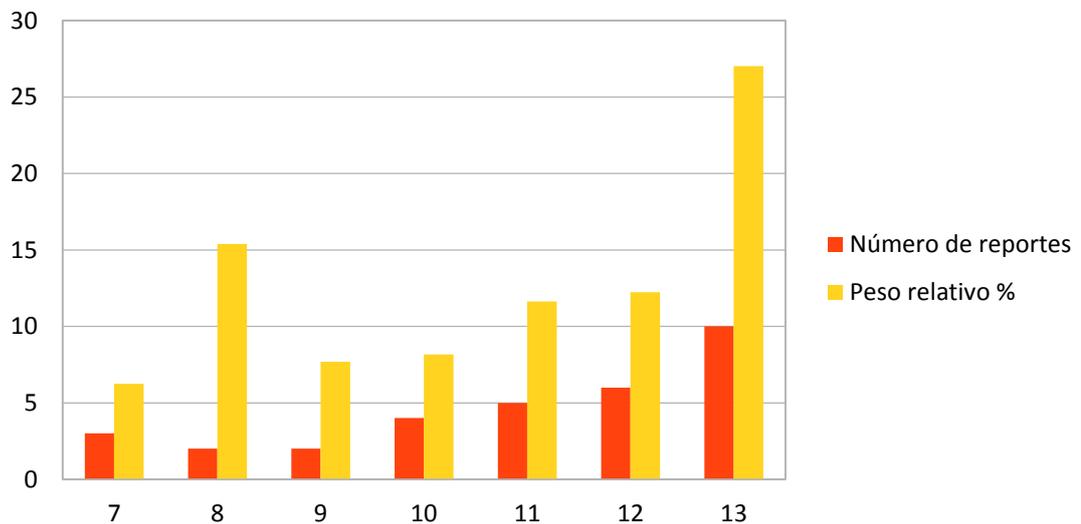


Gráfico 6: *Número de errores de consistencia y su peso relativo*

En este caso se observa que existió un importante aumento de los errores de consistencia en la última semana, que se explica porque los casos de uso que se liberaron en ese último período correspondían a desarrolladores que no estaban familiarizados con los estándares. De todas maneras, hay que considerar que la inclusión de nuevas funcionalidades siempre supone un

incremento en la dificultad para mantener los niveles de consistencia aceptables, por lo que aunque el peso relativo de este tipo de errores no haya aumentado sensiblemente durante el proyecto implica que existió cierto éxito en la política aplicada.

## **Funcionalidades**

Durante el proyecto se realizaron las estimaciones de qué funcionalidades serían las más prioritarias, en base a varios criterios como la necesidad de validación temprana y complejidad de los casos de uso.

Se reconocieron los siguientes grupos de funcionalidad:

- ⑩ Clientes
- ⑩ Compras
- ⑩ Facturación
- ⑩ General
- ⑩ Gestión de Stock
- ⑩ Productos
- ⑩ Seguridad
- ⑩ Sincronización

Finalizado el proceso de verificación, se obtienen los siguientes datos de errores para cada grupo:

<b>Grupo de funcionalidad</b>	<b>Número de incidencias</b>
Clientes	48
Compras	10
Facturación	107
General	21
Gestión de Stock	25
Productos	34
Seguridad	13
Sincronización	7

Estos datos se pueden visualizar en el siguiente gráfico.

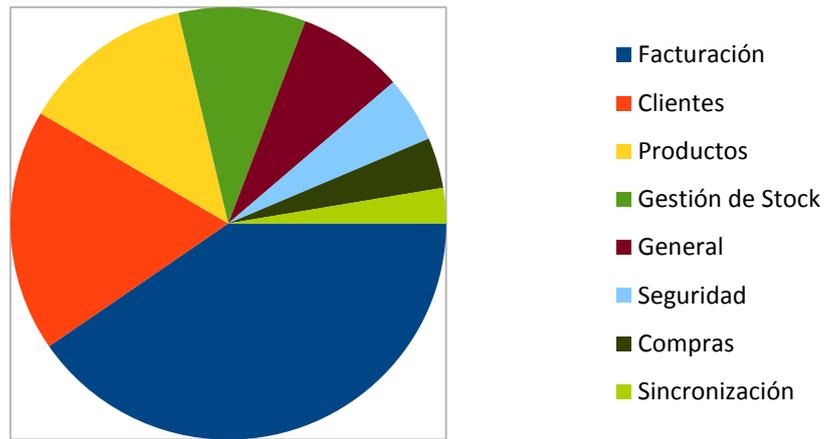


Gráfico 7:

*Distribución de las incidencias por funcionalidades*

Se observa que la gran mayoría de los reportes corresponden a las funcionalidades de facturación. Esto se encuentra alineado con lo que se había planificado, dado que era la funcionalidad más compleja y además resultaba imprescindible en el producto final.

Funcionalidad	Verificador	Reportes
Clientes	Alfonso	5
	Estera	1
	Gonzalo	20
	Juan Pablo	21
	Manuela	1
Compras	Juan Pablo	10
Facturación	Gonzalo	3
	Juan Pablo	97
	Fernanda	7
General	Gonzalo	1
	Juan Pablo	20
Stock	Gonzalo	20
	Juan Pablo	2
	Alfonso	3
Productos	Alfonso	3
	Gonzalo	8
	Juan Pablo	23
Seguridad	Gonzalo	3
	Juan Pablo	10
Sincronización	Estera	3
	Gonzalo	1

Si bien no se cuenta con información de esfuerzo por funcionalidad, sí es posible a partir del anterior conjunto de datos obtener una distribución de tareas según los reportes registrados. En la siguiente gráfica se encuentra la información estructurada de forma de observarse las distribuciones de tareas efectivas

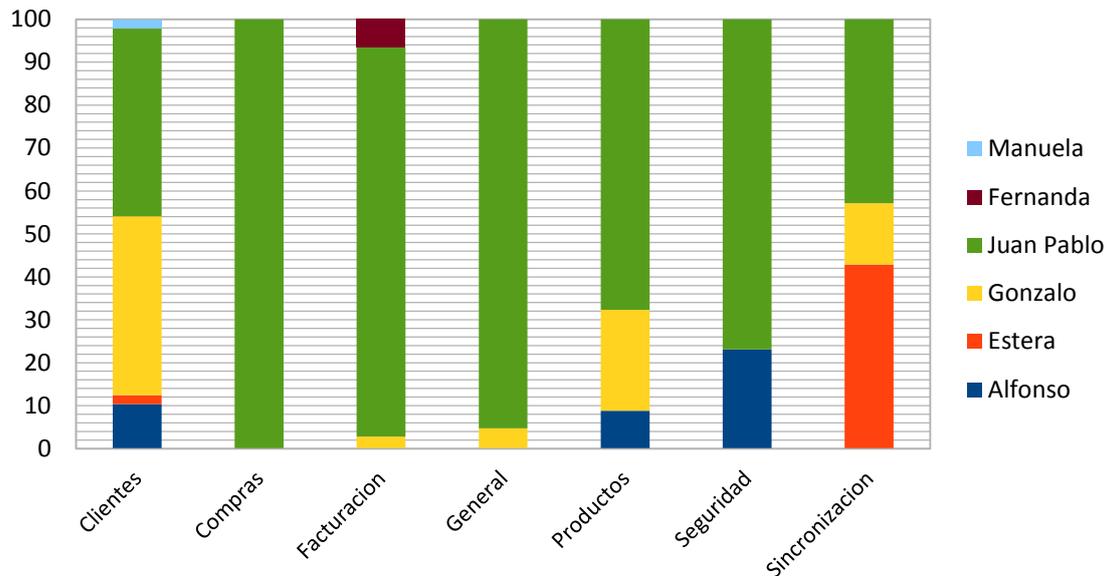


Gráfico 8: Distribución de errores por área y por verificador.

Notar que si bien se pautó la exclusividad de la herramienta de registro de errores para la notificación de los mismos, no todos los verificadores siguieron esa pauta, por lo que algún valor podría variar.

## Gravedad de los errores

Otro aspecto fundamental en el análisis de los errores es la gravedad de los mismos. Naturalmente, un proceso de verificación debería priorizar el hallazgo de los errores más críticos, mientras que además busca la reducción de las apariencias de los mismos.

Al cierre del proyecto, se cuenta con los siguientes totales.

Severidad del error	Total de hallazgos
Sugerencia	21
Trivial	8
Texto	15
Ajuste	7
Baja	133
Grave	74
Crash	2
Bloqueo	5

La primer lectura es que casi un tercio de los reportes corresponden a errores graves, bloqueantes o situaciones en las que el programa directamente presenta un error de sistema. Esto es sin duda preocupante, desde el punto de vista del producto, pero es un buen indicador en el sentido de que existían múltiples errores graves y pudieron encontrarse y solucionarse a tiempo.

Un indicador interesante es ver cómo ha variado el número de reportes graves desde el comienzo del proceso de verificación.

Semana	Número de hallazgos graves	Peso relativo (%)
7	18	37
8	1	8
9	8	31
10	9	18
11	13	30
12	22	44
13	10	27

El siguiente grafico ilustra estos valores

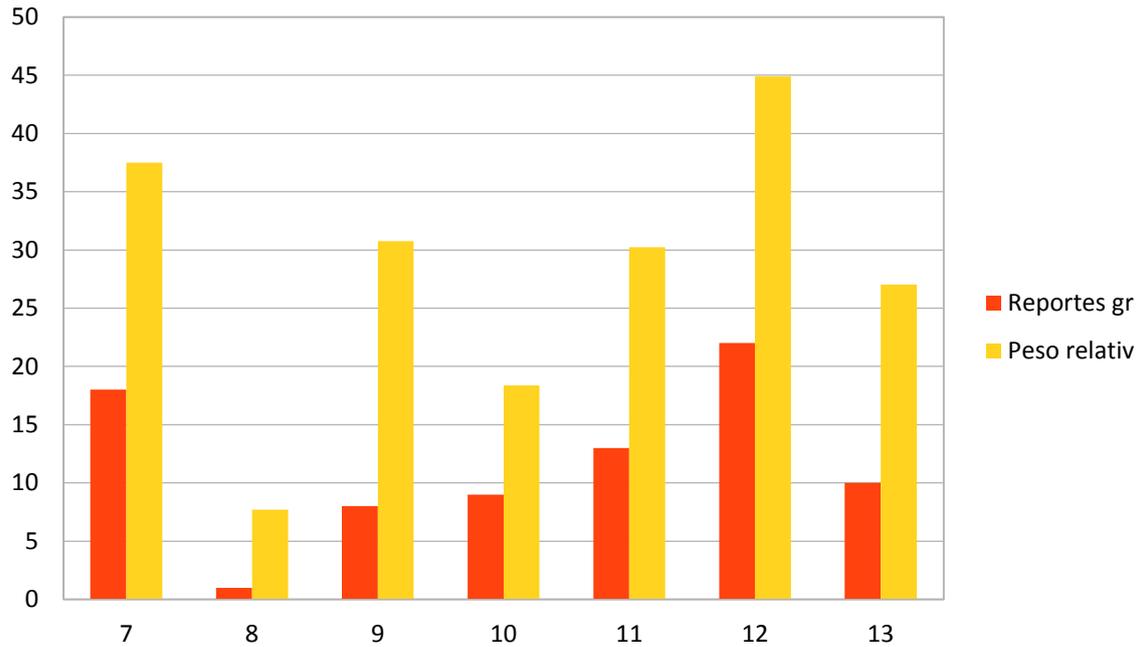


Gráfico 9: Errores graves y su peso relativo al total de errores

No se considera que pueda conjeturarse alguna tendencia respecto a la variación de los errores graves, por lo que los errores graves han acompañado al proyecto en todo momento y no se ha podido hallar algún mecanismo para evitarlo

## Cierre de la verificación

Al cierre de la verificación y de la corrección de errores, existen incidencias abiertas. Se reconocen dos tipos de errores; aquellos que eran conocidos pero su solución no pudo realizarse a tiempo para la última liberación y aquellos que surgieron a posteriori.

Se considera que de todas maneras, el número de incidencias que pudieran hallarse en el sistema es aún superior al que se menciona en este documento, por lo que quien continúe el mantenimiento de la aplicación deberá llevar a cabo procesos de verificación y corrección.

A continuación se detallan los errores conocidos que resta corregir.

## Errores detectados durante la verificación

Los errores detectados durante la verificación son aquellos que ya se habían reportado en instancias anteriores, pero debido a la falta de tiempo no pudieron solucionarse. Se trata de 4 reportes con una severidad baja y en su mayoría de funcionalidades implementadas en el último tramo del proyecto.

ID	Severidad	Resolución	Caso de uso	Descripción
232	Menor	Re-abierto	Controlar Compra Automatica.	CU Controlar Compra Automatica. Consistencia de abandono de CU
237	Menor	Abierto	Controlar Compra Automatica.	Control del campo Cant Ingresada
263	Menor	Re-abierto	Controlar Compra Automatica.	Control Popup luego de ingresar compra
241	Menor	Arreglado	Ingresar Pago Cuenta Corriente	Error de teclado luego de pop up de éxito

## Nuevos errores detectados

Desde el cierre de la verificación han existido errores que han sido detectados por los desarrolladores. Los mismos se detallan a continuación.

Severidad	Caso de uso	Descripción
Menor	Historial de operaciones	El popup de error tiene 2 colores y no uno como los demás popup de errores
Menor	Historial de operaciones	El ícono de buscar no esta alineado verticalmente con ningún campo

Menor	Historial de operaciones.	La tabla queda con un borde azul una vez terminada la búsqueda en chrome.
Menor	Historial de operaciones	Si se navega con el mouse en la flechas de paginacion, no se puede seguir navegando en tabla con las flechitas(del teclado).
Menor	Ver Información de cliente	El pager en compras y pagos cuenta corriente debería quedar siempre en la misma posicion y no moverse dependiendo de la cantidad de datos.
Menor	Ver Información de cliente	En las pestañas con datos , si no tiene datos no debería de bajar al presionar la flecha hacia abajo.
Menor	Realizar Factura	En la búsqueda de cliente, si el cliente no está registrado debería mostrar el mensaje "No se encuentran resultados para la búsqueda" de la misma manera que se muestra cuando hay resultados
Mayor	Realizar Factura	El archivo de configuración .xml para la numeración inicial de facturas no discrimina por tipo de documento
Menor	Ajustar Stock	Al buscar los productos si intento borrar una letra con backspace mientras aparece el "cargando índice", la página reacciona volviendo a la página anterior.
Menor	Información Producto	El pager de Descuentos debería quedar siempre en la misma posición y no cambiar dependiendo de los datos

## Evaluación

Como etapa de evaluación hay dos aspectos importantes a medir. Primero, la formalidad y adecuación al proceso establecido y por otra parte el éxito obtenido durante el mismo.

Primeramente, hay que considerar que muchas de las técnicas y métricas que serían esperables de un proceso como el presente no pudieron obtenerse dado que no hubo un ajuste demasiado estricto a las exigencias del modelo MUM. Tal es así que no es posible estimar la cobertura realizada de las pruebas o si restan aspectos a probar. Si bien hubieron intentos de realizarlo, no colmaban las expectativas. Mucha de la documentación no pudo generarse a tiempo o directamente desestimada, por lo que tampoco resultó ser un punto demasiado fuerte del proceso ejecutado. Tampoco se obtuvieron los resultados que se esperaban respecto de las pruebas unitarias, punto muy importantante para la calidad del producto final.

Por otra parte, se estima que los objetivos fijados inicialmente "que el producto presente altos niveles de usabilidad" y que "lo crucial funcione de la mejor manera" pudieron cumplirse a un nivel muy aceptable, punto en el cual el cliente coincide. Además, las políticas llevadas a cabo durante el proceso para alcanzar esos objetivos fueron claramente exitosas, tal como se ha evidenciado anteriormente. El producto contiene todas las funcionalidades acordadas e incluso más, y el equipo de verificación ha podido llevar a cabo las pruebas de sistema en tiempo y forma.

La interacción entre los equipos de desarrollo y verificación ha sido adecuada, siendo éste un punto que ha sido mencionado en anteriores evaluaciones como un aspecto crítico. Si bien tuvo sus complicaciones las mismas pudieron eventualmente sortearse y no significar un obstáculo para el proyecto.

Es posible que haya faltado organización de los recursos, una mayor planificación de los mismos a lo largo del proyecto y una mejor distribución de las tareas. De todas maneras, y a pesar de lo anterior, se considera que esos factores no han incidido de forma tan decisiva como para evidenciarse en el producto final.

En muchos casos la falta de disponibilidad del responsable de verificación y los problemas externos al proyecto que ocurrieron durante el mismo dificultaron el desarrollo del proceso que si bien no imposibilitó la fabricación del producto de software, no se considera el más apropiado.

Finalmente, se considera que si bien se pudo haber logrado una mayor adhesión al proceso propuesto, el mismo se ha adaptado a la realidad del proyecto en una forma muy positiva, dado que ha permitido alcanzar el fin último que es el de un producto de alta calidad.