

**Universidad de las Ciencias Informáticas**

**Facultad 7**



**Trabajo de Diploma para Optar por el Título de Ingeniero en Ciencias  
Informáticas**

**Título:** Solución para la obtención e interpretación de la Información asociada  
a las No Conformidades del software

**Autor:** Modesto de La Cruz Balbuena Zamora

**Tutor:** Ing. Yurien Ricardo Fuentes Guerra

**Co-Tutor:** Ing. Dariel Fernando Reyes Prieto

**La Habana, junio de 2011**

**“Año 53 de la Revolución”**

## DATOS DE CONTACTO

---

### Datos de Contacto

**Correo electrónico:** [mlbalbuena@estudiantes.uci.cu](mailto:mlbalbuena@estudiantes.uci.cu)

**Yurién Ricardo Fuentes:** Graduado de Ingeniería en Ciencias Informáticas en la Universidad de las Ciencias Informáticas en el año 2008. **Correo electrónico:** [yrfuentes@uci.cu](mailto:yrfuentes@uci.cu)

**Dariel Fernando Reyes Prieto:** Graduado de Ingeniería en Ciencias Informáticas en la Universidad de las Ciencias Informáticas en el año 2009. **Correo electrónico:** [dfreyes@uci.cu](mailto:dfreyes@uci.cu)

### **Dedicatoria**

*A Dios, por darme la oportunidad de formarme en esta universidad y ver hoy mi sueño hecho realidad.*

*A mami y a papi por darme la vida y ser más que mis padres, mis confidentes y amigos eternos. A mis hermanos ya que he sido el mechero de sus sueños.*

*A todos mis amigos especialmente a Reinier Ramos Oliva y a Reinier Alarcón González que me han apoyado y ofrecido una palabra de aliento cuando la he necesitado...gracias.*

*A Alexis y Alfredo por ser junto a Ramos mis padres incondicionales aquí en la universidad.*

*A todos los profes que han contribuido con mi formación.*

### **Agradecimientos**

*A las personas más importantes de mi vida:*

*Mami y Papi. Sin los cuales no estaría donde estoy ni sería hoy quien soy.*

*A Miruleidis (mi tatikota) por su confianza, apoyo y amor incondicional....gracias*

*A Ramos y Alarcón mis hermanos...gracias*

*Al tribunal de esta Tesis por señalarme con paciencia mis errores...gracias*

*A mi Co-tutor Dariel Fernández Prieto y Jenny por que sin ellos no hubiera podido terminar el desarrollo de esta investigación...gracias*

*A todas las personas que me han ayudado o han ayudado a las personas que quiero, directa o indirectamente.*

### **Resumen**

El presente trabajo de diploma define una serie de indicadores de medición para la interpretación de la información asociada a las No conformidades (NC). Así como la incorporación de nuevos campos y reportes a la herramienta Redmine, asociados a dichos indicadores. Constituyendo una solución que permite obtener e interpretar la información asociada a las NC generadas durante los procesos de pruebas, sirviendo de base para la toma de decisiones dentro de los proyectos y el avance del desarrollo de software.

Para el desarrollo de la solución fue necesario el análisis de los procesos de pruebas de liberación realizados en el Centro de Informática Médica (CESIM), así como el tratamiento de la información asociada a las NC registradas. Además se analizaron varias herramientas de seguimiento de errores para las propuestas de mejoras al sistema Redmine. Basadas también en los resultados de las entrevistas realizadas a los involucrados en los procesos de pruebas dentro de los proyectos productivos.

La utilización de la solución propuesta en el Centro permitirá facilitar el proceso de toma de decisiones en los proyectos, así como el desarrollo de software. Además servirá de base para futuras investigaciones.

**Palabras claves:** No conformidades, corrección, indicadores de medición.

**ÍNDICE**

<b>INTRODUCCIÓN</b> .....	<b>1</b>
<b>CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA</b> .....	<b>5</b>
<b>1.1 CONCEPTOS FUNDAMENTALES ASOCIADOS AL DOMINIO DEL PROBLEMA:</b> .....	<b>5</b>
<b>1.2 HERRAMIENTAS PARA EL SEGUIMIENTO DE ERRORES</b> .....	<b>9</b>
<b>1.2.1. ¿Qué es un sistema de seguimiento de errores?</b> .....	<b>9</b>
<b>CAPÍTULO 2. DIAGNÓSTICO DEL ESTADO ACTUAL</b> .....	<b>22</b>
<b>2.1 DESCRIPCIÓN DE LOS PROCESOS VINCULADOS AL CAMPO DE ACCIÓN</b> .....	<b>22</b>
<b>2.2 REALIZACIÓN DE LA ENTREVISTA</b> .....	<b>23</b>
<b>2.3 OBJETIVO DE LA ENTREVISTA</b> .....	<b>24</b>
<b>2.4 SELECCIÓN DEL PERSONAL PARA LA ENTREVISTA</b> .....	<b>24</b>
<b>2.5 DEPARTAMENTOS INVOLUCRADOS</b> .....	<b>24</b>
<b>2.6 RESULTADOS DE LA ENTREVISTA</b> .....	<b>25</b>
<b>2.7 PRINCIPALES PROBLEMAS DETECTADOS EN LOS DIFERENTES DEPARTAMENTOS DEL CESIM</b> .....	<b>28</b>
<b>CAPÍTULO 3. PROPUESTA DE SOLUCIÓN</b> .....	<b>30</b>
<b>3.1 MÉTODO UTILIZADO PARA DEFINIR LA MEDICIÓN</b> .....	<b>30</b>
<b>3.2 OBJETIVO DE INDICADORES DE MEDICIÓN</b> .....	<b>32</b>
<b>3.3 ALCANCE</b> .....	<b>32</b>
<b>3.4 OBJETIVOS DE MEDICIÓN</b> .....	<b>32</b>
<b>3.5 INDICADOR DE NO CONFORMIDADES NO VÁLIDAS</b> .....	<b>35</b>
<b>3.6 INDICADOR DE NO CONFORMIDADES DUPLICADAS.</b> .....	<b>37</b>
<b>3.7 INDICADOR DEL NIVEL DE IMPACTO DE UNA NO CONFORMIDAD</b> .....	<b>40</b>
<b>3.8 COMPORTAMIENTO DE LA NO CONFORMIDAD</b> .....	<b>42</b>
<b>3.9 MADUREZ DE UN PROBADOR EN DETECTAR UNA NO CONFORMIDAD</b> .....	<b>44</b>
<b>3.10 NIVEL DE COMPLETITUD PARA CORREGIR LA NC DETECTADA</b> .....	<b>46</b>
<b>3.11 TIEMPO DE RESPUESTA DEL DESARROLLADOR PARA UNA NO CONFORMIDAD</b> .....	<b>48</b>

3.12	PROPUESTA DE CONFIGURACIÓN DEL REDMINE .....	54
3.13	CLASIFICACIÓN DE ESTADOS DE ACUERDO AL ERROR DETECTADO EN EL SOFTWARE .....	55
3.14	ESTADOS DE UN BUG EN BUGZILLA .....	56
3.15	ESTADOS DE LA PROPUESTA PARA LAS NO CONFORMIDADES DE LIBERACIÓN.....	56
	CONCLUSIONES .....	59
	RECOMENDACIONES.....	60
	REFERENCIA BIBLIOGRÁFICA .....	61
	BIBLIOGRAFÍA.....	63
	ANEXOS .....	65

## INTRODUCCIÓN

En el desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC) a nivel mundial ha emergido un nuevo modelo económico basado en el conocimiento. El desarrollo del software es uno de los pilares fundamentales de este auge y al cual se dedican muchas horas de esfuerzos en empresas, academias y universidades. El software es el intermediario cada vez mayor entre la información y la inteligencia humana. Unido al desarrollo de las tecnologías va avanzando hacia un nivel superior en la rama de la informática, por lo que aparecen nuevas soluciones en este campo del conocimiento y un sin fin de herramientas que intentan hacer el trabajo del desarrollador un poco más fácil y mejorar el proceso de desarrollo de software, para obtener así mejores ganancias en los diferentes mercados.

Una de las instituciones cubanas protagonista del desarrollo de software es la Universidad de las Ciencias Informáticas (UCI), idea materializada por el Comandante en Jefe Fidel Castro Ruz y surgida al calor de la Batalla de Ideas. Esta casa de altos estudios fusiona los componentes de formación, investigación, producción y desarrolla un nuevo modelo con sus estudiantes -centrado en el aprendizaje- al relacionar la docencia con la producción. La UCI se organiza en facultades y centros productivos, uno de ellos es el Centro de informática Médica (CESIM), desde donde se desarrolla la presente investigación. Este centro cuenta con un grupo de calidad que vela por la optimización de los productos informáticos mediante la realización de pruebas de liberación.

Como parte de las dificultades presentadas se enmarca la **situación problemática**:

En la mayoría de los productos que se van a liberar se detectan varios errores pues no se tiene en cuenta la información asociada a las No Conformidades detectadas en revisiones anteriores. Lo que puede ocurrir por la falta de experiencia, la rapidez con que el producto llega a las pruebas de calidad o por la urgencia del cliente. Vale destacar que los diferentes departamentos que trabajan en la producción de software en el centro no tienen clientes fijos.

Actualmente el proceso de prueba de los sistemas desarrollados en el CESIM se realiza de manera separada por módulos y están estructurados por escenarios. Una vez que el probador del grupo de detecta un error en algunos de los escenarios, se registra una No Conformidad (NC) en la herramienta de gestión de procesos Redmine.

## Para registrar la NC no pueden faltar los siguientes parámetros:

- Elemento en pruebas (aplicación, documento, manual de usuario, etc.)
- Descripción de la NC
- Localización
- Clasificación
- Tipo de NC (Ortografía, funcionalidad, diseño, etc.)
- Estado

Una vez que la NC se detectada, es asignada al desarrollador que procederá a corregirla, el desarrollador tendrá acceso solo a insertar la Respuesta a la misma:

- Respuesta del Equipo de desarrollo (Resuelta, No Resuelta—con la justificación, No procede—con la justificación)

Cuando se corrige la NC y se registra en el Redmine una de las dificultades es que no se cuenta con un registro o resumen de los procesos de pruebas efectuados, específicamente con la información asociada a las NC detectadas. Los proyectos no cuentan con un registro histórico de la información aportada por las NC que pueda contribuir a la toma de decisiones durante el proceso de desarrollo del software.

A raíz de la situación actual y las deficiencias existentes, conllevan a que los esfuerzos estén encaminados al siguiente **Problema a resolver**: ¿Cómo obtener e interpretar la información asociada a las NC generadas durante los procesos de pruebas y que aporte a la toma de decisiones durante el proceso de desarrollo del software?

Donde el **objeto de estudio** es la información asociada a las NC generadas durante los procesos de pruebas y su **campo de acción** es la información asociada a las NC generadas durante los procesos de pruebas liberación que aporten a la toma de decisiones durante el desarrollo del software. Como **objetivo general** Desarrollar una **solución** que permita obtener e interpretar la información asociada a las NC generadas durante los procesos de pruebas que aporte a la toma de decisiones durante el desarrollo del software.

Para dar cumplimiento al objetivo general se proponen las siguientes **tareas de la investigación**:

- Analizar, de las herramientas existentes qué campos contemplan para las NC y qué reportes brindan que permitan la toma de decisiones para un equipo de desarrollo y su posible inserción en el Redmine.
- Analizar trabajadores que cumplen diferentes roles en los departamentos del Centro de Informática Médica (CESIM) para identificar necesidades en el proceso de pruebas con la información obtenida de las NC.
- Realizar una propuesta de mejora de la herramienta Redmine para la obtención de las no conformidades utilizando la información obtenida de ellas para orientar la toma de decisiones.
- Definir indicadores de medición para la interpretación de la información obtenida de las NC, que permita orientar su utilización para la toma de decisiones.
- Obtener de la información de las NC criterios que aporten a la toma de decisiones durante el proceso de desarrollo del software.

Los **Métodos de Investigación** utilizados son:

## 1. Métodos Teóricos

1.1. Lógico deductivo: Permitió hacer un análisis coherente que arrojó como resultado la identificación de los principales errores encontrados así como la elaboración de una propuesta de solución que resuelva los errores existentes y que como consecuencia mejore el proceso de prueba del software en el centro.

1.2. Histórico: Para el estudio de las evidencias que existen con respecto a la No Conformidad.

1.3. Modelación: Para crear una representación de la estructura de la propuesta de solución que permita facilitar su entendimiento.

## 2. Métodos Empíricos

2.1. Revisión de documentos para apoyar el estudio de las herramientas utilizadas actualmente para el tratamiento de las No Conformidades detectadas en el software tanto en el ámbito internacional como nacional.

2.2. Entrevistas para la obtención de respuestas acerca de la situación existente en el Centro de Informática Médica (CESIM) respecto al seguimiento de la información obtenida con respecto a las No Conformidades.

El presente documento está estructurado en tres capítulos como se presenta a continuación:

**Capítulo 1. Fundamentación Teórica:** En este capítulo se realiza un estudio del estado del arte a nivel nacional e internacional. Se establece un marco conceptual en correspondencia con la información obtenida de la NC. Se describen los conceptos fundamentales, asociados al dominio del problema, así como las tendencias de las tecnologías actuales a tener en cuenta.

**Capítulo 2. Diagnóstico Actual:** En este capítulo se hace un estudio del estado actual del proceso de prueba en el Centro de Informática Médica (CESIM) a través de una entrevista a los diferentes especialistas que cumplen con los roles de arquitecto, analista, administrador de la calidad y desarrolladores. Dicha entrevista arrojará por qué es necesario darle seguimiento a la información obtenida de la No Conformidad y qué aporta a la toma de decisiones.

**Capítulo 3. Propuesta de Solución:** En este capítulo se propone la propuesta de indicadores de medición que contribuirá a la interpretación de la información obtenida de la No Conformidad y apoyará la toma de decisiones.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

En el presente capítulo se analizarán los principales conceptos asociados al dominio del problema, definición de No conformidad, indicadores, acciones correctivas, además de un estudio de las herramientas en el seguimiento de errores para identificar posibles mejoras a incorporar en la herramienta Redmine.

### 1.1 Conceptos fundamentales asociados al dominio del problema:

#### ¿Qué es una no conformidad?

De acuerdo a la definición en la norma **ISO 9000: 2005**, una no conformidad es el incumplimiento de un requisito. (*Grupo de Prácticas de Auditoría ISO 9001 Guía sobre: Documentación de una no conformidad 2005*) o un error detectado en la documentación durante la revisión. (1)

Si no hay evidencia – no hay no conformidad.

Si hay evidencia – esta debe ser documentada como una no conformidad.

Si no se puede identificar el requisito, entonces no se puede levantar una no conformidad. El tratamiento que reciben las no conformidades es una parte esencial de cualquier sistema de gestión de la calidad. Si la organización desea mejorar, el primer paso es identificar lo que no se está haciendo bien, y generar un proceso de aprendizaje que permita utilizar esta información para detectar y eliminar las causas que provocan los problemas. (2)

#### Otros conceptos relacionados con las no conformidades

Otra norma de la familia 9000, la **ISO 9000:2005 - Fundamentos y vocabulario para los sistemas de Gestión de la Calidad**, establece las siguientes definiciones relacionadas a la conformidad del producto o servicio:

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

- **No conformidad:** no cumplimiento a un requisito
- **Corrección:** acción tomada para eliminar una no conformidad detectada
- **Acción correctiva:** acción tomada para eliminar la causa de una no conformidad detectada u otra situación indeseable.(3)

Se debe esperar tanto la corrección como la acción correctiva cuando se detecta una no conformidad. La **corrección** es una acción que se adopta para eliminar una no conformidad detectada, por ejemplo, el reemplazo del producto no conforme con un producto conforme o reemplazar un procedimiento obsoleto con la versión vigente. (4)

En cambio, la acción correctiva no puede ser tomada antes de hacer una determinación de la causa de la no conformidad. Para desarrollar este análisis, existen muchos métodos y herramientas disponibles, desde una simple tormenta de ideas hasta técnicas más complejas de resolución sistemática de problemas como análisis de causas raíz, diagramas Ishikawa o de espina de pescado, los cinco porqué.

La extensión y eficacia de la acción correctiva depende de la identificación de la verdadera causa raíz. En algunos casos esto ayudará a una organización a identificar y minimizar no conformidades similares en otras áreas.

Una manera de establecer un tratamiento de **No Conformidades** adecuado, consiste en verificar que el mismo contenga los siguientes elementos:

- Acciones a tomar claras y concisas
- Descripciones de acciones completas y con referencias a documentos si es apropiado
- Fecha de terminación de las acciones correctivas
- Evidencia acerca de la realización y eficacia de la acción correctiva

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

Si las no conformidades generan una respuesta adecuada por parte de la organización, conforman un proceso de aprendizaje de gran riqueza y utilidad, que puede favorecer la realización de importantes mejoras.

### **Existen diferentes definiciones de indicador entre las que se encuentran:**

Los Indicadores pueden ser medidos, números, hechos, opiniones o percepciones que señalen condiciones o situaciones específicas. (5)

También puede considerarse como magnitud utilizada para medir o comparar los resultados efectivamente obtenidos, en la ejecución de un proyecto, programa o actividad. Resultado cuantitativo de comparar dos variables.

Aunque los indicadores no son sino un instrumento de alcance limitado, su difusión se sustenta en dos de sus rasgos definitorios: su carácter sintético y su capacidad para orientar en la toma de decisiones.

La selección de un conjunto limitado pero significativo de indicadores permite hacer una idea sintética del funcionamiento de una determinada realidad.

Indicadores Cuantitativos: Son los que se refieren directamente a medidas en números o cantidades.(6)

Indicadores Cualitativos: Son los que se refieren a cualidades. Se trata de aspectos que no son cuantificados directamente. Se trata de opiniones, percepciones o juicio de parte de las personas sobre algo. (7)

La medición permite planificar con mayor certeza y confiabilidad, así como discernir con mayor precisión las oportunidades de mejora de un proceso dado. Se puede analizar y explicar cómo han sucedido los hechos.

### **¿En qué beneficia el seguimiento automatizado de los errores?**

Se pueden citar algunos beneficios según los roles más importantes relacionados con este tema:

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

**Probador:** Permite reportar cada detalle que encuentra en el software, clasificándolo para darle prioridad según su complejidad, brindando el máximo de información posible para su reproducción posterior. Permite asignar automáticamente las NC a la persona responsable de darle solución.

**Programador:** Los errores le llegarán ordenadamente, con la información necesaria para poder reproducirlo y arreglarlo, sin intercambios innecesarios con probadores o clientes. Existe un registro persistente de los errores. Permite comenzar a resolverlos de acuerdo a la criticidad de los errores, sin tener que clasificarlos personalmente. Al registrar la respuesta de la NC cualquiera de los implicados en el proceso de pruebas puede verificar la misma de forma rápida.

**Líder:** Tiene el control de todos los errores sin tener que verificarlo con cada programador o probador, permite verificar el estado de respuesta de las NC en todo momento, tomar medidas en cuanto al proceso de desarrollo del sistema y darle prioridad a los problemas a resolver,

Con la automatización del proceso de pruebas se eliminan muchos de los problemas que se presentan en el desarrollo de software. Beneficiando al software que será liberado en menor tiempo.

También facilita los procesos de medición y obtención de diversas métricas dentro del equipo, haciéndolas más fiables que al obtenerlas a partir de recogidas manuales de datos históricos.

Iniciarse en el uso de una herramienta de este tipo puede traer resistencia en los equipos acostumbrados al trabajo desordenado y no gestionado. Por lo que una vez instalada y adoptada la decisión de comenzar a utilizarla es necesario que se tomen algunas medidas iniciales, que permitan que el proceso de adaptación transcurra con mayor rapidez.

A continuación algunos consejos obtenidos de la experiencia:

- Realizar el proceso de selección de la herramienta con la participación del equipo implicado, con lo que se garantiza una selección acorde a sus necesidades.
- Presentar al equipo los beneficios de la herramienta, tanto para ellos como para el mejoramiento del producto, de manera que identifiquen la necesidad de utilización de la misma.

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

- Es necesario que el registro de las NC tengan una buena descripción, para que los desarrolladores puedan identificar y solucionar el problema detectado.
- Se debe mantener actualizado el estado de las NC para su seguimiento y control.
- Los directivos del proyecto deben mantenerse revisando constantemente la información relacionada con las NC, para poder detectar a tiempo problemas en el desarrollo, tales como atrasos, bajas productividades, entre otras deficiencias.
- El líder debe dar prioridad a los problemas, según las clasificaciones que se tengan. Nunca permitir que esté por encima de un error que bloquee un sistema una mejora o un cambio estético, aunque todos los problemas deban solucionarse.
- Debe existir un sistema de notificaciones a los implicados en el proceso de pruebas para gestionar de forma eficiente todo el proceso de detección, seguimiento y respuesta de la NC.

### 1.2 Herramientas para el seguimiento de errores

En la actualidad el uso de herramientas para el seguimiento de errores en el desarrollo de aplicaciones informáticas es una necesidad. Los beneficios que aportan al respecto, en rapidez de gestión de los errores, así como, el registro ordenado de la información relacionada con el mismo, traen numerosas ventajas, descritas en epígrafe anterior.

#### 1.2.1. *¿Qué es un sistema de seguimiento de errores?*

Un sistema de seguimiento de errores es una aplicación informática diseñada para ayudar a asegurar la calidad de software y asistir a los programadores y otras personas involucradas en el desarrollo y uso de sistemas informáticos en el seguimiento de los defectos de software. El término usado en inglés es Bug Tracking System, y frecuentemente se usa el acrónimo BTS. Puede considerarse como una especie de sistema de seguimiento de incidentes. Son usados intensivamente por cualquier empresa o institución que realice desarrollo de software. (8)

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

Muchos sistemas de seguimiento de errores de Software libre permiten que los usuarios directamente den de alta la incidencia detectada, en muchas empresas de desarrollo de software se usan de manera estrictamente interna. Muchos de los sistemas de seguimiento de errores de software se integran frecuentemente con otras herramientas, como pueden ser correo electrónico, control de versiones, y otras herramientas de gestión administrativa.

### **Code Track**

La característica principal es que no necesita ni base de datos ni servidor de correo (algo que puede ser una ventaja dependiendo de los recursos que se puedan obtener en un proyecto). La información la guarda en archivos de tipo XML<sup>10</sup> y tan sólo es necesario tener instalado Apache y PHP<sup>11</sup>. De fácil instalación, funciona en navegadores antiguos y seguro, con características que hacen que sea una aplicación sencilla y útil para crear, revisar y gestionar los informes de error y respuestas de desarrolladores para múltiples proyectos de una interfaz común. Especialmente adecuado para entornos de intranet y extranet, Code Track incluye una fuerte autenticación, y permite el control de acceso personalizado a los proyectos individuales. Los portales web por su creciente desarrollo en la red global ya cuentan con tres clasificaciones según sus diferentes usos y estrategias. (9)

### **Beneficios**

- Fuerte autenticación fuera de la caja.
- Code Track es rápida: búsqueda de miles de errores en cuestión de segundos.
- Completa la historia y la auditoría de error por defecto.
- Soporte completo para navegadores más antiguos (como Netscape 4.x).
- No hay base de datos necesarios para el backend<sup>12</sup>.
- No se necesita servidor de correo.
- Permite restringir el acceso según el usuario o papel, por sistema o por proyecto.

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

- La aplicación entera es un solo programa PHP.

### **Bugzilla**

Bugzilla utiliza un servidor HTTP (como puede ser Apache) y una base de datos (normalmente, MySQL) para llevar a cabo su trabajo. Los errores pueden ser enviados por cualquiera y pueden ser asignados a un desarrollador en particular. Cada error puede tener diferente prioridad y encontrarse en diferentes estados, así como ir acompañado de notas del usuario o ejemplos de código que ayuden a corregir el error. (10)

La noción de error en Bugzilla es muy general; por ejemplo, Mozilla.org lo utiliza también para registrar las peticiones de nuevas funcionalidades, con lo que el espectro de cuestiones sobre las que permite realizar un seguimiento se amplía.

El usuario sólo necesita un navegador web y una cuenta de correo electrónico para comenzar a usar la aplicación.

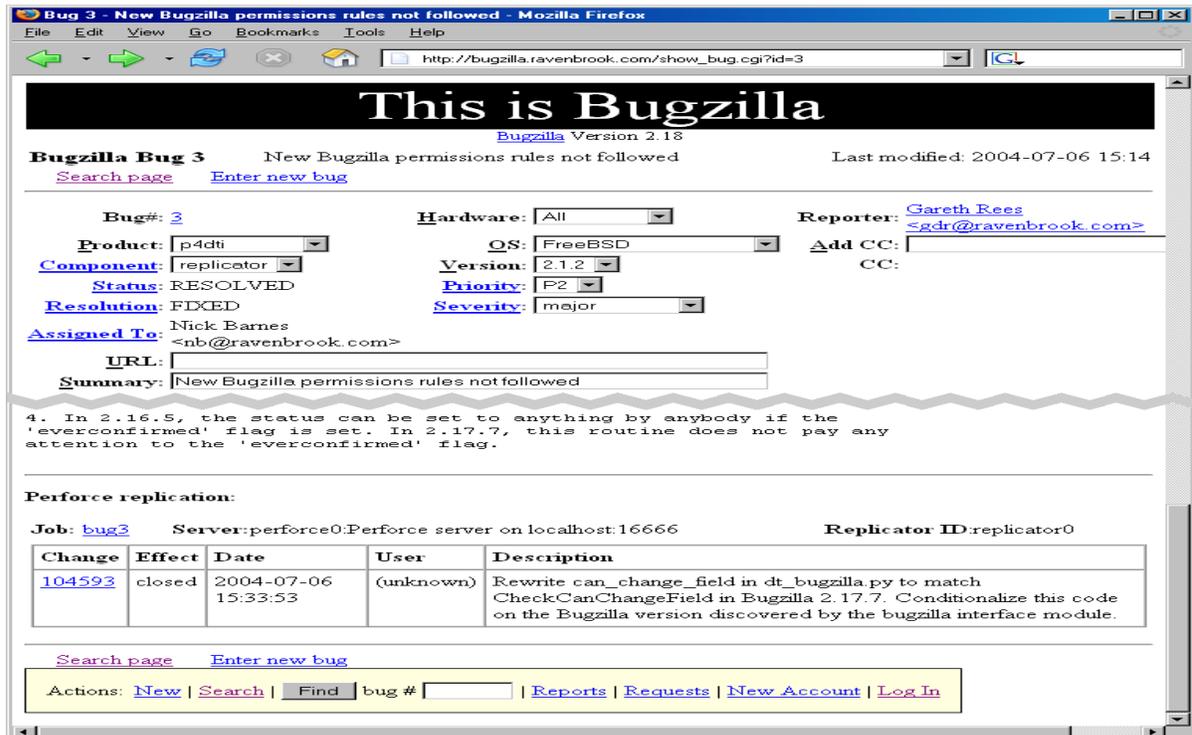
### **Características**

Bugzilla es una herramienta basada en Web de seguimiento de errores (Bug Tracking System o BTS, por sus siglas en inglés), originalmente desarrollada y usada por el proyecto Mozilla. Lanzado como software de código abierto por Netscape Communications en 1998, Bugzilla ha sido adoptado por una variedad de organizaciones para su empleo en el seguimiento de defectos (errores), tanto para software libre como para software propietario. (11)

Su licenciamiento es bajo la Licencia Pública de Mozilla.

Bugzilla permite organizar en múltiples formas los defectos de software, permitiendo el seguimiento de múltiples productos con diferentes versiones, a su vez compuestos de múltiples componentes. Permite además categorizar los defectos de software de acuerdo a su prioridad y severidad, así como asignarles versiones para su solución.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA



También permite anexar comentarios, propuestas de solución, designar a responsables a los que asignar la resolución y el tipo de solución que se aplicó al defecto, todo ello llevando un seguimiento de fechas en las cuales sucede cada evento y, si se configura adecuadamente, enviando mensajes de correo a los interesados en el error.

Durante años, Bugzilla ha sido elegida por gran cantidad de empresas y usuarios particulares, como una de las herramientas más confiables para el seguimiento de errores, y su popularidad son un claro ejemplo de su excelente funcionamiento.

Uno de los aspectos que hacen de Bugzilla una herramienta totalmente confiable, radica en que es constantemente actualizada para solucionar errores propios del programa y mejorar su funcionalidad,

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

gracias al trabajo permanente de los desarrolladores del software, que publican asiduamente actualizaciones para la aplicación.

### Estados de un Bug en Bugzilla

- Estados de un Bug en Bugzilla adaptados al proceso actual:
- NEW: El Bug ha sido registrado por parte del Equipo de Prueba (EP).
- UNCONFIRMED: El Bug ha sido registrado pero no ha sido considerado como tal.
- ASSIGNED: El Bug ha sido asignado a un miembro del Equipo de Desarrollo(ED) para su resolución.
- RESOLVED: El Bug ha sido procesado por el ED y aún no se ha verificado por el EP.
- FIXED: El Bug ha sido resuelto por el ED.
- INVALID: El ED no está de acuerdo con la validez del bug.
- WONTFIX: El ED está de acuerdo con la validez del bug pero no considera necesario resolverlo por lo menos en el presente.
- WORKSFORME: El ED no ha podido reproducir el bug.
- DUPLICATE: El EP o el ED considera que el bug es un duplicado de un bug ya registrado.
- VERIFIED: El EP ha verificado y está conforme con el estado de RESOLVED especificado por el ED.
- REOPEN: El EP ha verificado y no está conforme con el estado de RESOLVED especificado por el ED.
- CLOSED: El Bug ha sido resuelto y ya no se tiene en cuenta.

### Entre las características avanzadas que posee Bugzilla, se incluyen las siguientes:

- Esquema de seguridad integrado granular a nivel de producto.
- Dependencias entre bug y gráficos de dependencias.

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

- Capacidades avanzadas de informes.
- Repositorio estable y robusto basado en bases de datos relacionales.
- (MySQL), altamente configurable.
- Protocolo de resolución de fallos, fruto de un proceso de comprensión y pensamiento naturales.
- APIs de correo electrónico, XML, consola y HTTP.
- Integración disponible con sistemas automatizados de gestión de la configuración de software, incluyendo Perforce y CVS.

### **Ventajas:**

- Permite tener un seguimiento de errores y cambios, comunicación entre los miembros del proyecto, enviar y revisar parches y manejar la garantía de calidad. Además de llevar un control de tiempo, distintas formas de autenticación, gráficos y multilinguaje.
- Ofrece la posibilidad de organizar y gestionar defectos de software en múltiples formas, lo que hace posible el seguimiento simultáneo de diferentes productos y distintas versiones.
- Esta aplicación permite que los usuarios separen los distintos fallos detectados en diferentes categorías, según su severidad y prioridad de solución, como así también agregar propuestas y comentarios a cada error, y hasta enviar correos electrónicos con la información a los desarrolladores o usuarios del software en cuestión. (12)

### **Desventajas:**

- Se hace un poco difícil de usar.

### **Mantis Bug Tracker**

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

### **Arquitectura**

Mantis Bug Tracker es un sistema web de registro de incidencias ligero y simple. Escrito en el lenguaje PHP, requiere una base de datos (MySQL, MS SQL o PostgreSQL) y un servidor web. Mantis ha sido instalado en Windows, Mac OS, OS/2 y una variedad de sistemas operativos Unix. Cualquier navegador web debería ser capaz de funcionar como cliente. Mantis es libre de usar y modificar. (13)

Las herramientas de ejecución de mantis están basados en la suite LAMP (Linux-Apache-MySQL-Php), muy utilizada en el desarrollo Web.

Linux es el sistema operativo que albergará a mantis.

Apache es el servidor Web frontal que recibirá las peticiones http.

MySQL es una herramienta de SGBD recomendada para albergar la base de datos de mantis.

Php es la herramienta de interpretación del lenguaje PHP que permite tener páginas Web dinámicas.

# CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA



Anonymous | [Login](#) | [Signup for a new account](#)

2010-03-23 09:52 EDT

Project: All Projects

[Main](#) | [My View](#) | [View Issues](#) | [Change Log](#) | [Roadmap](#) | [Docs](#) | [Wiki](#) | [Billing](#)

Issue #

Unassigned [^] (1 - 10 / 251)	Resolved [^] (1 - 10 / 368)
<a href="#">0007970</a> test [Demo] GUI - 2010-03-23 09:30	<a href="#">0007970</a> test [Demo] GUI - 2010-03-23 09:30
<a href="#">0008052</a> Test: Summary Bug [Demo] GUI - 2010-03-22 17:02	<a href="#">0008052</a> Test: Summary Bug [Demo] GUI - 2010-03-22 17:02
<a href="#">0007968</a> test [Demo] GUI - 2010-03-08 14:48	<a href="#">0007925</a> Unicode. ???????? ??????? ??????? ??????? [Demo] GUI - 2010-03-03 05:37
<a href="#">0007964</a> test velan [Demo] GUI - 2010-03-08 08:57	<a href="#">0007916</a> Too bad mantis doesn't work with Oracle out of the box! [Demo] Other - 2010-03-02 21:58
<a href="#">0007963</a> test velan [Demo] GUI - 2010-03-08 08:57	<a href="#">0007908</a> that is a test of a report [Demo] Website - 2010-03-02 04:19
<a href="#">0007959</a> Summmmmm [Demo] Other - 2010-03-08 03:27	<a href="#">0007896</a> test [Demo] GUI - 2010-02-28 23:24
<a href="#">0007953</a> Test [Demo] Other - 2010-03-05 14:47	<a href="#">0006828</a> mantis test [Demo] Website - 2009-10-16 06:16
<a href="#">0006442</a> testicht [Demo] GUI - 2010-03-05 13:44	<a href="#">0006795</a> Password request does not work [Demo] GUI - 2009-10-16 06:16
<a href="#">0007936</a> L'écran se ferme [Demo] GUI - 2010-03-04 06:12	<a href="#">0006854</a> This bug exists! [Demo] Other - 2009-10-16 06:16
<a href="#">0007904</a> test [Demo] Other - 2010-03-01 17:04	<a href="#">0006849</a> Test Issue [Demo] GUI - 2009-10-16 06:16

## Sus principales funcionalidades son las siguientes:

- Los usuarios pueden tener diferentes niveles de acceso por proyecto.
- Creación de campos propios.
- Notificación por correo electrónico.
- Los usuarios puede monitorizar incidencias específicas.
- Distintas ubicaciones para los ficheros adjuntos (servidor web, base de datos o cuenta FTP).
- Flujos de trabajo configurables.

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

- Capacidad de exportar a CSV, Excel y Word.
- Informe de incidencias mediante correo electrónico o web.
- Control de integración de fuentes (SVN y CVS).
- Integración con Wiki.
- Seguimiento de tiempos.
- Niveles de acceso público/privado a proyectos, notas e incidencias.
- Capacidad para relacionar incidencias.
- News publicas/privadas asociables a proyectos particulares o a todos los proyectos.
- Acciones agrupadas: las acciones se puede aplicar a múltiples incidencias.
- Autenticación: propia de Mantis, integración con LDAP, HTTP Basic.
- Integración con Chat.
- Interface SOAP.
- Soporte para dispositivos móviles. (14)

### **Características**

Mantis es un software Open Source que trae un entorno totalmente ágil para poder registrar incidencias, asignar una incidencia a uno o muchos usuarios, crear roles para los usuarios (informador, desarrollador, etc), agregar documentos por incidencia, entre otras cosas.

Asimismo se puede integrar con muchas herramientas como un cliente de correo (si no se tiene un dominio propio se puede usar un correo) para que envíe un mail al momento de que Calidad reporte una nueva incidencia o cuando la incidencia cambie de estado. Se puede integrar con eclipse, CVS o SVN para que las incidencias aparezcan como tareas en el eclipse.

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

Los clientes (usuarios) del mantis, pueden acceder a sus tareas mediante cualquier browser web. Y pueden, entre otras cosas, añadir ficheros, relacionar tareas, agrupar tareas por categorías, búsqueda de tareas con un conjunto potentísimo de filtros.

### **Ventajas**

Posee una interfaz gráfica sencilla.

Anuncios globales, para anunciar una nueva versión o una noticia importante.

El flujo de trabajo también puede ser configurado desde la propia herramienta. La transición de estados también puede ser configurada sin mucho esfuerzo.

Es una herramienta gratuita de fácil configuración y mantenimiento, es ideal para el 90% de las empresas que necesitan establecer un flujo de trabajo formal.

En definitiva, una herramienta aconsejable para todo tipo de proyectos, que permite organizar mejor la resolución de incidencias sobre los mismos.

### **Redmine**

#### **Gestión de múltiples proyectos:**

Redmine permite gestionar múltiples proyectos desde una sola interfaz con una ventana de navegador. La navegación es muy sencilla y se puede saltar y cambiar de proyecto en cualquier momento. Además cada proyecto puede tener una configuración totalmente diferente y el usuario tener un rol distinto en cada uno. Los proyectos pueden definirse como privados, en los que el administrador debe dar acceso a cada miembro, o públicos, visibles para todo el mundo. También dentro de cada proyecto pueden definirse varios sub-proyectos.

#### **Personalización de proyectos:**

En Redmine cada proyecto es totalmente personalizable, pudiendo encontrar proyectos muy distintos entre sí según sus objetivos. Lo más importante son los módulos que se pueden desactivar o activar para

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

cada proyecto: wiki, foro, noticias, peticiones, control del tiempo, documentos, ficheros o repositorio, aunque hay módulos comunes a todos los proyectos como el de actividad y vistazo. Si un proyecto está enfocado a notificar incidencias, se puede configurar para incluir solo peticiones, si se busca un proyecto más colaborativo.

### **Sistema flexible de seguimiento de tareas:**

Una de las mecánicas más útiles para el desarrollo de un proyecto en Redmine son las peticiones y su visualización. Estas peticiones se dividen en 3 tipos iniciales (errores, tareas y soporte) y pueden asignarse a un miembro del proyecto. Se puede indicar una fecha de inicio y fin para esa petición, e incluso llevar un control del tiempo y porcentaje realizado. También se le puede asignar una prioridad, enlazar con la subida de un fichero, y encajar en una categoría (que se pueden definir tantas como se quieran). Con todos estos datos, pueden visualizarse las peticiones de manera personalizada estableciendo filtros, y servir así de informes de tareas o incidencias. Además dentro de un proyecto pueden establecerse versiones y asignar tareas a determinadas versiones, así conforme se marquen tareas completadas, las versiones irán completando su porcentaje automáticamente.

### **Integración en repositorios de código:**

Redmine puede integrarse con un repositorio de código (Subversión, Git, CVS, entre otros) que esté montado en la misma máquina, tan solo hay que indicarle el directorio local. La aplicación sirve así de interfaz web para el seguimiento del desarrollo de un proyecto. Pueden descargarse los ficheros, ver el historial, los cambios, e incluso descargar un archivo a modo de parche para aplicar a código desactualizado. Es un sistema de seguimiento de versiones, aunque no pueden actualizarse los ficheros directamente.

### **Uso de calendario y diagrama de Gantt:**

Redmine incluye un calendario para visualizar todas las peticiones a lo largo de un mes elegido, marcando claramente el día de inicio y de fin de cada petición. Igualmente ocurre con la vista en diagrama de Gantt,

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

que va marcando el porcentaje completado conforme avanzan los días. Las peticiones que se visualizan en ambos casos están sujetas a los filtros definidos por el usuario.

### **Notificaciones:**

Configurando previamente el servidor de correo SMTP, Redmine permite enviar notificaciones por correo electrónico en todos los proyectos, definiendo antes los eventos que activan estos avisos. Además cada usuario en su configuración puede elegir recibir notificaciones de cualquier evento, o solo las relacionadas con él (por ejemplo uno de los campos de las peticiones son las personas en seguimiento). Puede configurarse además el servidor de correo entrante, permitiendo así actualizar peticiones simplemente por email e incluso crear nuevas peticiones.

### **Exportación a distintos formatos:**

Los informes de peticiones que pueden generarse añadiendo filtros, y que permiten visualizar las diferentes tareas de un proyecto, pueden exportarse en PDF o formato CSV, pudiendo así imprimirlos posteriormente en un formato organizado.

### **Otras características:**

Redmine es una herramienta llena de funcionalidad y donde casi cualquier cosa puede configurarse o contiene opciones. Algunas otras funcionalidades que habría que destacar son la página personal de cada usuario, que ofrece una vista personalizable con información de todos los proyectos donde está participando, como un calendario global, o peticiones asignadas. También se pueden subir ficheros, definir campos personalizados para cada módulo, usar la barra de búsqueda global, y ampliar la funcionalidad con decenas de extensiones. Además cabe mencionar que admite como bases de datos MySQL, SQLite y PostgreSQL.

### **Plataformas disponibles:**

## CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

---

Redmine es una aplicación servidor multiplataforma. Los únicos requisitos para instalar Redmine en una máquina son: una base de datos (que puede ser MySQL, PostgreSQL o SQLite). Si una máquina sostiene esto, puede instalarse Redmine independientemente de la plataforma.

En este capítulo mediante el estudio teórico realizado se analizaron los principales conceptos relacionados al dominio del problema. Se describen herramientas para el seguimiento de las No Conformidades detectadas en el proceso de prueba, para determinar las posibles mejoras a incorporar a la herramienta Redmine, utilizada actualmente por el Centro de Informática Médica (CESIM) y por Calisoft que es el organismo rector de la calidad de los productos en la universidad.

## CAPÍTULO 2. DIAGNÓSTICO DEL ESTADO ACTUAL

---

### CAPÍTULO 2. DIAGNÓSTICO DEL ESTADO ACTUAL

En este capítulo se abordarán las características de la solución a desarrollar, describiendo la situación actual y el problema existente.

#### 2.1 Descripción de los procesos vinculados al campo de acción

En la actualidad en el Centro de Informática Médica (CESIM) para los procesos de pruebas de liberación se incluyen una serie de actividades encaminadas fundamentalmente al seguimiento y control de las No Conformidades (NC). Dentro de los cuales se encuentran la planificación de las pruebas, el registro de las NC, el seguimiento de las NC, las respuestas a las NC y el control de la Información asociada una vez terminado el proceso de pruebas. En estos procesos se ven involucrados diferentes personas que desempeñan roles dentro del proceso de desarrollo del software.

#### Flujo actual del proceso

Actualmente el proceso de prueba de los sistemas desarrollados en el CESIM se realiza a partir la solicitud de liberación realizada a Calisoft o al grupo de Calidad del mismo.

Aceptada la solicitud se procede a la planificación de las pruebas donde se definen: cronogramas, personal involucrado, y condiciones para las mismas.

Para la realización de las pruebas se le asigna la probador por parte del Jefe de pruebas o especialista del grupo de calidad una tarea en la herramienta Redmine donde se le especifica: Iteración, tiempo de realización, elemento o escenario a probar y la documentación correspondiente.

Una vez que el probador tiene asignada la tarea procede a realizar la misma, registrando la No Conformidad (NC) detecta en la herramienta Redmine en la sección designada para las pruebas al sistema en cuestión. Registradas las NC deben ser verificadas por el jefe de pruebas para identificar NC duplicadas o con problemas en los datos registrados.

## CAPÍTULO 2. DIAGNÓSTICO DEL ESTADO ACTUAL

---

Terminada la iteración de pruebas se procede a la respuesta por parte del equipo de desarrollo. Quienes cambian el estado de la NC en dependencia de la solución o no de la misma. Danto paso a una nueva fase por parte del equipo de pruebas que verifica las respuestas dadas, si todas fueron respondidas se pasa a una nueva iteración donde se vuelve a repetir todo el proceso antes descrito.

Concluidas las Iteraciones previstas para el sistema, si cumple con los requerimientos de calidad propuestos se emite una acta de liberación por parte equipo de pruebas. Toda la información asociada la proceso de pruebas queda registrado en la herramienta Redmine, utilizada por parte de los proyectos involucrados y los grupos de calidad como constancia de las pruebas realizas.

A esta información relacionada con las NC registradas no se les realiza un buen seguimiento. Los proyectos no cuentan con un registro histórico o resumen de la información aportada por las NC que pueda contribuir a la toma de decisiones durante el proceso de desarrollo del software.

Para la identificación de una mayor cantidad de datos relacionados con el proceso antes descrito, así como elementos que aporten a la interpretación de la información asociada a las NC, se decide realizar una entrevista a las personas que desempeñan los roles relacionados con este proceso de pruebas de liberación.

### **2.2 Realización de la entrevista**

Una de las técnicas de recopilación de información más utilizadas es la entrevista, mediante ella, se obtienen datos específicos y concretos, por este motivo se decidió utilizarla como sustento de la investigación.

La entrevista aplicada tuvo un carácter personal e individual, además de ser estructurada. Las preguntas realizadas fueron confeccionadas con el objetivo de obtener información detallada sobre el seguimiento que se le da actualmente a las informaciones obtenidas de las No Conformidades detectadas en las pruebas de liberación. **(Ver Anexo 1)**

## CAPÍTULO 2. DIAGNÓSTICO DEL ESTADO ACTUAL

---

### **2.3 Objetivo de la entrevista**

Evaluar y buscar posibles soluciones a los problemas existentes relacionados con las pruebas de liberación, e identificar otros posibles problemas.

### **2.4 Selección del personal para la entrevista**

Las entrevistas fueron realizadas principalmente a los directivos de los distintos departamentos y proyectos productivos, dentro de ellos a miembros que desempeñan roles principales como analista, arquitecto, desarrolladores, administradores de la calidad y al grupo de calidad interno. Ya que no todos los estudiantes y profesores vinculados a la producción tienen conocimientos sobre el tema en cuestión y esto puede provocar confusiones y falsas estadísticas.

### **2.5 Departamentos Involucrados**

- Sistema de Atención a la Salud (SAS)
- Atención Primaria a la Salud (APS)
- Tecnologías, Integración y Estándares (TIE)
- Software Médico Imagenológico (SWMI)
- Software de Gestión Hospitalaria (SGH)
- Grupo de trabajo de la calidad.

## CAPÍTULO 2. DIAGNÓSTICO DEL ESTADO ACTUAL

**Tabla 2.1:** Muestra la cantidad de especialistas por roles en las diferentes Áreas Temáticas.

Departamento	Administrador de la Calidad	Arquitecto	Desarrollador	Analista	TOTAL
SWMI	1	2	2	1	6
SAS	1	2	1	2	6
SGH	1	1	3	1	6
TIE	1	1	2	2	6
APS	1	3	2	1	6
TOTAL	5	7	10	8	30

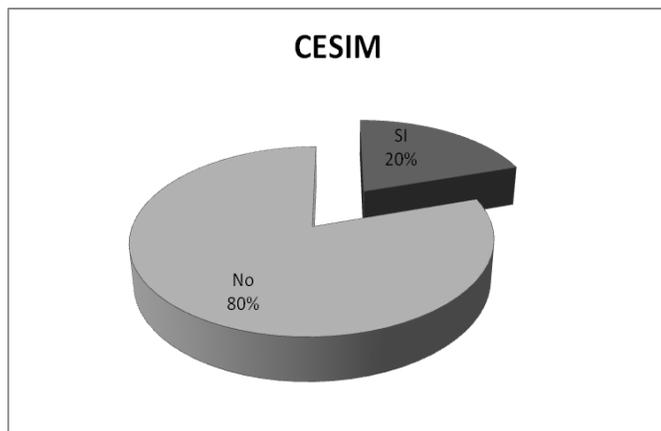
En las gráficas 2.1, 2.2, 2.3, 2.4 y 2.5 se muestran las gráficas del personal entrevistado (arquitectos, analistas, desarrolladores y administradores de la calidad) por departamentos productivos. **(Ver Anexos 2, 3, 4, 5, 6)**

### 2.6 Resultados de la Entrevista

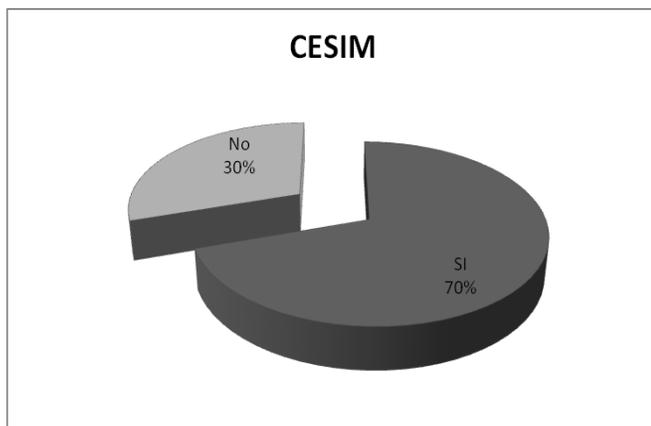
Resultados de la entrevista al personal capacitado y especializado que cumplen con los diferentes roles de arquitectos, analistas, desarrolladores y administradores de la calidad en los diferentes departamentos del CESIM lo que permitió evaluar el estado actual del Proceso de prueba en el seguimiento de la información obtenida de una No Conformidad detectada en las Pruebas de liberación.

## CAPÍTULO 2. DIAGNÓSTICO DEL ESTADO ACTUAL

**Figura 2.6:** Por ciento de departamentos en los cuales los proyectos dan seguimiento a las NC.

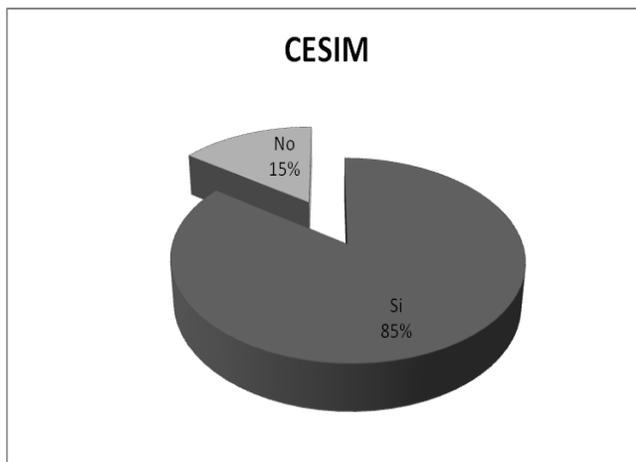


**Figura 2.7:** Por ciento de proyectos que clasifican la NC según su nivel de importancia con el fin de priorizar las acciones correctivas encaminadas a su solución.

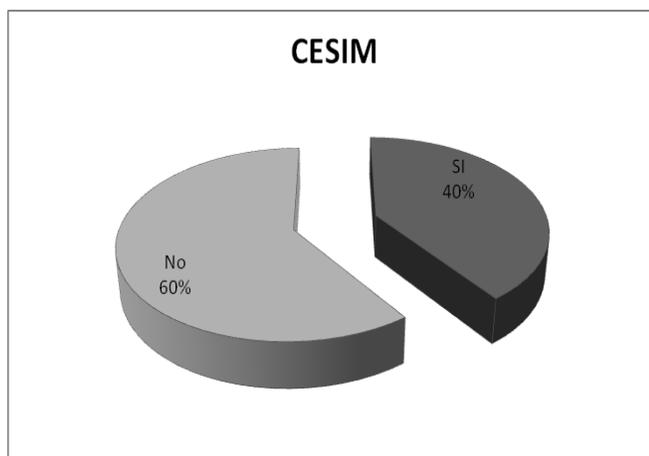


## CAPÍTULO 2. DIAGNÓSTICO DEL ESTADO ACTUAL

**Figura 2.8:** Por ciento de departamentos en los cuales los proyectos no cuentan con una guía de errores frecuentes en la programación con el fin de evitar errores ya encontrados en pruebas de liberación realizadas a diferentes módulos.



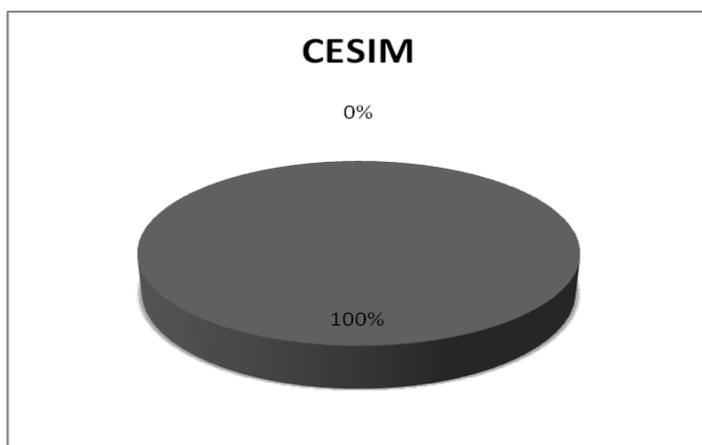
**Figura 2.9:** Por ciento de departamentos que no registran en la herramienta de gestión de proyecto Redmine las NC detectadas en las revisiones internas.



## CAPÍTULO 2. DIAGNÓSTICO DEL ESTADO ACTUAL

---

**Figura 2.10:** Por ciento de departamentos que no utilizan indicadores de medición de la calidad para darle solución y orientar la toma de decisiones a la información obtenida de las No Conformidades detectadas en las pruebas de liberación.



### 2.7 Principales problemas detectados en los diferentes departamentos del CESIM

A partir de los datos obtenidos se hizo una valoración general donde:

1. Los departamentos productivos no tienen definida una guía con el objetivo de brindar a los desarrolladores errores comunes o errores detectados en revisiones anteriores.
2. Los grupos internos de calidad se enfocan solamente a la realización de algunas pruebas a los módulos y revisiones de la documentación.
3. Solo se tiene en cuenta la calidad del producto final y no se realizan actividades para asegurar la calidad del proceso.

## CAPÍTULO 2. DIAGNÓSTICO DEL ESTADO ACTUAL

---

4. El funcionamiento de los grupos internos de calidad no se rige por indicadores de medición destinada al proceso de prueba del software.
5. Los entrevistados opinaron que no se utilizan indicadores de medición de la calidad para orientar la toma de decisiones y en algunos casos demostraron el desconocimiento de los mismos.
6. No se le da seguimiento a la información que brindan las No Conformidades detectadas en las pruebas de liberación.

En este capítulo se procesaron los datos obtenidos en la entrevista aplicada. Lo que permitió precisar el estado actual del proceso de prueba con respecto a las actividades para asegurar la calidad del mismo y de los productos en fase de liberación. Por medio de la recopilación de los datos, se definieron los problemas que más afectan la calidad del software en todos los departamentos productivos y que necesitan una solución más inmediata.

Todo lo cual evidenció la necesidad de una vía para darle solución a estas dificultades, de forma que todos los departamentos del CESIM trabajen uniformemente en el Proceso de prueba para darle un mejor tratamiento a la información obtenida de la No Conformidad, en todo su ciclo de vida.

### CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

Mediante la realización de diferentes entrevistas se logró conocer la situación actual de los departamentos y obtener información asociada a las No Conformidades, se pudo comprobar que no es satisfactorio el seguimiento a la información asociada a la NC en el Centro de Informática Médica (CESIM) para orientar la posible toma de decisiones, para lograr un trabajo productivo en el proceso de software, que permita alcanzar mayores niveles certeza, rapidez y validez.

Se presenta en el desarrollado de este capítulo la solución para la obtención e interpretación de la información asociada a las No Conformidades (NC) en el proceso del software, que tiene sus bases en indicadores para medir y evaluar la información asociada a las NC en el Centro de Informática Médica (CESIM) y la inserción de campos o parámetros a la herramienta de gestión de procesos Redmine para la perfeccionamiento continuo del ciclo de vida de una NC.

#### 3.1 Método utilizado para definir la medición

Para desarrollar las actividades de definiciones de los Objetivos de Medición, Preguntas, Indicadores y Medidas a utilizar durante la definición de la solución propuesta, se usará el Método GQ (I)M (Goal Question (Indicator) Measure).

Este Método se divide en 10 pasos, y deben ser ejecutados indistintamente por los roles involucrados y según las indicaciones de las tareas descritas en IPP-3550: 2009 Libro de Proceso para Medición y Análisis.

Paso 1. Identificar los Objetivos de Negocio.

Paso 2. Identificar que quiero saber o aprender.

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

---

Paso 3. Identificar los Sub-Objetivos.

Paso 4. Identificar Entidades y Atributos.

**Paso 5. Formalizar los Objetivos de Medición.**

**Paso 6. Identificar las Preguntas cuantificables e Indicadores.**

**Paso 7. Identificar los Elementos de Datos.**

**Paso 8. Definir las Medidas.**

Paso 9. Identificar las acciones

Paso 10. Prepare un Plan.

De los pasos propuestos por el método se utilizarán solamente los pasos 5, 6, 7 y 8, ya que son los que responden a las necesidades del problema existente, aunque también se tomaron algunos elementos de los otros pasos propuestos.

Los Indicadores pueden ser útiles en muchas formas, no sólo gráficos, sino también tablas. Las Preguntas cuantificables deben poder ser respondidas por estos Indicadores y se definen sobre las bases establecidas previamente. Es importante notar los casos en que con un Indicador se responda a más de una Pregunta.

Para llevar a cabo este paso:

1. Seleccionar uno de los Objetivos de Medición.

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

---

2. Identificar las Preguntas cuantificables relacionadas con este Objetivo que se desean responder (Objetivos, Preguntas e Indicadores).
3. Preparar los bocetos de Indicadores que ayudarán a dirigir las preguntas y comunicar los resultados del análisis a los demás, para ser evaluados por la audiencia de los mismos (lectores) y validar la efectividad de los mismos (Objetivos, Preguntas e Indicadores).
4. Repetir las 3 actividades anteriores para todos los Objetivos de Medición.

### 3.2 Objetivo de indicadores de medición

Interpretación de los indicadores medición definidos para orientar la posible toma de decisiones en futuras revisiones a los diferentes productos del CESIM.

### 3.3 Alcance

Obtener e interpretar la información asociada a la No Conformidad.

Analizar actividades de medición y análisis para la toma de decisiones.

Proveer sus reportes asociados y sus resultados de medición.

### 3.4 Objetivos de Medición

**No. del Objetivo de Medición:** Analizar las No Conformidades detectadas en el Proceso de prueba a partir de su clasificación y cantidad con el fin de caracterizar el Proceso de prueba y el Proceso de desarrollo.

<b>Objeto de Interés</b>	Evaluar el Proceso de prueba en el Centro de Informática Médica (CESIM) para darle seguimiento a la información obtenida de las No Conformidades en las pruebas de liberación.
<b>Propósito</b>	Analizar las No Conformidades detectadas en el Proceso de Prueba con

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

	el fin de mejorar la calidad del producto y el proceso de desarrollo del software.
<b>Perspectiva</b>	Examinar la Clasificación y Cantidad de las No Conformidades detectadas en el proceso de prueba para las No conformidades de liberación desde el punto de vista del Equipo de prueba y el Equipo de desarrollo.
<b>Ambiente</b>	<p>Universidad Productiva: Cumpliendo con los principios de formación, investigación, producción.</p> <p>Trabajando por alcanzar el Nivel 2 CMMI.</p> <p>Equipos de Desarrollo y Equipo de Prueba compuestos por estudiantes, profesores y especialistas.</p> <p>Herramienta: Redmine Gestor de múltiples proyectos.</p> <p>Alcance: Analizar los departamentos del Centro de Informática Médica y proponer requerimientos de calidad para orientar la toma de decisiones.</p>

Indicador	Enfocado a	Responsables			Reporte en el que se utiliza
		Obtener	Interpretar	Analizar	
Cantidad NC No válidas	Probador	Equipo de Prueba	Equipo de Prueba	Jefe de Equipo de Prueba	<i>Cantidad de NC detectadas No válidas por cada probador</i>

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

Cantidad duplicadas	NC	Probador	Equipo Prueba	de	Equipo Prueba	de	Jefe de Equipo Prueba	del	Cantidad de NC detectadas duplicadas
Nivel de impacto de una NC		Desarrollador	Equipo Desarrollo	de	Equipo Desarrollo	de	Jefe de Equipo Desarrollo	del	Nivel de impacto de la NC
Nivel de completitud de la tarea		Probador	Equipo Prueba	de	Equipo Prueba	de	Jefe de Equipo Prueba	del	Éxito de un probador al completar una tarea
Tiempo de respuesta del desarrollador para una NC		Desarrollador	Equipo Desarrollo	de	Equipo Desarrollo	de	Jefe de Equipo Desarrollo	del	Rapidez del desarrollador en eliminar una NC
Madurez del probador detectar una NC		Probador	Equipo Prueba	de	Equipo Prueba	de	Jefe de Equipo Prueba	del	Rapidez del probador en detectar una NC
Comportamiento de		Desarrollador	Equipo	de	Equipo	de	Jefe de Equipo	del	Comportamiento de

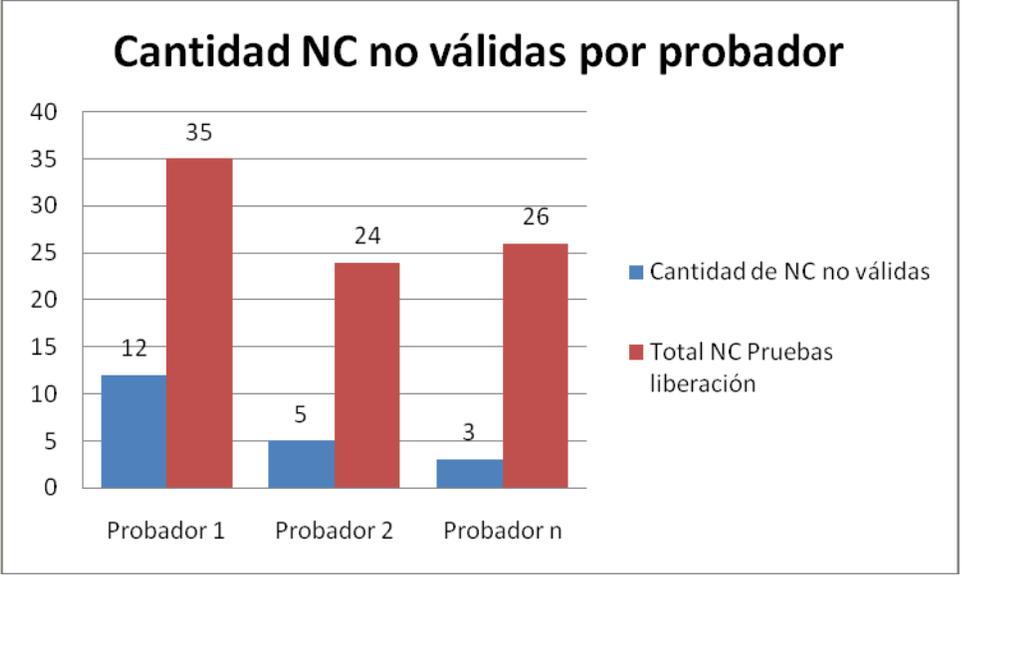
## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

la NC		Desarrollo	Desarrollo	Desarrollo	de la NC
-------	--	------------	------------	------------	----------

### 3.5 Indicador de No Conformidades No válidas

<b>#1 del Objetivo de Medición</b>	Analizar probadores que se le detectan mayor número de No Conformidades No válidas en las pruebas de liberación.				
<b>Preguntas</b>	¿Promedio de equivocación del equipo de prueba? ¿Qué probadores inciden negativamente en el Proceso de prueba?				
<b>Indicadores</b>					
<b>Indicador:</b> Cantidad NC No válidas					<b>Fecha:</b>
<b>Enfocado a:</b> Probador			<b>Frecuencia:</b>	Por cada	<b>15/01/2011</b>
			iteración		

\* La Prioridad de los Indicadores responde a la utilidad apreciable de estos en el momento de responder las preguntas planteadas. No construir Indicadores innecesarios, implica (probablemente) la necesidad de recopilar menos datos posteriormente. El rango previsto se interpreta de la siguiente manera: 1: Muy útil. 2: Útil. 3: Poco o Nada útil.

<p><b>Gráfico</b></p>	 <table border="1"> <caption>Cantidad NC no válidas por probador</caption> <thead> <tr> <th>Probador</th> <th>Cantidad de NC no válidas</th> <th>Total NC Pruebas liberación</th> </tr> </thead> <tbody> <tr> <td>Probador 1</td> <td>12</td> <td>35</td> </tr> <tr> <td>Probador 2</td> <td>5</td> <td>24</td> </tr> <tr> <td>Probador n</td> <td>3</td> <td>26</td> </tr> </tbody> </table>	Probador	Cantidad de NC no válidas	Total NC Pruebas liberación	Probador 1	12	35	Probador 2	5	24	Probador n	3	26
Probador	Cantidad de NC no válidas	Total NC Pruebas liberación											
Probador 1	12	35											
Probador 2	5	24											
Probador n	3	26											
<p><b>Medidas Base y/o Derivadas implicadas</b></p>	<p><b>Medida Base:</b> Cantidad de No Conformidades No válidas</p> <p><b>Medida Derivada:</b>                  Cantidad de No Conformidades No válidas/ Total de NC detectadas en las Pruebas de liberación por cada probador.</p> <p>Media no válidas = <math>\sum</math>Promedio No válidas por probador / Cantidad Probadores.</p>												
<p><b>Algoritmo de obtención</b></p>	<p>Datos y Ejes representados:</p> <p>Eje Horizontal: Se listan los probadores que detectan las No Conformidades No válidas.</p> <p>Eje Vertical: Se listan los valores entre los cuales oscilan las medias de las</p>												

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

	cantidades de No Conformidades detectadas No válidas.
<b>Guías para la interpretación</b>	<p>Cuando la línea de puntos tiene una pendiente negativa, se infiere que en el transcurso del tiempo se le ha dado un seguimiento satisfactorio a las no conformidades detectadas No válidas.</p> <p>Cuando la línea de puntos tiene pendiente positiva, se infiere que en el transcurso del tiempo no se le ha dado un seguimiento satisfactorio a las no conformidades detectadas No válidas.</p> <p>Caso ideal: Cuando la línea de puntos coincide con el eje cero horizontal, se infiere que en la iteración en cuestión no se encontraron no conformidades No válidas. En este caso la organización sigue los procesos de desarrollo y esto incide positivamente en los productos que elaboran.</p>

### 3.6 Indicador de No conformidades duplicadas.

<b>#2 del Objetivo de Medición</b>	Analizar módulos que se le detecten con mayor número de No Conformidades duplicadas en las pruebas de liberación.	
<b>Preguntas</b>	<p>¿Cantidad de No Conformidades detectadas duplicadas que tiene cada probador?</p> <p>¿Promedio de duplicadas en el Proceso de prueba?</p> <p>¿Qué probadores inciden negativamente en el Proceso de prueba?</p>	
<b>Indicadores</b>		
<b>Indicador:</b>	Cantidad de No Conformidades duplicadas	<b>Fecha:</b>

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

<b>Enfocado a:</b> Jefe de prueba		<b>Frecuencia:</b> Por cada 28/01/2011												
		Iteración												
<b>Gráfico</b>	<p style="text-align: center;"><b>Cantidad NC duplicadas por probador</b></p> <table border="1"> <caption>Data for 'Cantidad NC duplicadas por probador'</caption> <thead> <tr> <th>Probador</th> <th>Cantidad de NC duplicadas</th> <th>Total NC Pruebas liberación</th> </tr> </thead> <tbody> <tr> <td>Probador 1</td> <td>9</td> <td>20</td> </tr> <tr> <td>Probador 2</td> <td>5</td> <td>15</td> </tr> <tr> <td>Probador n</td> <td>2</td> <td>8</td> </tr> </tbody> </table>		Probador	Cantidad de NC duplicadas	Total NC Pruebas liberación	Probador 1	9	20	Probador 2	5	15	Probador n	2	8
Probador	Cantidad de NC duplicadas	Total NC Pruebas liberación												
Probador 1	9	20												
Probador 2	5	15												
Probador n	2	8												
<b>Medidas Base y/o Derivadas implicadas</b>	<p><b>Medida Base:</b> Cantidad de No Conformidades duplicadas.</p> <p><b>Medida Derivada:</b></p> <p>Cantidad de No Conformidades duplicadas/Total de No Conformidades detectadas en las Pruebas de liberación.</p> <p>Media duplicadas = <math>\frac{\sum \text{Promedio duplicadas por probador}}{\text{Cantidad Probadores}}</math>.</p>													
<b>Algoritmo de</b>	Datos y Ejes representados:													

\* La Prioridad de los Indicadores responde a la utilidad apreciable de estos en el momento de responder las preguntas planteadas. No construir Indicadores innecesarios, implica (probablemente) la necesidad de recopilar menos datos posteriormente. El rango previsto se interpreta de la siguiente manera: 1: Muy útil. 2: Útil. 3: Poco o Nada útil.

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

<b>obtención</b>	<p>Eje Horizontal: Se listan los probadores que detectan las No Conformidades duplicadas.</p> <p>Eje Vertical: Se listan los valores entre los cuales oscilan las medias de las cantidades de no conformidades detectadas duplicadas.</p>
<b>Guías para la interpretación</b>	<p>Cuando la línea de puntos tiene una pendiente negativa, se infiere que en el transcurso del tiempo se le ha dado un seguimiento satisfactorio a las no conformidades duplicadas encontradas.</p> <p>Cuando la línea de puntos tiene pendiente positiva, se infiere que en el transcurso del tiempo no se le ha dado un seguimiento satisfactorio a las no conformidades duplicadas encontradas.</p> <p>Caso ideal: Cuando la línea de puntos coincide con el eje cero horizontal, se infiere que en la iteración en cuestión no se encontraron no conformidades duplicadas. En este caso la organización sigue los procesos de desarrollo y esto incide positivamente en los productos que elaboran.</p>

**Tabla 3.5: Propuesta dado el nivel de impacto y prioridad de una No Conformidad.**

<b>Alto</b>	<b>Medio</b>	<b>Bajo</b>
Funcionalidad(Aplicación)	Error técnico	Ortografía(documentación)
Excepciones(Aplicación)	Formato(Documentación)	Redacción(documentación)
Opciones que no funcionan (Aplicación)	Ortografía(Aplicación)	
Validación(Aplicación)	Correspondencia con otra documentación(Documentación)	

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

Correspondencia con la documentación(Aplicación)		
--	--	--

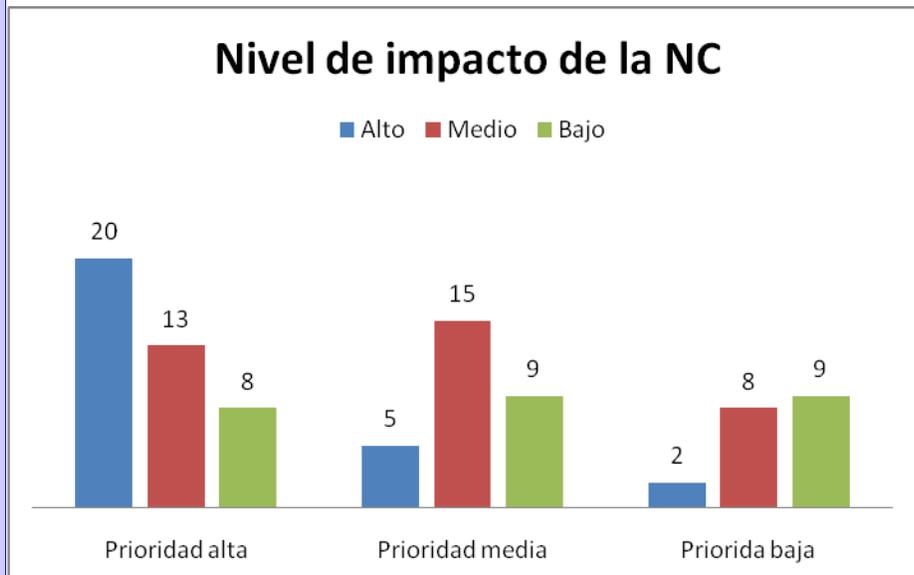
### 3.7 Indicador del nivel de impacto de una No Conformidad

<b>#3 del Objetivo de Medición</b>	Analizar el nivel de impacto que tienen las No Conformidades y como repercuten negativamente en el software.		
<b>Preguntas</b>	¿Cuál es el nivel de prioridad que tienen las No Conformidades para su posible corrección? ¿Cuál es la distribución de impacto de las no conformidades detectadas en las pruebas de liberación? ¿Cuáles son las No Conformidades que repercuten negativamente dado su nivel de impacto en el software?		
<b>Indicadores</b>			
<b>Indicador:</b> Nivel de impacto de una No Conformidad			<b>Fecha:</b>
<b>Enfocado a:</b> Desarrollador	<b>Frecuencia:</b>	Por cada	<b>13/02/2011</b>
	iteración		

\* La Prioridad de los Indicadores responde a la utilidad apreciable de estos en el momento de responder las preguntas planteadas. No construir Indicadores innecesarios, implica (probablemente) la necesidad de recopilar menos datos posteriormente. El rango previsto se interpreta de la siguiente manera: 1: Muy útil. 2: Útil. 3: Poco o Nada útil.

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

### Gráfico



**Medidas Base y/o Derivadas implicadas**

- Medida Base:** Cantidad de No Conformidades de impacto alto.
- Medida Base:** Cantidad de No Conformidades de impacto medio.
- Medida Base:** Cantidad de No Conformidades de impacto bajo.
- Medida Base:** Nivel de Prioridad alta.
- Medida Base:** Nivel de Prioridad media.
- Medida Base:** Nivel de Prioridad baja.

**Algoritmo de obtención**

Datos y Ejes representados:

Eje Horizontal: Se listan los niveles de prioridad alto, medio y bajo en los que se analiza las no conformidades para su corrección dado su impacto.

Eje Vertical: Se listan los valores entre los cuales oscilan las cantidades de No Conformidades detectadas por impacto.

**Guías para la** Cuando la línea de puntos tiene una pendiente negativa, se infiere que en

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

<b>interpretación</b>	<p>el transcurso del tiempo se le ha dado un seguimiento satisfactorio a las no conformidades detectadas de impacto alto, medio y bajo.</p> <p>Cuando la línea de puntos tiene pendiente positiva, se infiere que en el transcurso del tiempo no se le ha dado un seguimiento satisfactorio a las no conformidades detectadas de impacto alto, medio y bajo.</p> <p>Caso ideal: Cuando la línea de puntos coincide con el eje cero horizontal, se infiere que en la iteración en cuestión no se encontraron no conformidades. En este caso la organización sigue los procesos de desarrollo y esto incide positivamente en los productos que elaboran.</p>
-----------------------	--

### 3.8 Comportamiento de la No Conformidad

<b>#4 Objetivo de Medición</b>	Analizar el comportamiento y la Cantidad de No Conformidades detectadas en las Pruebas de liberación.		
<b>Preguntas</b>	<p>¿Cuál es el aumento o decremento en el cantidad de no conformidades?</p> <p>¿Qué departamentos influyen negativamente en el proceso de prueba de liberación?</p>		
<b>Indicadores</b>			
<b>Indicador:</b> Comportamiento de la No Conformidad			<b>Fecha:</b>
<b>Enfocado a:</b> Desarrollador	<b>Frecuencia:</b> Por cada iteración.	Por cada	23/02/2011

\* La Prioridad de los Indicadores responde a la utilidad apreciable de estos en el momento de responder las preguntas planteadas. No construir Indicadores innecesarios, implica (probablemente) la necesidad de recopilar menos datos posteriormente. El rango previsto se interpreta de la siguiente manera: 1: Muy útil. 2: Útil. 3: Poco o Nada útil.

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

<b>Gráfico</b>	<table border="1" style="margin-top: 10px;"> <caption>Datos del Gráfico</caption> <thead> <tr> <th>Iteración</th> <th>Alto</th> <th>Medio</th> <th>Bajo</th> </tr> </thead> <tbody> <tr> <td>Iteración1</td> <td>75</td> <td>30</td> <td>45</td> </tr> <tr> <td>Iteración2</td> <td>28</td> <td>25</td> <td>35</td> </tr> <tr> <td>Iteración3</td> <td>12</td> <td>18</td> <td>28</td> </tr> <tr> <td>Iteraciónn(n..)</td> <td>5</td> <td>10</td> <td>15</td> </tr> </tbody> </table>	Iteración	Alto	Medio	Bajo	Iteración1	75	30	45	Iteración2	28	25	35	Iteración3	12	18	28	Iteraciónn(n..)	5	10	15
Iteración	Alto	Medio	Bajo																		
Iteración1	75	30	45																		
Iteración2	28	25	35																		
Iteración3	12	18	28																		
Iteraciónn(n..)	5	10	15																		
<b>Medidas Base y/o implicadas</b>	<p><b>Medida Base:</b> Cantidad de NC por cada iteración</p> <p><b>Medida Derivada:</b>          Media de NC por cada Iteración <math>(n) = \frac{\sum \text{Cantidad de NC detectadas en los escenarios}}{\text{Cantidad de NC detectadas para esa iteración}}</math>.</p>																				
<b>Algoritmo de obtención</b>	<p>Datos y Ejes representados:</p> <p>Eje Horizontal: Se listan las iteraciones en las que se analiza el comportamiento de las no conformidades.</p> <p>Eje Vertical: Se listan los valores entre los cuales oscilan las medias de las cantidades de no conformidades detectadas por impacto alto, medio y bajo.</p>																				

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

<b>Guías para la interpretación</b>	<p>Cuando la línea de puntos tiene una pendiente negativa, se infiere que en el transcurso del tiempo se le ha dado un seguimiento satisfactorio a las no conformidades encontradas.</p> <p>Cuando la línea de puntos tiene pendiente positiva, se infiere que en el transcurso del tiempo no se le ha dado un seguimiento satisfactorio a las no conformidades encontradas.</p> <p>Caso ideal: Cuando la línea de puntos coincide con el eje cero horizontal, se infiere que en la iteración en cuestión no se encontraron no conformidades. En este caso la organización sigue los procesos de desarrollo y esto incide positivamente en los productos que elaboran.</p>
<b>Referencia</b>	Este indicador está vinculado con el del nivel de impacto de una NC.

### 3.9 Madurez de un probador en detectar una No Conformidad

<b>No. del Objetivo de Medición</b>	Analizar la madurez que un probador tiene en detectar una No Conformidad correctamente en los diferentes escenarios asignados.		
<b>Preguntas</b>	¿Cuáles son los probadores que influyen positivamente en el proceso de prueba?		
<b>Indicadores</b>			
<b>Indicador:</b> Madurez de un probador en detectar una No Conformidad			<b>Fecha:</b>
<b>Enfocado a:</b> Probador	<b>Frecuencia:</b> Por cada	Iteración	04/03/2011

\* La Prioridad de los Indicadores responde a la utilidad apreciable de estos en el momento de responder las preguntas planteadas. No construir Indicadores innecesarios, implica (probablemente) la necesidad de recopilar menos datos posteriormente. El rango previsto se interpreta de la siguiente manera: 1: Muy útil. 2: Útil. 3: Poco o Nada útil.

	<h3 style="text-align: center;">Madurez del probador en detectar una No Conformidad</h3> <p style="text-align: center;">■ Cantidad de NC verificadas ■ Total de NC en las Pruebas de liberación</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>Probador</th> <th>Cantidad de NC verificadas</th> <th>Total de NC en las Pruebas de liberación</th> </tr> </thead> <tbody> <tr> <td>Probador 1</td> <td>15</td> <td>27</td> </tr> <tr> <td>Probador 2</td> <td>6</td> <td>15</td> </tr> <tr> <td>Probador (n)</td> <td>2</td> <td>8</td> </tr> </tbody> </table>	Probador	Cantidad de NC verificadas	Total de NC en las Pruebas de liberación	Probador 1	15	27	Probador 2	6	15	Probador (n)	2	8
Probador	Cantidad de NC verificadas	Total de NC en las Pruebas de liberación											
Probador 1	15	27											
Probador 2	6	15											
Probador (n)	2	8											
<p><b>Medidas y/o implicadas</b></p>	<p><b>Medida Base:</b> Cantidad de NC detectadas verificadas.</p> <p><b>Medida Derivada:</b>                  Cantidad de NC detectadas verificadas/Cantidad de NC detectadas en la Prueba de liberación por cada probador.</p> <p>Media no válidas = <math>\frac{\sum \text{Promedio verificadas por probador}}{\text{Cantidad Probadores}}</math>.</p>												
<p><b>Algoritmo de obtención</b></p>	<p>Datos y Ejes representados:</p> <p>Eje Horizontal: Se listan los probadores que detectan las No Conformidades y el Jefe del Equipo de prueba la pone en el estado de verificada.</p> <p>Eje Vertical: Se listan los valores entre los cuales oscilan las medias de</p>												

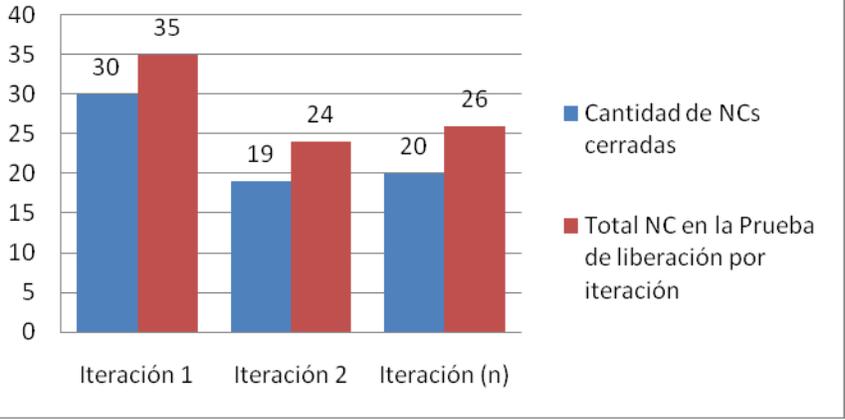
## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

	las cantidades de No Conformidades detectadas verificadas.
<b>Guías para la interpretación</b>	<p>Cuando la línea de puntos tiene una pendiente negativa, se infiere que en el transcurso del tiempo no se han detectado satisfactoriamente las no conformidades.</p> <p>Cuando la línea de puntos tiene pendiente positiva, se infiere que en el transcurso del tiempo se han dado detectado satisfactoriamente las no conformidades.</p> <p>Caso ideal: Cuando la línea de puntos coincide con el eje cero horizontal, se infiere que en la iteración en cuestión no se encontraron no conformidades. En este caso la organización sigue los procesos de desarrollo y esto incide positivamente en los productos que elaboran.</p>

### 3.10 Nivel de completitud para corregir la NC detectada

<b>No. del Objetivo de Medición</b>	Analizar el nivel de cumplimiento de las No conformidades detectadas en las pruebas de liberación con el fin de controlarlas.	
<b>Preguntas</b>	<p>¿Cuál es la diferencia entre el total de no Conformidades detectadas y lo avanzado en las cerradas?</p> <p>¿Cuál es la cantidad de no conformidades cerradas correspondiente al hito actual?</p> <p>¿En cuánto afecta el nivel de completitud de las no conformidades cerradas para los compromisos del grupo de calidad?</p>	
<b>Indicadores</b>		
<b>Indicador:</b>	Nivel de completitud para corregir la NC detectada.	<b>Fecha:</b>

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

<b>Enfocado a:</b> Desarrollador.	<b>Frecuencia:</b> Por cada iteración. <b>21/03/2011</b>												
	<div style="text-align: center;"> <h3>Nivel de completitud de la No Conformidad detectada</h3>  <table border="1" style="margin: auto;"> <thead> <tr> <th>Iteración</th> <th>Cantidad de NCs cerradas</th> <th>Total NC en la Prueba de liberación por iteración</th> </tr> </thead> <tbody> <tr> <td>Iteración 1</td> <td>30</td> <td>35</td> </tr> <tr> <td>Iteración 2</td> <td>19</td> <td>24</td> </tr> <tr> <td>Iteración (n)</td> <td>20</td> <td>26</td> </tr> </tbody> </table> </div>	Iteración	Cantidad de NCs cerradas	Total NC en la Prueba de liberación por iteración	Iteración 1	30	35	Iteración 2	19	24	Iteración (n)	20	26
Iteración	Cantidad de NCs cerradas	Total NC en la Prueba de liberación por iteración											
Iteración 1	30	35											
Iteración 2	19	24											
Iteración (n)	20	26											
<b>Medidas y/o implicadas</b>	<p><b>Medida Base:</b> Cantidad de NC cerradas.</p> <p><b>Derivada:</b> Cantidad de NC cerradas por iteración/ Cantidad de NC en la prueba de PL por iteración.</p> <p>La media de la Cantidad de NC cerradas por iteraciones = <math>\Sigma</math> Cantidad de NC cerradas por iteraciones/Cantidad de iteraciones.</p> <p>Cantidad de NC cerradas por iteraciones /Cantidad de NC en la prueba de PL del modulo.</p> <p>Cantidad de NC cerradas por módulos /Cantidad de NC detectadas en la prueba de PL del total de módulos.</p>												
<b>Algoritmo de</b>	Datos y Ejes representados:												

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

<b>obtención</b>	<p>Eje Horizontal: Se listan los probadores que detectan las No Conformidades y el Jefe del Equipo de prueba la pone en el estado de verificada.</p> <p>Eje Vertical: Se listan los valores entre los cuales oscilan las medias de las cantidades de No Conformidades detectadas verificadas.</p>
<b>Guías para la interpretación</b>	<p>Cuando la línea de puntos tiene una pendiente negativa, se infiere que en el transcurso del tiempo no se le ha dado completitud satisfactoriamente a las no conformidades.</p> <p>Cuando la línea de puntos tiene pendiente positiva, se infiere que en el transcurso del tiempo se ha dado completitud satisfactoriamente a las no conformidades.</p> <p>Caso ideal: Cuando la línea de puntos coincide con el eje cero horizontal, se infiere que en la iteración en cuestión no se encontraron no conformidades. En este caso la organización sigue los procesos de desarrollo y esto incide positivamente en los productos que elaboran.</p>

### 3.11 Tiempo de respuesta del desarrollador para una No conformidad

<b>No. del Objetivo de Medición</b>	Analizar en que tiempo un desarrollador responde a una No Conformidad.	
<b>Preguntas</b>	¿Cuál es el tiempo que un desarrollador necesita para darle respuesta a una no conformidad?	
<b>Indicadores</b>		
<b>Indicador:</b>	Tiempo de respuesta del desarrollador para una No conformidad	<b>Fecha:</b>

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

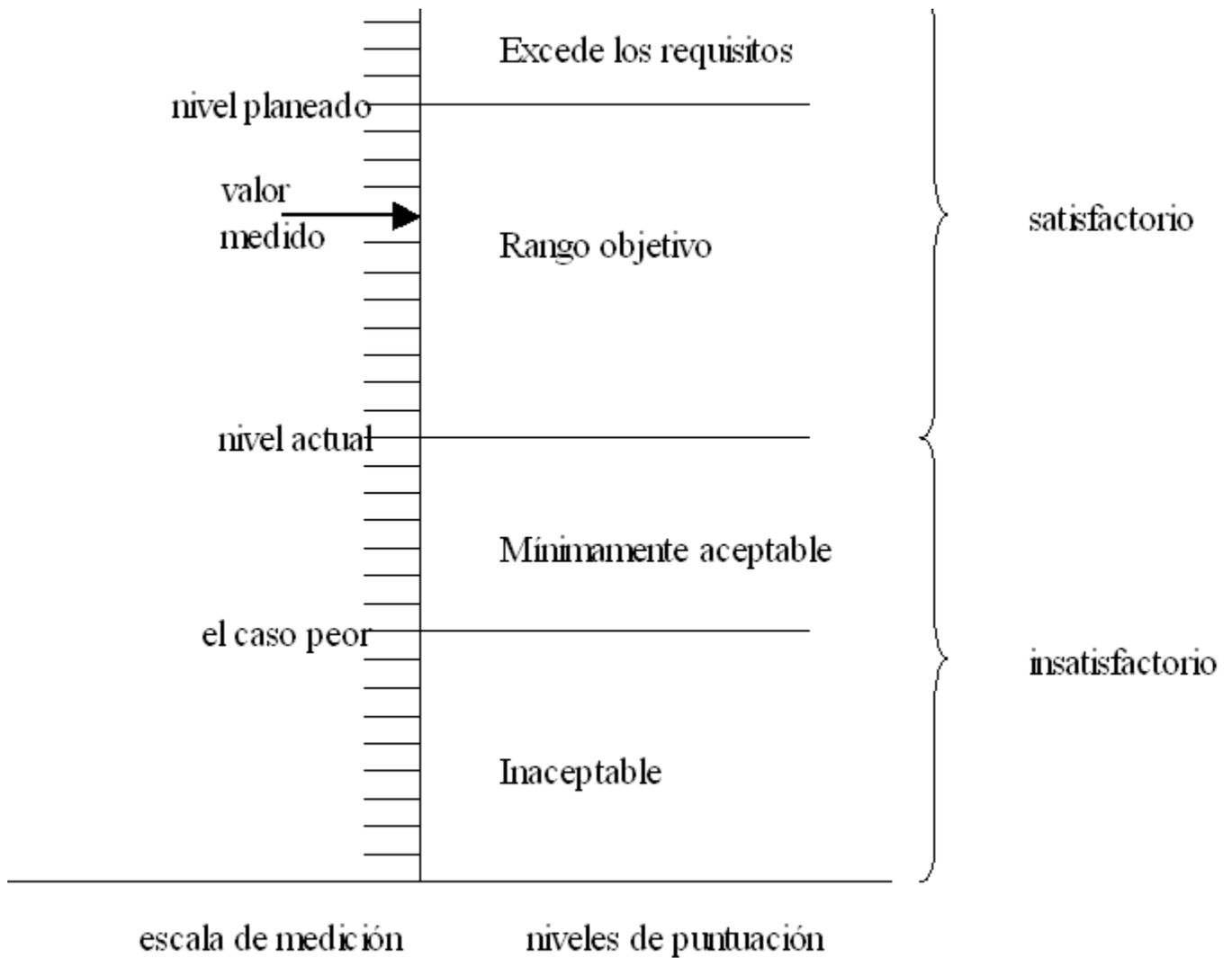
<b>Enfocado a:</b> Desarrollador	<b>Frecuencia:</b> Por cada iteración	03/04/2011												
<b>Gráfico</b>	<table border="1" style="margin-top: 10px;"> <caption>Tiempo de respuesta del desarrollador para una NC</caption> <thead> <tr> <th>Iteración</th> <th>Tiempo de NC en ser resuelta (horas/hombre)</th> <th>Tiempo total de la NC establecido (horas/hombre)</th> </tr> </thead> <tbody> <tr> <td>Iteración 1</td> <td>30</td> <td>72</td> </tr> <tr> <td>Iteración 2</td> <td>18</td> <td>24</td> </tr> <tr> <td>Iteración (n)</td> <td>20</td> <td>48</td> </tr> </tbody> </table>		Iteración	Tiempo de NC en ser resuelta (horas/hombre)	Tiempo total de la NC establecido (horas/hombre)	Iteración 1	30	72	Iteración 2	18	24	Iteración (n)	20	48
Iteración	Tiempo de NC en ser resuelta (horas/hombre)	Tiempo total de la NC establecido (horas/hombre)												
Iteración 1	30	72												
Iteración 2	18	24												
Iteración (n)	20	48												
<b>Medidas Base y/o Derivadas implicadas</b>	<b>Medida Base:</b> Tiempo de NC en ser resuelta <b>Medida Derivada:</b> Tiempo de NC en ser resuelta/Tiempo total de NC establecido.													
<b>Algoritmo de obtención</b>	<b>Datos y Ejes representados:</b> Eje Horizontal: Se listan las iteraciones donde se detectan las No Conformidades.													

\* La Prioridad de los Indicadores responde a la utilidad apreciable de estos en el momento de responder las preguntas planteadas. No construir Indicadores innecesarios, implica (probablemente) la necesidad de recopilar menos datos posteriormente. El rango previsto se interpreta de la siguiente manera: 1: Muy útil. 2: Útil. 3: Poco o Nada útil.

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

	<p>Eje Vertical: Se listan los valores entre los cuales oscilan las horas en que el desarrollador da respuesta a la no conformidad detectada y el tiempo total con que cuenta para dar dicha respuesta.</p>
<b>Guías para la interpretación</b>	<p>Cuando la línea de puntos tiene una pendiente negativa, se infiere que en el transcurso del tiempo se le ha dado respuesta satisfactoriamente a las no conformidades.</p> <p>Cuando la línea de puntos tiene pendiente positiva, se infiere que en el transcurso del tiempo no se ha dado respuesta satisfactoriamente a las no conformidades.</p> <p>Caso ideal: Cuando la línea de puntos coincide con el eje cero horizontal, se infiere que en la iteración en cuestión no se encontraron no conformidades. En este caso la organización sigue los procesos de desarrollo y esto incide positivamente en los productos que elaboran.</p>

Rango propuesto para el análisis del resultado de la medida derivada del indicador de medición, obtenido de la interpretación de la información asociada a la No Conformidad y que permita orientar la toma de decisiones. El rango del indicador de medición va estar entre 0 y 1 con respecto a los parámetros que se miden, las cantidades entre más se acercan a 1 el valor medido es satisfactorio y en el tiempo el valor medido si se acerca a 0 es satisfactorio. Ver figura 3.1.



**Figura 3.1** Niveles de puntuación propuesto

**Propuesta de mejora para los reportes:**

- Especificar que los reportes estén filtrados por tipo de error de la No Conformidad (NC).

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

---

- Permitir que los reportes exporten las imágenes donde se encuentran detectada la NC para que se tenga la evidencia visible de la NC en imágenes.
- Certificar que los formatos no pierdan la descripción en su configuración original.
- Establecer Listado de Reportes en el Redmine
  - Cantidad de NC detectadas No válidas por probador
  - Cantidad de NC detectadas duplicadas
  - Nivel de impacto de la NC
  - Éxito de un probador al completar una tarea
  - Rapidez del desarrollador en eliminar una NC
  - Rapidez del probador en detectar una NC
  - Comportamiento de la NC

### Reportes

Se definen como **campos** (que serán leídos), **valores** (por los cuáles se filtrará) y **columnas** (que se mostrarán en la tabla resultado del reporte).

Nombre reporte: Cantidad de NC detectadas No válidas por probador.

Campos:

- Secuencia

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

---

- Autor

Valores:

- Secuencia = *Nueva-Reabierta -> NC no válida -> Verificada*
- Autor = 'nombre Probador'

Columnas:

- Secuencia
- Autor

Nombre reporte: Cantidad de NC detectadas duplicadas por probador.

Campos:

- Secuencia
- Autor

Valores:

- Secuencia = *Nueva-Reabierta -> NC duplicada-> Verificada*
- Autor = 'nombre Probador'

Columnas:

- Secuencia
- Autor

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

### 3.12 Propuesta de configuración del Redmine

El Redmine debe tener campos tipificados para las No conformidades de liberación:

- Debe tener un campo tipificado con la secuencia cronológica de los estados de la No Conformidad. Ver la tabla 3.7.

No Confirmada	Nueva	Resuelta	Verificada	Cerrada
---------------	-------	----------	------------	---------

**Tabla 3.7:** Ejemplo de secuencia de antecedentes cronológicos de una No conformidad. Debe tener un campo tipificado con la localización de la No Conformidad. Ver figura propuesta 3.2.

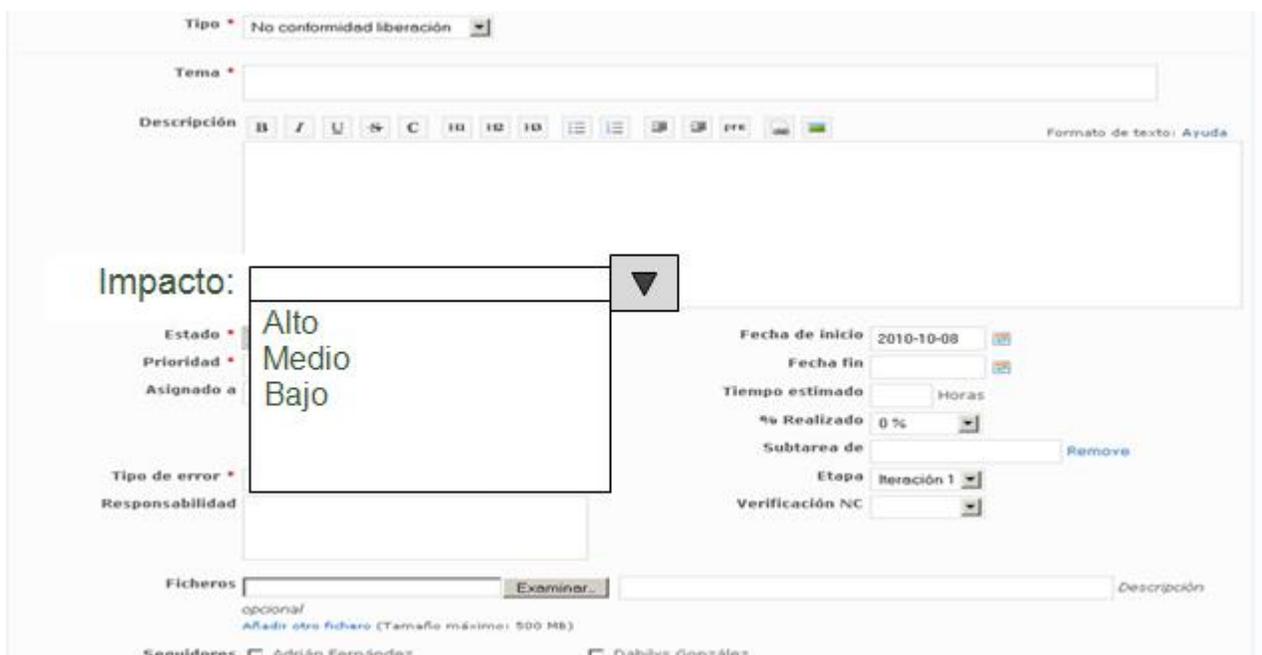
The image shows a web form for creating a release non-conformity in Redmine. The form is titled 'No conformidad liberación'. It includes several fields: 'Tema' (empty), 'Descripción' (with a rich text editor toolbar), 'Estado' (set to 'Nueva'), 'Prioridad' (set to 'Normal'), 'Asignado a' (empty), 'Fecha de inicio' (set to '2010-10-08'), 'Fecha fin' (empty), 'Tiempo estimado' (empty), '% Realizado' (set to '0 %'), 'Subtarea de' (empty), 'Etapa' (set to 'Iteración 1'), 'Verificación NC' (empty), 'Tipo de error' (set to '-- Por favor seleccione --'), 'Responsabilidad' (empty), and 'Ficheros' (empty). A text box labeled 'Localización:' contains the text 'Acápite o Camino'. There is also a 'Remove' button next to the 'Subtarea de' field.

**Figura 3.2** Localización para la No Conformidad.

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

El acápite es para los defectos detectados en la documentación del software y el camino si es un defecto detectado en la aplicación.

Debe tener un campo tipificado con el nivel de impacto de la No Conformidad. Ver figura propuesta **Figura 3.2** Nivel de impacto para la No Conformidad.



The image shows a screenshot of a Bugzilla form for creating a new bug. The form is titled 'No conformidad liberación'. The 'Impacto' field is highlighted with a red box, and a dropdown menu is open, showing three options: 'Alto', 'Medio', and 'Bajo'. The form includes various fields for metadata, such as 'Fecha de inicio', 'Fecha fin', 'Tiempo estimado', '% Realizado', 'Subtarea de', 'Etapo', and 'Verificación NC'. There is also a section for 'Ficheros' (Attachments) and a 'Formato de texto: Ayuda' link.

### 3.13 Clasificación de estados de acuerdo al error detectado en el software

- La herramienta de Bugzilla brinda buenas prácticas para el seguimiento del Bug, en todo su ciclo de vida por lo que se propone como guía para los estados de las No conformidades en el Proceso de prueba actual. Traduciendo lo que es **RESOLVED** sea el **No procede** con todas sub-características **INVALID** (no válida), **WONTFIX** (no se resolverá), **WORKSFORME** (No se puede

reproducir) DUPLICATE (duplicada) y que FIXED (resuelta) no se tome como un sub-capo sino un campo como tal en los estados de una No Conformidad.

### 3.14 Estados de un Bug en Bugzilla

- Estados de un Bug en Bugzilla adaptados al proceso actual:
- NEW: El Bug ha sido registrado por parte del Equipo de Prueba (EP).
- UNCONFIRMED: El Bug ha sido registrado pero no ha sido considerado como tal.
- ASIGNED: El Bug ha sido asignado a un miembro del Equipo de Desarrollo(ED) para su resolución.
- RESOLVED: El Bug ha sido procesado por el ED y aún no se ha verificado por el EP.
- FIXED: El Bug ha sido resuelto por el ED.
- INVALID: El ED no está de acuerdo con la validez del bug.
- WONTFIX: El ED está de acuerdo con la validez del bug pero no considera necesario resolverlo por lo menos en el presente.
- WORKSFORME: El ED no ha podido reproducir el bug.
- DUPLICATE: El EP o el ED considera que el bug es un duplicado de un bug ya registrado.
- VERIFIED: El EP ha verificado y está conforme con el estado de RESOLVED especificado por el ED.
- REOPEN: El EP ha verificado y no está conforme con el estado de RESOLVED especificado por el ED.
- CLOSED: El Bug ha sido resuelto y ya no se tiene en cuenta.

### 3.15 Estados de la Propuesta para las No Conformidades de liberación

No confirmada: La No Conformidad (NC) ha sido registrada por el probador (estudiante) pero aún no es visible para el Equipo de desarrollo (ED) porque no ha sido aprobada por el Jefe de pruebas (profesor).

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

---

Nueva: La NC ha sido aprobada por el jefe de pruebas, es visible para el ED y está en espera de que sea corregida.

Reabierto: La NC fue tratada por el ED pero al verificar el equipo de pruebas (EP) considera que la respuesta no es correcta. Análogo al No resuelta pero es notificado por el EP.

Verificada: La NC ha sido respondida por el ED y al verificar el EP considera que la respuesta es correcta.

Resuelta: La NC ha sido respondida por el ED y la han notificado como resuelta. No ha sido verificada por el EP.

No válida: La NC ha sido respondida por el ED pero consideran que no es una NC o no están de acuerdo con la NC. Es una de las causas del No procede.

No se resolverá: La NC ha sido respondida por el ED y consideran que efectivamente es una NC pero por alguna razón no la van a resolver. Es otra de las causas del No procede.

No se pudo reproducir: La NC no pudo ser encontrada por el ED. Es otra de las causas del No procede.

Duplicada: El ED o el EP han detectado una NC como duplicada. Es otra de las causas del No procede.

Cerrada: La NC ha sido resuelta y ya no se tiene en cuenta.

Ver figura 3.4.

## CAPÍTULO 3. PROPUESTA DE SOLUCIÓN

The image shows a web-based form for managing non-conformities. At the top, there is a dropdown menu for 'Tipo' (Type) set to 'No conformidad liberación'. Below it is a 'Tema' (Topic) field. The main area is a 'Descripción' (Description) field with a rich text editor toolbar. On the left, there is a sidebar with labels for 'Estado', 'Estado', 'Prioridad', 'Asignado a', 'Tipo de error', 'Responsabilidad', 'Ficheros', and 'Seguidores'. The 'Estado' dropdown menu is open, showing the following options: 'No confirmada', 'Nueva', 'Reabierto', 'Verificada', 'Resuelta', 'No válida', 'No se resolverá', 'No se pudo reproducir', 'Duplicada', and 'Cerrada'. On the right side, there are several input fields: 'Fecha de inicio' (2010-10-08), 'Fecha fin', 'Tiempo estimado' (with a 'Horas' unit), '% Realizado' (0%), 'Subtarea de' (with a 'Remove' button), 'Etapa' (Reacción 1), and 'Verificación NC'. At the bottom right, there is a 'Descripción' field and the name 'Dahilys González'.

**Figura 3.4** Estados propuestos para la No Conformidad.

En este capítulo se elaboraron los indicadores de medición y reportes asociados que van a brindar buenas prácticas en el seguimiento de la información obtenida de las No conformidades detectadas en las pruebas de liberación, para guiar el proceso de prueba y desarrollo en los diferentes departamentos del Centro de Informática Médica (CESIM). Se propusieron campos a insertar en la herramienta Redmine para ofrecer una mejor guía e interpretación de la información asociada a las No conformidades.

### CONCLUSIONES

Al concluir el presente trabajo de diploma se ha cumplido el objetivo y las tareas de la investigación. Se realizó un estudio de la situación actual del tratamiento que brinda la herramienta Redmine a las No Conformidades a nivel global y específicamente en los diferentes departamentos del Centro de Informática Médica (CESIM).

1. A partir del diagnóstico realizado se identificó que no se da seguimiento a la información asociada a las No conformidades detectadas en las pruebas de liberación. Demostrando la necesidad de una mejora de configuración a la herramienta Redmine.
2. Se propuso una solución de indicadores de medición para la obtención e interpretación de la información asociada a las No conformidades, dando seguimiento a las evidencias dejadas por las No Conformidades y así poder tomar decisiones en el proceso productivo del CESIM.
3. Se propuso un listado de reportes asociados a los indicadores de medición para interpretar el estado del proceso de pruebas.
4. Se propuso una serie de campos a tipificar en la herramienta Redmine para mejorar el seguimiento del ciclo de vida de la No conformidad.

### RECOMENDACIONES

Para futuras investigaciones, se recomienda:

- Proponer nuevos indicadores que ayuden a mejorar el trabajo con el Redmine y obtener resultados que permitan una óptima toma de decisiones
- Realizar un análisis más detallado sobre los reportes de la No Conformidad (NC) para una mejor acción correctiva sobre la causa que dio lugar a defectos encontrados en el software detectadas en las Pruebas de liberación.
- Elaborar nuevas entrevistas para la recopilación de datos referentes a las No conformidades detectas.
- Analizar nuevas herramientas en el seguimiento de errores para identificar posibles mejoras a incorporar en la herramienta Redmine.

### REFERENCIA BIBLIOGRÁFICA

1. Normas ISO 9000 y calidad. [Online] [Cited: 20 9 2010.] <http://normas-iso-9000.blogspot.com/2007/11/conformidad-y-no-conformidad.html>.
2. Calidad & Gestión. Escuchamos problemas, devolvemos soluciones. [Online] 2008 - 2010. [Cited: 5 11 2010.] [http://calidad-gestion.com.ar/boletin/41\\_tratamiento\\_no\\_conformidades.html](http://calidad-gestion.com.ar/boletin/41_tratamiento_no_conformidades.html).
3. Monografias.com. El Centro de Tesis, Documentos, Publicaciones y Recursos Educativos más amplio de la Red. [Online] <http://www.monografias.com/trabajos55/organizaciones-con-control-calidad/organizaciones-con-control-calidad.shtml>.
4. Definicion.de. [Online] 2011. <http://definicion.de>.
5. Monografias.com. [Online][Http://www.monografias.com/trabajos15/valoracion/valoracion.shtml](http://www.monografias.com/trabajos15/valoracion/valoracion.shtml).
6. Alfonso Juan López Baena, Miguel Valcarce Cases, Manuel Barbancho Medina. Indicadores Cuantitativos y Cualitativos para la Evaluación de la Actividad Investigadora: Complementarios, Contradictorios, Excluyentes. [Online] [http://www2.uca.es/serv/consejo\\_social/doc/transfencia14.pdf](http://www2.uca.es/serv/consejo_social/doc/transfencia14.pdf).
7. Informática Hoy. Información para estar actualizado y disfrutar de la informática y la tecnología. Bugzilla, la mejor herramienta para el seguimiento de errores. [Online] 2010. [www.informatica-hoy.com.ar](http://www.informatica-hoy.com.ar).
8. Seguimiento de errores. . Capítulo 3. Infraestructura Técnica. [Online] [Cited: 27 01 2011.] <http://producingoss.com/es/bug-tracker.html>.
9. ACT. Government. Environment and Sustainable Development. [Online] 31 agosto 2010. [Cited: 5 02 2011.]  
[Http://www.actpla.act.gov.au/topics/design\\_build/da\\_assessment/which\\_track\\_will\\_by\\_da\\_be\\_assessed\\_in code,\\_merit\\_or\\_impact/code\\_track](http://www.actpla.act.gov.au/topics/design_build/da_assessment/which_track_will_by_da_be_assessed_in_code,_merit_or_impact/code_track).
10. Portions of this work are. Bugzilla. [Online] 1998-2011. [Cited: 15 02 2011.] <http://www.bugzilla.org/>.

## REFERENCIA BIBLIOGRÁFICA

---

11. SlideShare Inc. All rights reserved. SlideShare is the world's largest community for sharing presentations. [Online] 2011. <http://www.slideshare.net>.
12. Mantis manual de usuario. [Online] 2011. <http://www.scribd.com/doc/52516332/30/Funcionalidades-permitidas-dentro-de-Mantis>.
13. Mantis manual de usuario. [Online] 2011. <http://www.scribd.com/doc/52516332/30/Funcionalidades-permitidas-dentro-de-Mantis>.
14. Mantis. [Online] [Cited: 25 02 2011.] <http://www.mantisbt.org/privacy.php>.

### BIBLIOGRAFÍA

1. Normas ISO 9000 y calidad. [Online] [Cited: 20 9 2010.] <http://normas-iso-9000.blogspot.com/2007/11/conformidad-y-no-conformidad.html>.
2. Calidad & Gestión. Escuchamos problemas, devolvemos soluciones. [Online] 2008 - 2010. [Cited: 5 11 2010.] [http://calidad-gestion.com.ar/boletin/41\\_tratamiento\\_no\\_conformidades.html](http://calidad-gestion.com.ar/boletin/41_tratamiento_no_conformidades.html).
3. Monografias.com. El Centro de Tesis, Documentos, Publicaciones y Recursos Educativos más amplio de la Red. [Online] <http://www.monografias.com/trabajos55/organizaciones-con-control-calidad/organizaciones-con-control-calidad.shtml>.
4. Definicion.de. [Online] 2011. <http://definicion.de>.
5. Monografias.com. [Online][Http://www.monografias.com/trabajos15/valoracion/valoracion.shtml](http://www.monografias.com/trabajos15/valoracion/valoracion.shtml).
6. Alfonso Juan López Baena, Miguel Valcarce Cases, Manuel Barbancho Medina. Indicadores Cuantitativos y Cualitativos para la Evaluación de la Actividad Investigadora: Complementarios, Contradictorios, Excluyentes. [Online] [http://www2.uca.es/serv/consejo\\_social/doc/transfereencia14.pdf](http://www2.uca.es/serv/consejo_social/doc/transfereencia14.pdf).
7. Informática Hoy. Información para estar actualizado y disfrutar de la informática y la tecnología. Bugzilla, la mejor herramienta para el seguimiento de errores. [Online] 2010. [www.informatica-hoy.com.ar](http://www.informatica-hoy.com.ar).
8. Seguimiento de errores. . Capítulo 3. Infraestructura Técnica. [Online] [Cited: 27 01 2011.] <http://producingoss.com/es/bug-tracker.html>.
9. ACT. Government. Environment and Sustainable Development. [Online] 31 agosto 2010. [Cited: 5 02 2011.]  
[Http://www.actpla.act.gov.au/topics/design\\_build/da\\_assessment/which\\_track\\_will\\_by\\_da\\_be\\_assessed\\_in\\_code,\\_merit\\_or\\_impact/code\\_track](http://www.actpla.act.gov.au/topics/design_build/da_assessment/which_track_will_by_da_be_assessed_in_code,_merit_or_impact/code_track).
10. Portions of this work are. Bugzilla. [Online] 1998-2011. [Cited: 15 02 2011.] <http://www.bugzilla.org/>.

11. SlideShare Inc. All rights reserved. SlideShare is the world's largest community for sharing presentations. [Online] 2011. <http://www.slideshare.net>.
12. Mantis manual de usuario. [Online] 2011. <http://www.scribd.com/doc/52516332/30/Funcionalidades-permitidas-dentro-de-Mantis>.
13. Mantis manual de usuario. [Online] 2011. <http://www.scribd.com/doc/52516332/30/Funcionalidades-permitidas-dentro-de-Mantis>.
14. Mantis. [Online] [Cited: 25 02 2011.] <http://www.mantisbt.org/privacy.php>.

### ANEXOS

#### Anexo #1

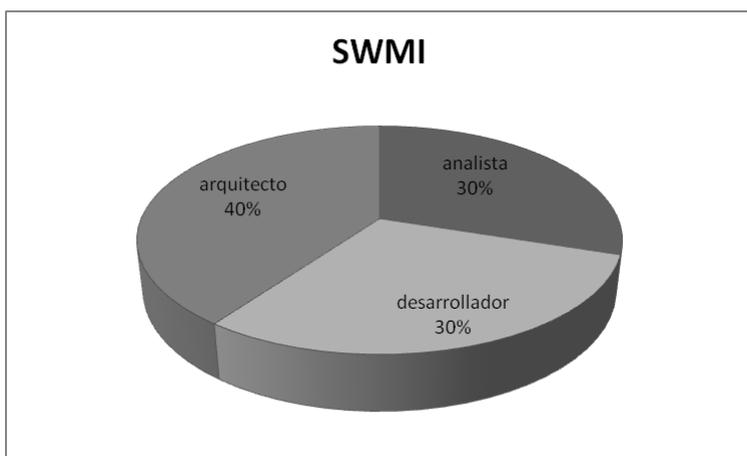
Entrevista realizada a los diferentes especialistas que cumplen con los roles de analistas, arquitectos, desarrolladores y administradores de la calidad en los diferentes departamentos productivos del Centro de Informática Médica (CESIM).

1. El proyecto a liberado algún módulo.
2. Se le ha dado seguimiento a las No Conformidades (NC) generadas en las pruebas de liberación.
3. El equipo de desarrollo analiza las NC de los módulos liberados con el fin de no incurrir en ellas.
4. En el desarrollo de su proyecto utilizan alguna guía que indiquen los errores más frecuentes en la programación.
5. Responder si tienen grupo de calidad interno. ¿Se registran en la herramienta de gestión de proyecto Redmine las NC detectadas en las revisiones internas?
6. Se clasifica la NC según su nivel de importancia con el fin de priorizar acciones correctivas encaminadas a su solución.
7. El proyecto se guía por documentos de Pautas definidos para las aplicaciones desarrolladas en el Centro.
8. Utilizan indicadores de medición de la calidad para darle solución y orientar la toma de decisiones a la información obtenida de las No Conformidades detectadas en las pruebas de liberación.
9. Se comprueban los casos de CU que estén acorde con las pautas del diseño de cada módulo antes de que se lleve al grupo de calidad para las pruebas de liberación.
10. Se revisa la ortografía del módulo antes de entregar.
11. ¿Cuáles módulos generan mayor número de no conformidades?

12. ¿Cuáles son los problemas relevantes?

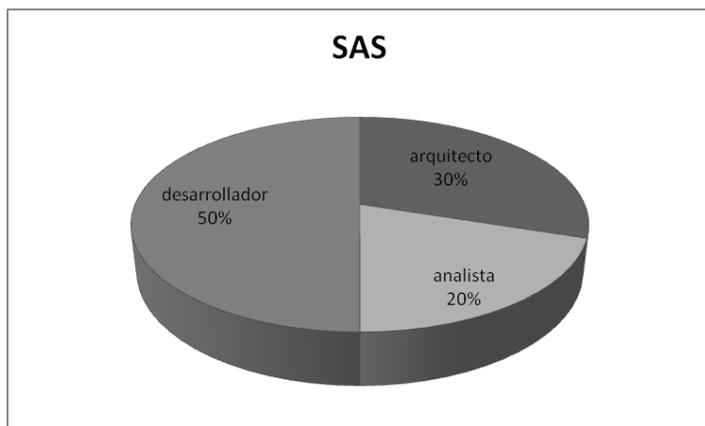
13. ¿Cuáles son las causas que generan mayor número de no conformidades?

**Anexo #2**



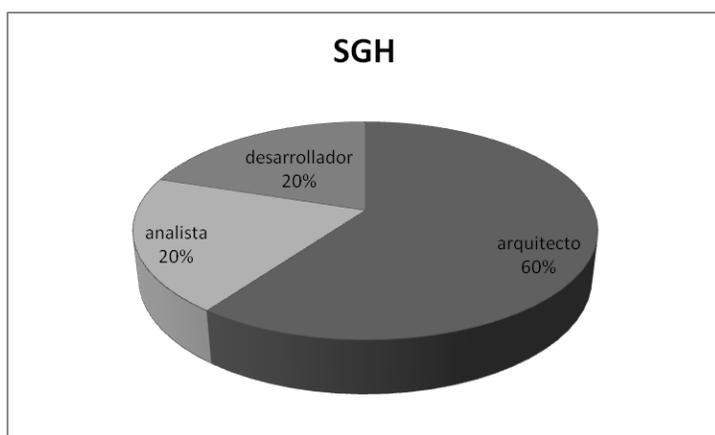
Entrevistados del Departamento de Software Médico Imagenológico. **Figura 2.1**

**Anexo #3**



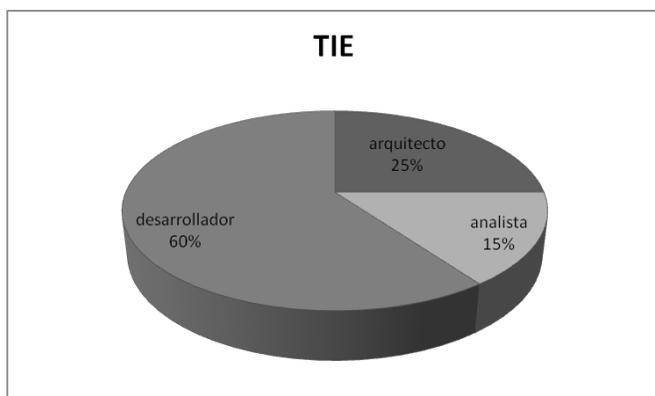
Entrevistados del Departamento de Sistema de Apoyo a la Salud. **Figura 2.2**

**Anexo #4**



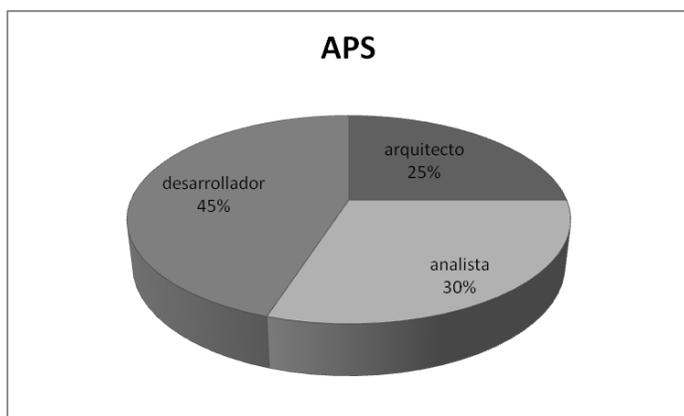
Entrevistados de Tecnologías, Integración y Estándares. **Figura 2.3**

**Anexo #5**



Entrevistados de Tecnologías, Integración y Estándares. **Figura 2.4**

Anexo #6



Entrevistados del Departamento de sistemas para la Atención Primaria de Salud **Figura 2.5**