GESTIÓN DE INCIDENTES APLICANDO ITIL – LITE EN UN CENTREX IP

Trabajo de Grado

Oscar Eduardo Muñoz Muñoz

Danilo Ibarra Narváez

Director

Mag. Andrés Lara Silva

Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Telemática
Línea de Investigación en Servicios Avanzados de Telecomunicaciones
Popayán, 2011

TABLA DE CONTENIDO

Lista de Figuras

Lista de Tablas

Introducción

Los operadores de servicios de Tecnologías de la Información (IT: Information Technologies) con el fin de generar servicios de valor agregado para satisfacer las necesidades actuales y futuras de la sociedad, así mismo para crear nuevas fuentes de ingresos y competitividad, se han visto en la necesidad de encontrar una forma de gestionar sus servicios de forma eficiente e inocua para el usuario final [1], surgiendo en el mercado multitud de equipos, técnicas, estándares, tecnologías y protocolos [2], como la Gestión de Servicios IT (ITSM: IT Service Management). ITSM ha sido una de las áreas más investigadas en el desarrollo de metodologías y herramientas que facilitan proveer alta calidad del servicio con máxima eficiencia [1]. Dentro de estas metodologías y herramientas emergió lo que hoy se conoce como Librería de la Infraestructura de IT (ITIL: Information Technology Infrastructure Library), que consiste en aplicar las mejores prácticas en la gestión de infraestructuras IT para la entrega exitosa de servicios, llegando a ser un estándar de facto, cuyo objetivo fundamental es el de generar valor a un negocio [3], actualmente está implementada por grandes empresas multinacionales y nacionales, pero aún no se utiliza en gran medida en las MIPYME (MIPYME: Micro, Pequeñas y Medianas Empresas), a pesar de que éstas tienen un gran potencial para usarlo.

La adopción de ITIL, debería ser el objetivo inicial de cualquier organización durante la puesta en marcha de un nuevo proyecto, y se debería observar hasta qué punto las MIPYME pueden implementarla, pues esta supone inversiones en tiempo y dinero que la mayoría de las organizaciones no disponen o las consideran inapropiadas. De esta manera es necesario buscar la forma de incrementar la productividad IT y aumentar el nivel del servicio al usuario final sin grandes costos de implementación, proponiéndose una infraestructura de gestión unificada para negocios, denominada ITIL-LITE, la cual busca aplicar ITIL de manera más simple y eficiente. ITIL-LITE tiene en su núcleo los servicios IT y alrededor los procesos de gestión, estos procesos buscan dar solución a fallas críticas en las operaciones del día a día [3].

Dentro de los procesos definidos por ITIL-LITE, la Gestión de Incidentes tiene como objetivo resolver cualquier incidente que cause una interrupción en el servicio de la manera más rápida y eficaz posible [4], los incidentes que se deben afrontar pueden ser del tipo aplicación, como fallas en los programas o bases de datos, de tipo hardware como que la impresora no está funcionando y del tipo pedido de servicio, como acceso denegado [5]. La solución de este tipo de incidentes es uno de los temas más difíciles que la operación del servicio debe afrontar diariamente.

Es por esta razón que el presente trabajo tiene por objetivo diseñar una solución que sirva para la implementación de un prototipo tal que permita realizar la Gestión de Incidentes en los servicios de un Centrex IP[1], utilizando los conceptos de ITIL-LITE. Para este fin se estudian distintas tecnologías de gestión de servicios IT existentes, se escoge la más apropiada según los requisitos del sistema y las necesidades existentes que se hayananalizado. Se construye el prototipo haciendo uso de herramientas libres (Open Source), finalmente se gestionan algunos servicios del Centrex IP.

[1] Centrex IP: Central telefónica remota alojada por el proveedor de servicios de Voz sobre IP.

El presente documento está estructurado en cuatro capítulos:

El primer capítulo describe toda la base teórica de los temas de gestión referentes para el desarrollo del proyecto; además se presentan las generalidades de las tecnologías de gestión, definiéndola más apropiada para soportar la gestión de incidentes.

El segundo capítulo contiene las especificaciones de diseño de la solución, en las cuales se aplican todos los fundamentos vistos en el capítulo anterior y se obtiene un diagrama final que sirve como base para desarrollar la aplicación de Gestión de Incidentes.

El tercer capítulo consta de la descripción del prototipo. En este capítulo se describen las diferentes decisiones de implementación y desarrollo, esto involucra modelos descriptivos, diagramas correspondientes, tecnologías empleadas, herramientas de implementación, etc.

El cuarto capítulo muestra las funcionalidades ya implementadas en un prototipo, estas funcionalidades se verifican realizando pruebas en el sistema por medio de la generación de un incidente en uno de los servicios del Centrex IP.

El quinto capítulo contiene las conclusiones respectivas como resultado de la realización del proyecto, tanto a nivel de desarrollo como a nivel teórico y de la utilización de herramientas, de igual manera, contiene las recomendaciones producto de la experiencia adquirida, que puedan ser aplicadas en el desarrollo de trabajos futuros.

Finalmente se presenta la Bibliografía, la cual contiene la referencia a todos los documentos o parte de ellos que fueron consultados como fuente de conocimiento en el desarrollo del proyecto, también se presentan a lo largo del documento una serie de notas y pie de páginas acerca de los términos que deban ser tenidos en cuenta para un mejor entendimiento de la documentación.

Capítulo 1. Generalidades, Fundamentos y Definición de la Tecnología de Gestión.

1.1 ¿Qué es un servicio IT?

Es un servicio proporcionado a uno o más Clientes por un Proveedor de Servicios de IT. Un Servicio de IT se basa en el uso de las Tecnologías de la Información y soporta los Procesos de Negocio del Cliente. Un Servicio de IT se compone de una combinación de personas, procesos y tecnologías [6].

1.1.1 ¿Qué es una Aplicación?

Un programa informático que lleva a cabo una función con el objeto de ayudar a un usuario a realizar una determinada actividad. WWW, FTP, correo electrónico y Telnet son ejemplos de aplicaciones en el ámbito de Internet [7]. Esta se diferencia principalmente de otros tipos de programas como los sistemas operativos, ya que estos hacen funcionar al computador. Otro tipo de programas son las utilidades, que realizan tareas de mantenimiento o de uso general, y por último los lenguajes de programación, con el cual se crean los programas informáticos. Una Aplicación es una solución informática para la automatización de tareas complejas como pueden ser la simulación de sistemas electrónicos, la redacción de documentos, o la gestión de una red.

Algunas aplicaciones hechas a la medida suelen ofrecer un gran potencial, ya que se diseñan exclusivamente para resolver un problema específico. Otros, llamados paquetes integrados de software, ofrecen menos potencial pero a cambio incluyen varias aplicaciones, como un programa procesador de textos o una base de datos.

1.1.2 Gestión de Servicios

El origen de la Gestión de Servicios se encuentra en la experiencia adquirida por las empresas de servicios tales como bancos, líneas aéreas y compañías de telefonía al gestionar sus negocios. Sus prácticas han mejorado con el tiempo, ya que las organizaciones de IT han adoptado un enfoque orientado al servicio, para administrar aplicaciones, infraestructuras y procesos de IT.

La Gestión del Servicio es un conjunto de capacidades organizacionales especializadas en dar valor al cliente en forma de servicios [8]. Según ITIL estas capacidades organizacionales se transforman en funciones y procesos para administrar los servicios IT a lo largo de su ciclo de vida, entrando en énfasis en la estrategia, diseño, transición, operación y mejora continua del servicio. Transformar recursos en servicios de valor agregado es el núcleo de la Gestión del Servicio IT. Adoptar buenas prácticas puede ayudar a un proveedor de servicios a ser eficiente en la gestión de sus servicios, las buenas prácticas es simplemente aplicar las cosas que se han comprobado que funcionan eficazmente. Estas prácticas vienen de distintas fuentes como ITIL.

1.1.3 Servicios de Telefonía IP

En este proyecto se hace uso de las capacidades de la voz sobre IP, la cuales una tecnología que permite realizar llamadas telefónicas sobre redes LAN (LAN: Local Area Network), internet, en general redes de computadores [9]. La tecnología de VoIP convierte la voz analógica en paquetes de datos digitales que soportan la comunicación sobre IP y que emplea protocolos para aplicaciones en tiempo real como el RTP (RTP: Real Time Protocol).

Así el servicio de VoIP sustituye el tradicional teléfono residencial por un teléfono IP que utiliza las redes de computadores para hacer y recibir llamadas. Con el servicio de VoIP, se puede hacer una llamada IP desde un computador, desde un teléfono especial IP o desde un teléfono tradicional con adaptador. Esta tecnología permite realizar más de una llamada telefónica simultáneamente. Además las llamadas entrantes pueden ser automáticamente dirigidas al teléfono IP independiente de dónde se esté conectado a la red. Los operadores de servicios de VoIP pueden hacer todas las funciones de las tradicionales Redes Públicas de Telefonía Conmutada (PSTN: Public Switched Telephone Network) y estos servicios de valor agregado se cargan con una tarifa extra.

1.1.4 Service Desk

El Service Desk es un sistema que se implementa en las organizaciones con el objetivo de dar soporte a los usuarios, a medida que requieran ayuda para hacer uso de los servicios IT, también monitoriza el ambiente de IT para el cumplimiento de niveles predeterminados de servicio y escalar las incidencias de manera adecuada. Está compuesto por software, hardware y un grupo de personas que recogen todo tipo de peticiones e incidencias y que tienen la destreza técnica para contestar a prácticamente cualquier pregunta o queja [10].

1.2 ITIL- Information Technology Infrastructure Library

Se define ITIL como un marco de trabajo de las buenas prácticas destinadas a facilitar entrega de servicios de IT [11]. ITIL resume un extenso conjunto de procedimientos de gestión ideados para ayudar a las organizaciones a lograr calidad y eficiencia en las operaciones de IT. Está definida en un conjunto de libros, en los cuales se encuentran documentados todos los procesos referentes a la provisión de servicios de tecnologías de la información hacia las organizaciones [12].

¿Porque usar ITIL? es el enfoque más ampliamente aceptado para la dirección de servicios IT en el mundo entero, debido a sus mejores prácticas, tanto para sectores públicos como privados [13]. En una encuesta realizada a diferentes organismos IT, se les preguntó si conocían las recomendaciones o marcos de referencia descritas a continuación, dando como resultado la Figura 1.

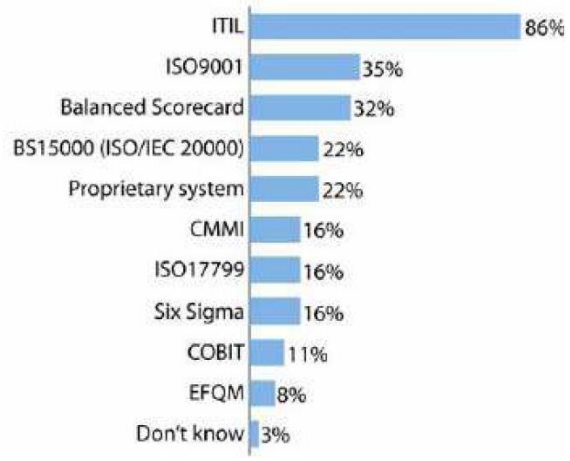


Figura 1. Principales Marcos de Trabajo
Fuente: Xelere IT Optimization

Actualmente ITIL se encuentra en su tercera versión (ITIL v3), sin embargo en junio de 2009 la OGC (OGC: Office of Government Commerce) en el ITSMF (ITSMF: ITSM Forum) Internacional, anunció que realizó una encuesta a la comunidad de gestión de servicios IT, para obtener opiniones sobre la adopción de ITIL versión 3 y la necesidad de mantener ITIL versión 2. La encuesta tenía más de 1300 respuestas, con una amplia gama de organizaciones que proporcionan sus puntos de vista [14]. La gran mayoría de las organizaciones cuyo personal respondió a las encuestas que se dieron a conocer en Australia por el ITSMF [15], dicen que han adoptado ITIL y están haciendo importantes progresos en la aplicación de este marco. Dando prioridad a la aplicación del ServiceDesk, Gestión del Cambio y los procesos de Gestión de incidentes.

1.2.1 ITIL – LITE

Se observa ITIL – LITE como una infraestructura unificada de procesos basada en ITIL v3 [16], que busca aplicar ITIL de manera más simple y eficiente para adecuarlo a las MIPYME, además se apoya en herramientas que permiten la automatización de los procesos para alcanzar este fin.

ITIL–LITE dentro de la Operación del Servicio utiliza las siguientes áreas de gestión:

1.2.1.1 Gestión de Incidentes

Es el proceso para hacer frente a todos los incidentes, lo que puede incluir fallas, dudas o consultas reportadas por los usuarios. La Gestión de Incidentes busca resolver cualquier incidente que cause una interrupción en el servicio de la manera más rápida y eficaz posible [17]. Donde un incidente es cualquier evento que no sea parte de la operación estándar de un servicio, que ocasione, o pueda ocasionar, una interrupción inesperada o una reducción en la calidad del servicio. La Gestión de Incidentes no debe confundirse con la Gestión de Problemas, pues a diferencia de esta última, no se preocupa por encontrar y analizar las causas subyacentes a un determinado incidente sino exclusivamente a restaurar el servicio. Sin embargo, es obvio, que existe una fuerte interrelación entre ambas [4]. Algunos de los objetivos principales para

garantizar la calidad del servicio con Gestión de Incidentes en ITIL v3 son: identificación del incidente, registrar el incidente, categorización de la incidencia, entre otros.

1.2.1.2 Gestión de Problemas

Cuando algún tipo de incidente se convierte en recurrente o tiene un fuerte impacto en la infraestructura IT, es función de la Gestión de Problemas determinar sus causas y encontrar posibles soluciones. Entre las funciones principales de la Gestión de Problemas están: Identificar, registrar y clasificar los problemas, dar soporte a la Gestión de Incidentes proporcionando información y soluciones temporales o parches, analizar y determinar las causas de los problemas y proponer soluciones. Se define un problema como la causa subyacente, aún no identificada, de una serie de incidentes o un incidente aislado de importancia significativa [18].

1.2.1.3 Gestión de la Configuración

La Gestión de la Configuración hace parte integral de todos los demás procesos de gestión de servicios. Se encarga de todos los procesos, herramientas y técnicas necesarias para controlar los activos y elementos de configuración que forman parte de la infraestructura IT. También proporciona información confiable y actualizada no solamente de los elementos específicos de la infraestructura (Elementos de Configuración o CI (CI: Configured Items)) necesarios para ejecutar los procesos de negocio, sino también sobre las relaciones entre ellos, asegurando la integración con las demás disciplinas de la Gestión del Servicio. Permite el desarrollo de los servicios informáticos para mejorar la calidad de una manera económicamente viable, suministra información importante para el cálculo de los costos y la facturación de los servicios ejecutados [19]. Algunos ítems configurables son el hardware, el software y la documentación asociada, entre otros.

1.2.1.4 Gestión del Nivel del Servicio

La función de la Gestión del Nivel de Servicio (SLM: Service Level Management) es negociar SLA (SLA: Service Level Agreement) con los clientes y diseñar servicios de acuerdo con los objetivos propuestos. También es responsable de asegurar que todos los Acuerdos de Nivel Operacional (OLA: Operational Level Agreements) y los Contratos de Apoyo (UC: Underpinning Contracts) sean apropiados, y de monitorear e informar acerca de los niveles de servicio [20].

1.3 Fundamentos de la Gestión de Incidentes en ITIL-LITE

1.3.1 Diferencia entre Incidente y Evento

Hay eventos que son reportados por el usuario al Service Desk en la mayoría de los casos esto no significa que sean incidentes. Hay eventos que no se relacionan con interrupciones del servicio, en cambio son indicadores del normal funcionamiento o simplemente información.

Incidentes y solicitudes de servicio son reportadas al Service Desk, esto no quiere decir que sean lo mismo. Las solicitudes de servicio no representan una interrupción del

servicio, pero son una forma de conocer las necesidades del cliente y poder hacer frente a un objetivo acordado.

1.3.2 Principalesincidentes

Se puede utilizar un procedimiento aparte, con plazos más cortos y con mayor urgencia, para tratar los principales incidentes. Se recomienda definir qué constituye que un incidente sea importante e idealmente sería asignado al sistema general de priorización de incidentes, de tal manera que se traten a través de los procesos de incidentes graves o principales.

1.3.3 Actividades de procesos, métodos y técnicas

El proceso a seguir durante la Gestión de Incidentes se muestra en la siguiente figura 2.

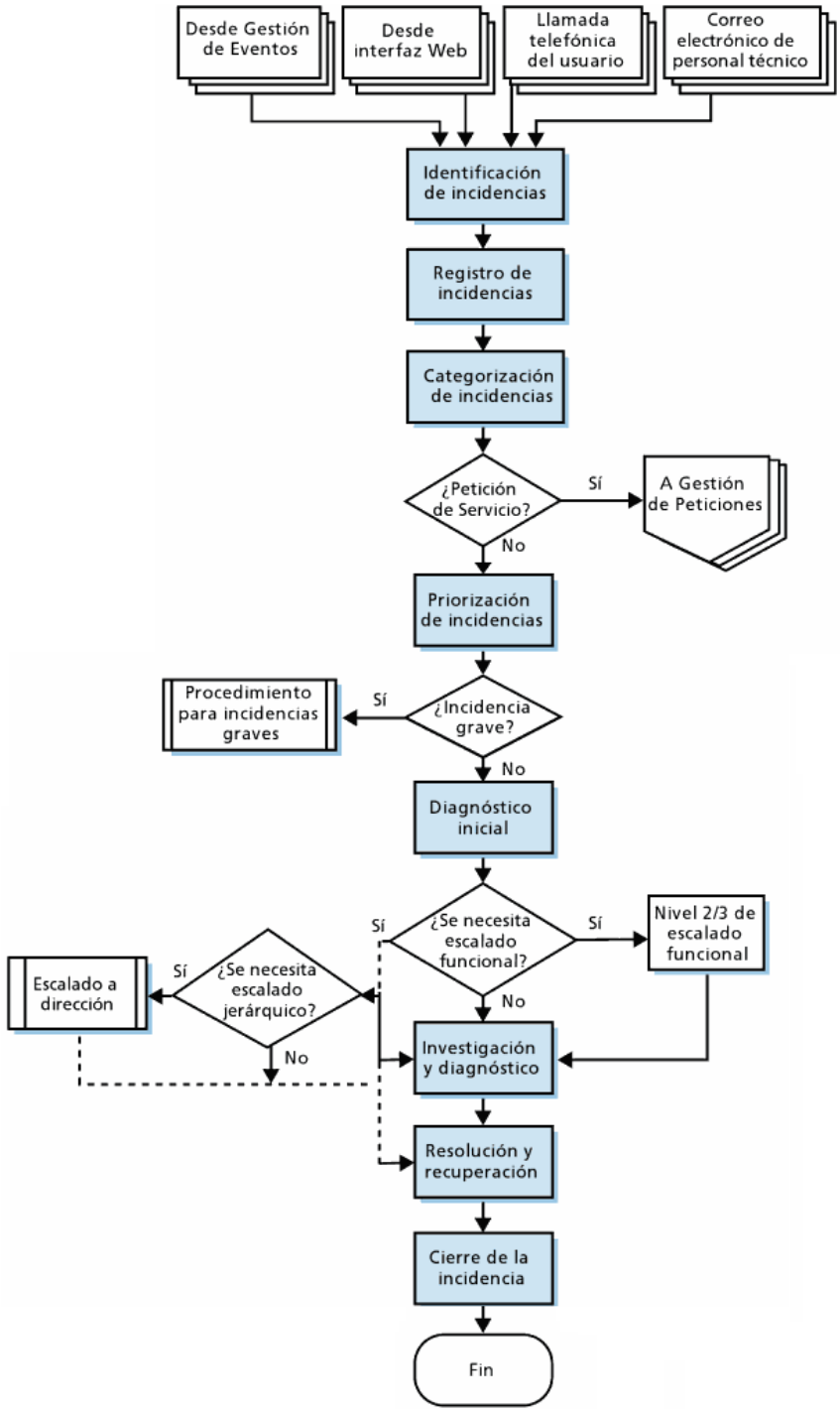


Figura 2. Flujo de Procesos en la Gestión de Incidentes ITIL v3.

1.3.3.1 Identificación de Incidentes

Se hace frente a un incidente cuando se sabe que ha ocurrido uno. Por lo general, es inaceptable, desde una perspectiva empresarial, esperar hasta que un usuario sea afectado para que entre en contacto con el Service Desk. Siempre que sea posible, todos los componentes claves deben ser monitoreados de manera que las fallas o posibles fallas se detecten a tiempo para que el proceso de Gestión de Incidentes se

pueda iniciar rápidamente. Idealmente, los incidentes deben ser resueltos antes de que tengan un impacto en los usuarios.

Para llevar a cabo la gestión en un Centrex IP es necesario conocer qué incidentes se van afrontar, para esto se hace uso de herramientas de monitoreo, con el fin de identificar incidentes y que el Service Desk pueda iniciar el proceso. Este proceso de identificación es utilizado en todas las organizaciones ya que sin este paso no existiría la Gestión de Incidentes ni otro tipo de gestión como la Gestión de Problemas.

1.3.3.2 Registro de Incidentes

Todos los incidentes deben ser registrados con su fecha y hora, independientemente del método por el cual sean reportados. La información del incidente que se considere relevante debe ser registrada y almacenada completamente en un registro histórico, de modo que si el incidente tiene que ser referido a otro(s) grupo(s) de soporte, ellos contarán con la información necesaria para asistirlos. La información de cada incidente puede incluir: número único de referencia, categorización de incidentes (a menudo se divide entre dos a cuatro niveles), urgencia del incidente, impacto del incidente, priorización del incidente, nombre/ID de la persona y/o grupo que almacenó el incidente, método de notificación; telefónico, automático, Service Desk, en persona, etc.

Este proceso es importante para una óptima gestión, ya que permite ordenar la información del incidente, para evitar posteriores sobrecostos, por ejemplo puede darse el caso de que un incidente no sea registrado, y que más de un usuario notifique la misma incidencia, por lo tanto han de evitarse duplicaciones, que se traducen al final en inversión de tiempo y costos innecesarios.

1.3.3.3 Categorización o Clasificación de Incidencias

Comienza con el registro inicial, debería ser asignada adecuadamente. Esto será importante más adelante cuando se busque el tipo y/o frecuencia del incidente, para establecer tendencias de uso en la Gestión de Problemas, Gestión de proveedores y otras actividades de ITSM. La Categorización multinivel es por lo general a tres o cuatro niveles de granularidad. Por ejemplo, un incidente puede ser categorizado como se muestra en la Figura 3.

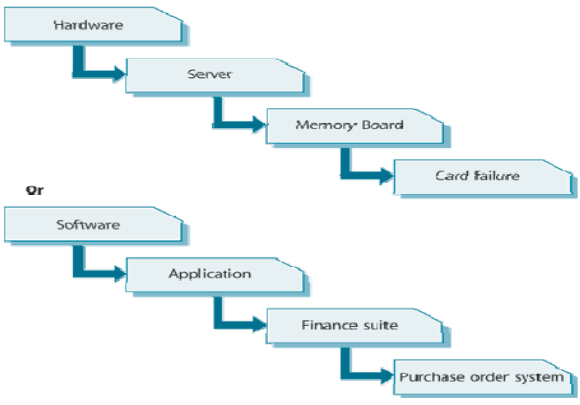


Figura3. Categorización de Incidentes Multinivel.

Toda organización es única, por lo que es difícil dar orientaciones genéricas sobre las categorías que una organización debe emplear, en particular en los niveles inferiores de los diagramas anteriores. Sin embargo, es una técnica que puede utilizarse para conseguir un conjunto correcto y completo de las categorías. Cuando se parte de cero los pasos pueden incluir:

1. Convocar una sesión para intercambiar ideas entre los grupos de apoyo pertinentes, con la participación del supervisor del Service Desk, el administrador de incidentes y los administradores de problemas.
2. Utilizar las sesiones para decidir las categorías de mejor estimación de nivel superior, e incluir otra categoría, si es necesario.
3. Utilizar las categorías durante un período de prueba corto, el tiempo suficiente para que ocurra incidentes correspondientes a cada categoría, pero no tomar demasiado tiempo para llevar a cabo un análisis.
4. Realizar un análisis de los incidentes registrados durante el período de prueba. El número de incidentes registrados en cada categoría de nivel superior confirmará si vale la pena tener esas categorías, y un análisis detallado de la "otra" categoría debe permitir la identificación de cualquier categoría adicional que se necesite.
5. Un análisis detallado de los incidentes dentro de cada categoría de nivel superior se debe utilizar para decidir las categorías de menor nivel que se requieran.
6. Revisar y repetir estas actividades después de un período de tiempo, por ejemplo, uno a tres meses, y de nuevo con regularidad para asegurarse de que sean aplicables. Se debe tener en cuenta que cualquier cambio en la clasificación puede causar algunas dificultades en el manejo de tendencias de incidentes o de informes de gestión, por lo que deben permanecer estables a menos que los cambios sean realmente necesarios.

La Categorización o clasificación de Incidentes es indispensable tenerla en cuenta en este proyecto ya que se puede agrupar tipos similares de Incidentes, lo que permite asignar el personal adecuado con la capacitación necesaria para atenderle. La categorización o clasificación también se usa para agrupar distintas clases de Elementos de Configuración. La Clasificación se utiliza con el objeto de asegurar la calidad de la información y una gestión consistente.

1.3.3.4 Priorización de Incidentes

Otro aspecto importante al registrar cada incidente es ponerse de acuerdo y asignar un código de prioridad adecuado, ya que esto determinará si el incidente es manejado tanto por las herramientas de apoyo y/o personal de apoyo.

La Priorización se puede determinar teniendo en cuenta la urgencia del incidente y el nivel de impacto que éste causa en una empresa. Un indicador del impacto es a menudo el número de usuarios que se vean afectados. Pero en algunos casos, la pérdida del servicio a un único usuario puede tener un impacto importante en el negocio, de esta manera los números por si solos no son suficientes para evaluar la prioridad general. Otros factores que también pueden decidir los niveles de impacto son: el riesgo a la vida o la integridad física, el número de servicios afectados, el nivel de pérdidas financieras, efecto en la reputación empresarial, reglamentación o incumplimientos legislativos.

Una forma eficaz para el cálculo de estos elementos y la deducción de un nivel completo de prioridad para cada incidente se da en la Tabla 1:

Tabla 1. Sistema Codificado de Prioridad Simple.

			Impact	
		High	Medium	Low
	High	1	2	3
Urgency	Medium	2	3	4
	Low	3	4	5

Priority code	Description	Target resolution time
1	Critical	1 hour
2	High	8 hours
3	Medium	24 hours
4	Low	48 hours
5	Planning	Planned

Hay que señalar que habrá ocasiones en las que, debido a la conveniencia en particular de un negocio, los niveles normales de prioridad tiene que ser anulados. Cuando un usuario está convencido de que el nivel de prioridad de un incidente debe exceder las recomendaciones normales, el Service Desk podría cumplir con dicha solicitud, y si posteriormente resulta ser incorrecta esto puede ser resuelto como un caso de nivel de gestión fuera de línea, para evitar que ocurra una controversia cuando el usuario está en el teléfono.

Cabe señalar que la prioridad de un incidente puede ser dinámica, si las circunstancias cambian, o si un incidente no se resuelve dentro de los límites del tiempo del SLA, a continuación, la prioridad debe ser modificada para reflejar la nueva situación.

En cualquier organización sin importar su tamaño es necesario llevar a cabo este paso, pues será empleado para identificar la importancia relativa de un incidente, ya que cada incidente afecta a la organización en forma diferente. Además es utilizado para identificar los plazos requeridos para la realización de las diferentes acciones.

1.3.3.5 El diagnóstico inicial

El analista del Service Desk debe llevar a cabo el diagnóstico inicial, por lo general mientras el usuario todavía está en el teléfono, si la llamada se produce de esta manera, para tratar de descubrir los síntomas del incidente y para determinar exactamente qué falló y cómo corregirlo. Es en esta etapa, en donde el papel del diagnóstico y la información conocida del error, puede ser más valioso para permitir un diagnóstico temprano y preciso.

Si es posible, el analista del Service Desk resolverá el incidente mientras el usuario todavía está en el teléfono, y cerrará el incidente si la solución es correcta. Si el analista del Service Desk no puede resolver el incidente mientras el usuario aún está en el teléfono, pero hay una posibilidad de que el Service Desk puede ser capaz de hacerlo en el plazo acordado sin la ayuda de otros grupos de apoyo, el analista debe informar sus intenciones al usuario, dar al usuario el número de referencia del incidente y tratar de encontrar una solución.

El propósito del Diagnóstico en este proyecto es identificar alternativas que permitan una pronta solución al incidente de forma definitiva o temporal. Este paso permite recoger y analizar datos para evaluar incidentes de diversa naturaleza. En caso de no encontrarse una solución después de realizado este proceso se asignará a otro nivel de gestión.

1.3.3.6 Escala de Incidentes

Escala funcional. Cuando el Service Desk no es capaz de resolver el incidente en sí mismo o cuando los límites de tiempo para la solución del primer punto se han superado, el incidente debería ser inmediatamente escalado para darle más apoyo.

Si se cuenta con un grupo de apoyo de segundo nivel y el Service Desk cree que el incidente puede ser resuelto por dicho grupo, este debe ser remitido a ellos. Si es evidente que el incidente necesita conocimientos técnicos más profundos, o cuando el grupo de segundo nivel no ha podido resolver el incidente dentro de los tiempos limites pactados, el incidente debe ser inmediatamente escalado al correspondiente grupo de soporte de tercer nivel. Se debe tener en cuenta que los grupos de soporte de tercer nivel pueden ser internos, también puede ser un tercer grupo semejante que provee software, fabrica hardware o hace mantenimiento. Las reglas para la escalada y el manejo de incidentes deben ser acordados previamente con grupos de soporte internos y externos.

Escala jerárquica. Si los incidentes son de carácter grave (por ejemplo, los incidentes de prioridad uno) los administradores de IT apropiados deben ser notificados, por lo menos para efectos de información. La escala jerárquica también se utiliza si los pasos de investigación, diagnóstico, solución y recuperación son demasiado largos o demasiado difíciles de alcanzar. La escala jerárquica debe continuar la cadena de gestión para que los altos directivos sean conscientes y puedan estar preparados para tomar las medidas necesarias, tales como la asignación de recursos adicionales o en relación con los proveedores/personal de mantenimiento. La escala jerárquica también se utiliza cuando hay entorpecimiento en la asignación de quién debe manejar el incidente.

Los niveles exactos y los plazos para la escala funcional y jerárquica deben ser acordados, teniendo en cuenta los objetivos del SLA, e integrarse en las herramientas de apoyo las cuales pueden ser utilizadas para la vigilancia y el control del flujo del proceso dentro de los plazos acordados.

El Service Desk debe mantener informado al usuario de cualquier escalada relevante que se lleve a cabo y garantizar que el registro de incidentes se actualice en consecuencia para mantener un historial completo de las acciones.

Nota relativa a la asignación del incidente: puede haber muchos incidentes en una cola con el mismo nivel de prioridad, por lo que será el trabajo del Service Desk y/o el personal de Gestión de Incidentes, junto con los directores de los distintos grupos de soporte a los que los incidentes son escalados, para decidir el orden en el que los incidentes deben ser recogidos y trabajados activamente. Estos administradores deben

asegurarse de que los incidentes se tratan en el verdadero orden de prioridad de negocio.

Esta actividad es importante para el proyecto, ya que aquí se puede obtener recursos adicionales para resolver un incidente, estos recursos pueden ser asignarle a dicho incidente personal con mayor capacitación, cuando sean necesarios para alcanzar las metas de Nivel de Servicio o las expectativas del Cliente. Se escala la incidencia porque se necesita más conocimiento, también si se necesita más soporte, más dinero, más poder de decisión para resolver la incidencia y que más personas trabajen en ésta. Se puede informar a los directivos de qué ocurre y mantenerles al día, o pedir su soporte en el progreso del proceso del incidente.

1.3.3.7 Investigación y Diagnóstico

Si el usuario está en búsqueda de información de los incidentes, el Service Desk debe ser capaz de proporcionar ésta con rapidez, un incidente probablemente requiera cierto grado de investigación y diagnóstico. Cada grupo de soporte involucrado con el manejo y diagnóstico de incidentes, las actividades y medidas adoptadas para tratar de resolver el incidente, deben estar plenamente documentadas para que se mantenga un registro histórico completo de las actividades.

El tiempo es valioso y se puede perder si la investigación y la acción de diagnóstico, incluyendo acciones de solución y de recuperación, se realizan en serie. Siempre que sea posible, estas actividades deben llevarse a cabo en paralelo para reducir los plazos, y las herramientas de apoyo deben ser diseñadas y/o seleccionadas para permitir esto. También se debe tener cuidado en la coordinación de las actividades, en particular las actividades de solución o de recuperación, de lo contrario las acciones de los distintos grupos pueden entrar en conflicto y complicar aún más la solución. La investigación es probable que incluya acciones como: establecer exactamente que ha salido mal o que es lo que busca el usuario, comprender el orden cronológico de los eventos, el impacto del incidente incluyendo el número y el alcance de los usuarios afectados, identificar los eventos que podrían haber provocado el incidente (por ejemplo, un cambio reciente, una acción del usuario), entre otros.

Este paso es importante llevarlo a cabo en grandes organizaciones ya que éstas sí cuentan con el personal y los recursos financieros para mantener los distintos grupos de soporte, que dediquen su tiempo y esfuerzo exclusivamente a estas actividades anteriormente mencionadas. En el caso de una microempresa es diferente, debido a que estas disponen de personal limitado, que tendrían asignadas tareas diferentes a las mencionadas y que les demandaría todo o gran parte de su tiempo, por ejemplo un técnico de hardware no tendría tiempo ni conocimiento para identificar los eventos que podrían haber provocado un incidente. Además, en el caso de Colombia una microempresa es aquella que no tiene más de diez empleados y posee activos totales por valor inferior a quinientos uno (501) salarios mínimos mensuales legales vigentes. Lo que dificultaría aún más alcanzar este paso con un alto grado de profundidad.

1.3.3.8 Solución y Recuperación

Si una potencial solución al incidente ha sido identificada, ésta debería ser aplicada y probada. Las acciones específicas que deben realizarse y las personas que participarán

en la toma de acciones de recuperación del Nivel del Servicio pueden variar, dependiendo de la naturaleza de la falla, estas acciones podrían incluir: pedir al usuario que lleve a cabo actividades dirigidas sobre su propio escritorio o equipo remoto.

- El Service Desk puede implementar la solución de forma centralizada, por ejemplo reiniciando un servidor, o de forma remota, mediante software para controlar el escritorio del usuario, con el fin de diagnosticar y aplicar una solución.
- A los especialistas de los grupos de soporte se les pide llevar a cabo acciones específicas de recuperación, por ejemplo, al soporte de red se le puede pedir que realice la reconfiguración de un router.
- Un proveedor o personal de mantenimiento que se le pida que resuelva la falla.

Cuando una solución se ha identificado, se debe hacer pruebas para garantizar que las medidas de recuperación estén completas y que el servicio al cliente ha sido totalmente restaurado.

Las acciones necesarias de recuperación en algunos casos pueden ser tomadas por separado por dos o más grupos, pero de forma coordinada a la hora de ser implementadas. Para estos casos se deben coordinar las actividades y establecer enlaces con todas las partes implicadas.

Independientemente de las medidas adoptadas, o quien las realice, el registro del incidente debe ser actualizado de acuerdo con toda la información pertinente y detallado para mantener un historial completo. El grupo de solución debe pasar el incidente de nuevo al Service Desk para la acción de cierre.

Este paso contribuye a alcanzar el objetivo propuesto, porque luego de identificar el incidente y resolverlo, se agrega dicha solución al registro histórico o base de conocimiento (soluciones), ayuda a responder con mayor rapidez si otro usuario viene con el mismo incidente. Este último usuario obtendría una solución inmediata lo que ayudaría a lograr mejor satisfacción del cliente. En caso de ser necesario se puede emitir una petición de cambio, si la incidencia fuera recurrente y no se encuentra una solución definitiva, se debería informar a la Gestión de Problemas para que estudie las causas subyacentes.

1.3.3.9 Cierre de Incidentes

El Service Desk debe comprobar que el incidente se ha resuelto y que los usuarios están satisfechos y dispuestos a aceptar que el incidente se cierre. El Service Desk también debe comprobar lo siguiente:

Categorización de Cierre. Comprobar y confirmar que la categorización inicial del incidente fue correcta, si la clasificación resultó incorrecta, se debe actualizar el registro de modo que una correcta categorización de cierre se registre para el incidente, buscando asesoramiento o la orientación del(los) grupo(s) de solución.
Encuesta de satisfacción del usuario. Realizar una llamada o hacer una encuesta por e-mail para saber la satisfacción de los usuarios en cuanto a la calidad del Service Desk.

Documentación de Incidentes. Búsqueda de detalles pendientes para garantizar que el registro de Incidentes este plenamente documentado, para que un registro histórico este completo a un nivel suficiente de detalle.
Progreso o recurrencia del problema. Determinar si es probable que el incidente pudiera repetirse y decidir si las medidas preventivas son necesarias para evitar esto. En relación con la Gestión de Problemas, construir un registro de problemas en todos esos casos para que se inicie la acción preventiva.
Cierre oficial. Formalmente cerrar el registro de incidentes. Algunas organizaciones pueden decidir utilizar un tiempo de cierre específico y automático en el cierre de un incidente, o incluso todos, por ejemplo el incidente se cerrará automáticamente después de dos días hábiles si no hay más contactos hechos por el usuario. Cuando éste enfoque se considera, en primer lugar, debe ser plenamente discutido y acordado con los usuarios, y realizar una amplia publicidad a fin de que todos los usuarios y el personal de IT sean conscientes de ello. Puede ser inapropiado usar este método para determinados tipos de incidentes, como los incidentes de prioridad uno o los relacionados con los VIP, etc.

Este paso es fundamental pues garantiza que el cliente quede satisfecho con la resolución del incidente antes de realizar el cierre. A veces, cuando el usuario no confirma que el incidente fue resuelto, un criterio de cierre automático ayudará a cerrar incidentes en estas ocasiones. Los incidentes serán cerrados luego de un número específico de días cuando no haya respuesta del usuario. Las reglas anteriormente mencionadas ayudan a asegurarse que todos los incidentes tengan una solución antes de poder cerrarse.

1.3.4 Reglas para la reapertura de los incidentes.

A pesar de todos los cuidados adecuados, habrá ocasiones en que los incidentes se repiten después de que han sido oficialmente cerrados. Debido a estos casos, es aconsejable disponer de reglas predefinidas sobre si se reabre y cuando se puede volver a abrir un incidente. Podría tener sentido, por ejemplo, si el incidente se repite dentro de un día de trabajo, entonces se puede volver a abrir, pero más allá de este punto, un nuevo incidente debe ser abierto pero vinculado con el incidente anterior. Las reglas para el tiempo límite pueden variar entre las distintas organizaciones, pero estas deben ser claras, acordadas, documentadas y entregadas a todo el personal del Service Desk, para que la uniformidad se aplique.

Tener acordadas reglas para reapertura de incidentes permite que no se abran nuevos incidentes y por esto no se gaste tiempo y esfuerzos, ayuda a que no exista duplicidad del incidente. De presentarse estos casos lo que se debería hacer es buscar donde se encuentra el incidente, mirar hasta ese momento que tratamiento ha recibido para tomar nuevas acciones o remitirlos a otros procesos de gestión.

1.3.5 Factores críticos de éxito

Los siguientes factores serán críticos para el éxito de la Gestión de Incidentes:

- Un buen Service Desk es la clave para el éxito de la Gestión de Incidentes.
- Definición clara de objetivos para trabajar.

- Capacitación y orientación adecuada al cliente y entrenamiento técnico al personal de soporte con los niveles de competencia correcta, en todas las etapas del proceso.
- Herramientas de Soporte integradas para conducir y controlar los procesos.

1.3.6 Riesgos

Los riesgos para el éxito de la Gestión de Incidentes en realidad son similares a algunos de los desafíos antes mencionados en los factores críticos de éxito. Ellos incluyen:

• El ser inundado con incidentes que no pueden ser manejados dentro de los plazos aceptables debido a la falta de entrenamiento o capacitación en los recursos disponibles.
• Los incidentes no muestran señal de mejora debido a las herramientas de apoyo insuficientes para generar alertas e impulsar el progreso del sistema.
• La falta de fuentes de información adecuada y/o puntual, debido a herramientas inadecuadas o falta de integración.

Después de hacer un estudio del proceso de Gestión de Incidentes del marco de referencia ITIL, se mencionan algunas de las tecnologías para gestionar servicios en el campo de las telecomunicaciones, con el propósito de deducir cuál de estas tecnologías es la más adecuada para el desarrollo de este proyecto.

1.4 Red de Gestión de Telecomunicaciones (TMN: Telecommunication Management Network)

TMN se encuentra definida en la recomendación M.3000 de la ITU-T, utiliza el modelo de las FCAPS (FCAPS: Fault, Configuration, Accounting, Performance, Security)Fallos, Configuración, Contabilidad, Rendimiento y Seguridad de la ISO (ISO: International Organization for Standardization). Está definida como el aprovisionamiento de una arquitectura organizada para realizar la interconexión entre varios tipos de sistemas de operaciones y/o equipos de telecomunicaciones para el intercambio de información de gestión usando interfaces normalizadas [21].

TMN consta de tres partes fundamentales, arquitectura funcional, física y de información. La arquitectura funcional de TMN identifica una serie de funciones y sus relaciones de intercambio de información, conocidos como puntos de referencia. La arquitectura física tiene que ver con las interfaces físicas y protocolos. La arquitectura de información tiene que ver con los paradigmas de orientación a objetos y de gestor/agente.

1.4.1 Objetivo básico

El objetivo de TMN es proporcionar un marco para la gestión de las telecomunicaciones, introduciendo el concepto de los modelos genéricos de la red para la gestión. Es posible realizar la gestión general del equipo, de la red y de diversos servicios usando modelos genéricos de la información.

1.4.2 Arquitectura funcional

La arquitectura funcional (Figura 4) divide un dominio TMN en diferentes bloques funcionales, los cuales agrupan funciones básicas [22]. Combinando bloques funcionales de diferentes maneras se puedellegara funcionalidades TMN de distintas complejidades.

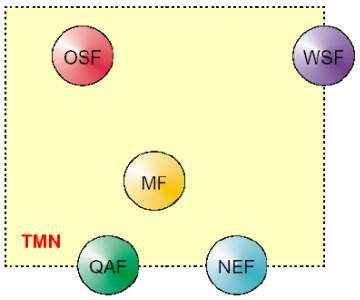


Figura 4. Bloques Funcionales de TMN

A continuación se describen los diferentes tipos de bloques de funcionales:

Funciones de los Sistemas de Operación (OSF: Operation System Function)
Proporcionan las funciones de procesamiento de la información de gestión y funciones de planificación para la red de telecomunicaciones a diseñar. A su vez existen cuatro tipos de bloques funcionales OSF: Element OSF, Network OSF, Service OSF, Business OSF.
Funciones de elementos de Red (NEF: Network Element Function)
Reúne las funciones que se relacionan con un elemento de red. Los componentes de una red de telecomunicaciones son representados por uno o más bloques NEF.
Funciones de Estación de Trabajo (WSF: Workstation Function)
Habilita la información de gestión para que pueda ser vista por los usuarios. Esto implica la traducción de los datos de gestión que llegan del bloque OSF.
Funciones de Mediación (MF: Mediation Function)
Usadas para el intercambio de información de gestión cuando los bloques funcionales tienen distintos puntos de referencia. Estas funciones pueden archivar, convertir, rutear, mapear direcciones, filtrar y condensar información de gestión.
Funciones de Adaptador Q (QAF: Q Adaptor Function)
Agrupa funciones para traducir información de gestión entre redes TMN y otras que no lo son. Estas funciones pueden tener una parte fuera de la frontera TMN. Una vez determinados los bloques funcionales, queda constituida la Red de Gestión de Telecomunicaciones.

1.4.3 Arquitectura lógica

Cuando un proyecto se vuelve complejo, es normal dividir la complejidad, para lograr un tratamiento más sencillo. En materia de sistemas y telecomunicaciones, un enfoque utilizado es la división en capas. Donde se considera la relación existente entre Gestores y Agentes, esto puede generar un modelo en capas como sigue [22].

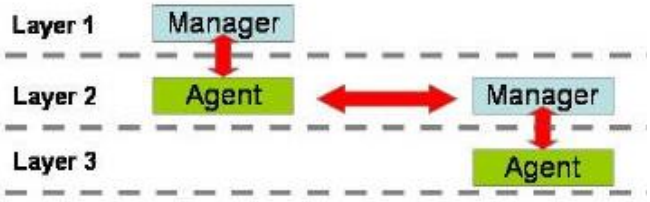


Figura 5. Modelo de Capas

Este modelo sigue una estructura jerárquica que proporciona un panorama lógico de combinaciones entre componentes de gestión [21]. Ésta estructura fue introducida por British Telecomm en su arquitectura CNA (CNA: Cooperative Network Architecture), según esta arquitectura, los sistemas de operación se dividen en cuatro grandes grupos de gestión, en donde las capas superiores utilizan los servicios proporcionados por las capas inferiores, formando una estructura piramidal como se observa en la siguiente figura.

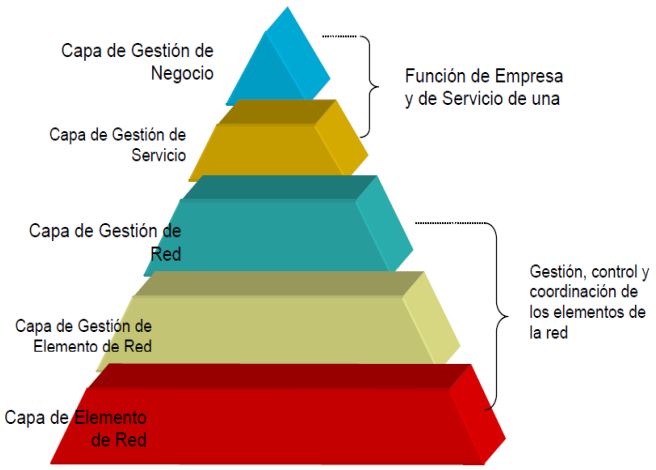


Figura 6. Arquitectura Lógica por Capas de TMN

Las distintas capas TMN, llamadas Capas de Gestión Funcional, las cuales se pueden observar en el gráfico anterior, son las siguientes:

Nivel de Gestión del Negocio (BML: Business Management Layer). Esta capa involucra funciones de dominio privado, es decir que no manejan puntos de referencia con otras redes. Se incluyen funcionalidades tales como acuerdos de planificación de red entre operadores de telecomunicaciones y actividades de nivel ejecutivo como la planificación estratégica, presupuestos y finanzas. Sus funciones tienen más relación con la fijación y seguimiento de las metas empresariales que con la implementación de los detalles.

Nivel de Gestión del servicio (SML: Service Management Layer). Provee funcionalidades de contacto con el cliente. Gestiona los aspectos contractuales y las negociaciones de los servicios suministrados al cliente, gracias a la información recibida del Nivel de Gestión de Red y el Nivel de Gestión del Negocio. También tiene funciones de solución de reclamos de clientes, reporte de fallas y el mantenimiento de los datos de calidad de servicio QoS (QoS: Quality of Service).

Nivel de Gestión de Red (NML: Network Management Layer). Soporta las demandas de red hechas por el Nivel de Gestión de Servicios. Brinda a la capa superior SML una visión de la red end-to-end, independientemente de la tecnología involucrada y basada en la información presentada por el Nivel de Gestión de Elementos. Se comunica con las otras capas utilizando interfaces estandarizadas. Facilita la disponibilidad y accesibilidad de recursos de la red, cómo están interrelacionados y asignados geográficamente.

Nivel de Gestión de Elemento (EML: Element Management Layer). Se encarga de funciones de gestión, control y coordinación de elementos de red y de subredes. En esta capa los datos de control de los elementos de red son analizados e interpretados para darles un significado en el monitoreo de la subred. Como una subred es un subconjunto de la red, estos datos son pasados a la capa NML para obtener así la información de la red completa.

Nivel de Elemento de Red (NEL: Element Layer). Integran esta capa las entidades físicas que necesitan ser gestionadas. Sus funciones son: recolección de datos de la red como tráfico, alarmas, llevar a cabo autodiagnósticos, monitoreo de alarmas, también realiza traducción de direcciones, conversión de protocolos, conversión y análisis de datos.

1.4.4 Arquitectura de Información

La gestión de red necesita el intercambio de información entre procesos de gestión, y para esto TMN utiliza un modelo de información que se basa, en su mayor parte, sobre el modelo de gestión de red OSI (OSI: Open System Interconnection)/CMIP [23]. Los conceptos básicos usados en la definición de la arquitectura de información de TMN son similares a aquellos aplicados en SNMP y OSI/CMIP, estos conceptos son: Objeto Gestionado (MO: ManagedObject), Agente (Agent), Gestor (Manager), Base de Información de Gestión (MIB: Management Information Base).

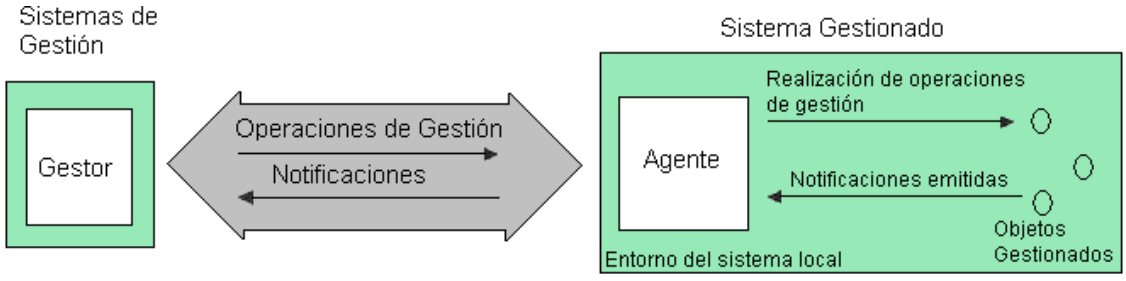


Figura 7. La Interacción entre un Gestor, un Agente, y los Objetos Gestionados

La interacción entre un agente, un gestor, y los objetos gestionados de acuerdo a ITU-T M.3010 se muestra en la figura 7. El gestor y el agente se comunican utilizando protocolos "Q" estándares construidos de acuerdo al modelo de comunicación de siete capas de OSI. Los componentes de un protocolo Q son: Interfaz de aplicación, protocolo de aplicación (la séptima capa del modelo OSI), protocolos de soporte (capas cuatro y seis del modelo OSI), protocolos de red (capas uno y tres del modelo OSI).

Posee un papel clave al permitir adecuar la organización de la gestión de red a las particularidades de los operadores de redes y servicios. La modularidad y flexibilidad de su tecnología es muy adecuada para las redes heterogéneas que existen en la actualidad. Usa como arquitectura de redes el modelo OSI de la ISO.

Figura 8. Modelo OSI

1.4.5 Arquitectura Física

Agrupa todos los elementos materiales, hardware, software y las interfaces entre ellos, que son las que llevan a cabo las funciones de procesamiento y de comunicación [22]. Aquí se identifican Sistemas de Operación (OS: Operation Systems), Dispositivos de Mediación (MD: Mediation Device), Dispositivos de Adaptación Q (QA: Q Adaptor), Estación de Trabajo (WS: Workstation), Elementos de Red (NE: Network Elements).

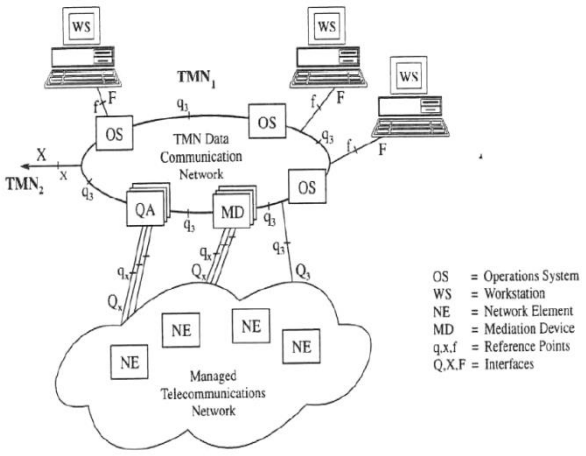


Figura 9. Representación de la Arquitectura Física TMN

Como se observa en la gráfica (figura 9), entre las distintas interfaces tendremos:

Interfaz Q. La interface q existe entre 2 bloques funcionales que se encuentran en el mismo dominio TMN. Existen 2 tipos de interfaz Q como es Qx que transporta información que se comparte entre los dispositivos de mediación y los elementos de red. Existe también entre QA y MD, y entre MD y MD. Este tipo de interfaz se utiliza cuando no es necesario transmitir la totalidad de información que manejan las 7 capas del modelo OSI. Q3 es la interfaz utilizada entre un NE y OS, entre QA y OS, entre MD y OS, y entre OS y OS. Este tipo de interfaz requiere el pasaje de toda la información correspondiente a las 7 capas del modelo OSI.
Interfaz F. Esta interfaz es la que comunica una estación de trabajo con un Sistema de Operación y con un dispositivo de mediación.
Interfaz X. Utilizada para conectar Sistemas de Operación de distintas redes TMN, o con otras redes que utilizan interfaces del tipo TMN.
Interfaz M. Ubicada fuera de la frontera TMN. Provee conexión entre QAF y entidades gestionadas no TMN.
Interfaz G. No es considerada parte de una TMN. Provee acceso a la interfaz de usuario en una estación de trabajo.

1.5 WBEM (WBEM: Web Based Enterprise Management)

Nace de la tendencia de integrar los tipos de sistemas de información en los entornos de WWW e intranet, esto se conoce como Gestión Basada en Web (WBM: Web Based Management), ya que la industria cambia la manera en que los empleados trabajan, utilizando aplicaciones cliente-servidor, las cuales permiten el acceso a recursos distribuidos, al crecimiento del Internet y la creciente demanda de aplicaciones multimedia, por esto se requiere el desarrollo y uso de herramientas que permitan analizar y medir eficazmente el tráfico en línea, para que se pueda obtener una información en tiempo real que permita solventar y plantear nuevas estrategias a los problemas de congestionamiento de la red [24]. Como solución a lo anterior surgió WBEM, basando la estructura de la información en el Modelo Común de Información (CIM: Common Information Model), representando la información de manera jerárquica, altamente estructurada, organizada y conformada por objetos capaces de relacionarse entre sí. Sus estándares de codificación son XML (XML: eXtensible Mark-up Language)-CIM y utilizando para la transferencia el protocolo HTTP (HTTP: Hyper Text Transfer Protocol) (figura 10).

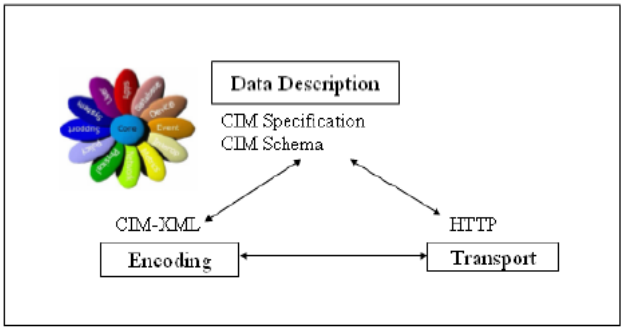


Figura 10. Las Relaciones entre las Tecnologías Estándar WBEM

La iniciativa de WBEM se presentó como un único mecanismo o estándar que permitiera analizar, describir y compartir la información de gestión de una red y varios recursos, debido a la enorme necesidad que existe de analizar el tráfico en Internet, el desempeño de los recursos y contar con una herramienta que permitiera el intercambio de datos sin pensar en la compatibilidad entre plataformas. Con estas características se observa que la base de WBEM está en desarrollar una arquitectura abierta que sea el punto de la integración para las organizaciones mundiales, proporcionando un fácil acceso a los datos de gestión de una variedad de fuentes. Web Based Enterprise Management es un estándar o conjunto de gestión y tecnologías, actualmente estandarizado por el Grupo de Trabajo de Gestión Distribuida (DMTF: Distributed Management TaskForce).

1.5.1 Arquitectura y Elementos

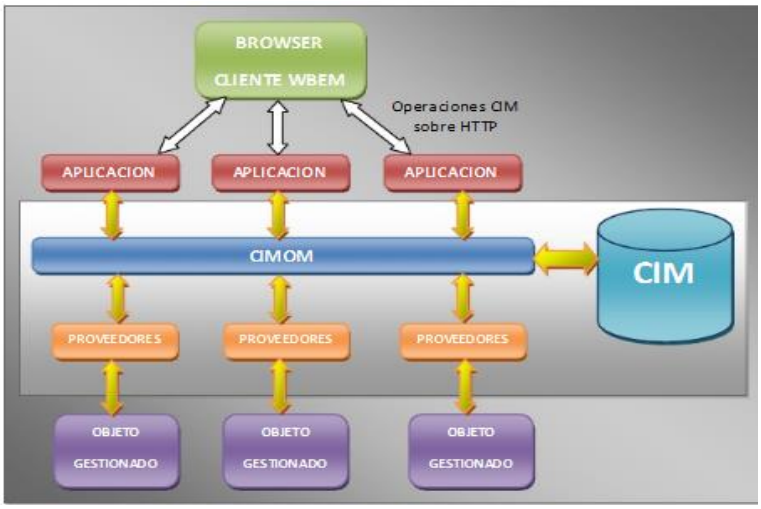


Figura 11. Modelo de la Arquitectura WBEM

Cliente WBEM. Dicho cliente será la interfaz de usuario del operador del sistema gestionado. Se comunica con el siguiente nivel empleando el HTTP y XML como método para la codificación de la información [25]. El empleo de estas tecnologías es habitual para los servicios basados en Web, permitiendo reutilizar herramientas.

Gestor de Objetos CIM. Elemento nuclear de la arquitectura. Maneja un modelo de información independiente de los dominios de gestión, llamado Modelo de Información Común. Orientado a objetos y representable en XML.

Conjunto de Proveedores. Pasarelas entre el CIMOM (CIMOM: CIM Object Management) y los agentes de los distintos dominios de gestión. Estos proveedores deberán ser capaces de interactuar con el CIMOM. Utilizando HTTP y XML. Y con cada agente, usando el protocolo respectivo para acceder a cada agente.

Conjunto de agentes. Los cuales se encuentran en los recursos gestionados e interactúan con los proveedores empleando el protocolo asociado a su domino de gestión.

1.6 CORBA (CORBA: Common Object Request Broker Architecture) Arquitectura Común de Intermediarios en Peticiones a Objetos

Es una infraestructura computacional abierta de objetos distribuidos, especificada por el Grupo de Gestión de Objetos OMG (OMG: Object Management Group) con el ánimo de describir todas las características del ORB (ORB: Object Request Broker) de OMA (OMA: Object Management Architecture) [26]. En la siguiente figurase puede apreciar los componentes CORBA.

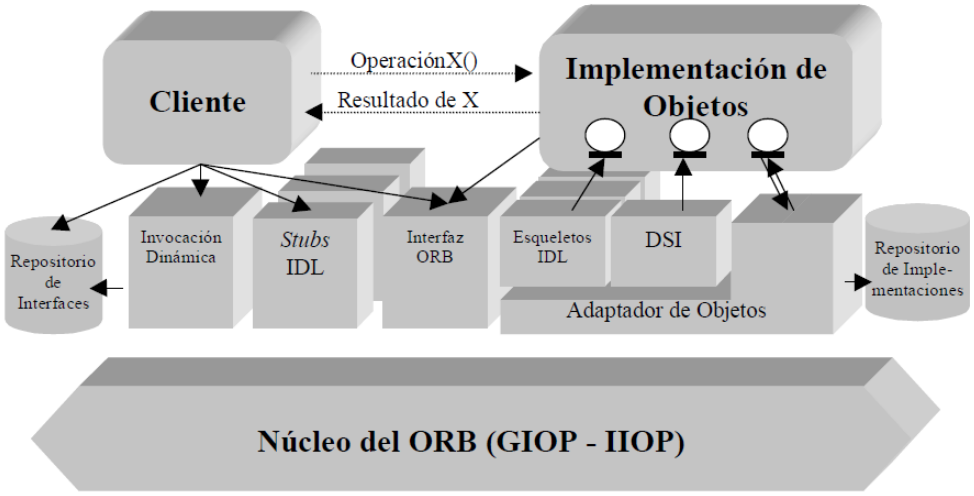


Figura 12. Componentes de CORBA

1.6.1 El Cliente.

Es la entidad que invoca operaciones sobre un objeto de la implementación de objetos y estos servicios que brinda son transparentes al cliente, así este objeto sea remoto, se comporta como si fuese local.

1.6.2 El Intermediario de Peticiones de Objetos (ORB: Object Request Broker)

El ORB es el encargado de dar transparencia en la comunicación a los clientes, en lo que se refiere al envió de peticiones y la devolución de respuestas ante una solicitud de servicios de un objeto. El objeto solicitado por un cliente y al que el ORB envía sus peticiones, es llamado Objeto Destino (Target Object).

La transparencia a la que antes se hace alusión se refleja en que, el cliente no conoce ni necesita la localización de los objetos. Tampoco conoce la implementación de los objetos con los que desea interactuar, no conoce el lenguaje de programación en que están escritos, el sistema operativo, ni el hardware sobre el cual está corriendo. El cliente no se preocupa de la activación de los objetos solicitados, ni tampoco de los mecanismos de comunicación TCP/IP, llamada de métodos locales, etc.

1.6.3 La interfaz del ORB.

Es un conjunto de Interfaces de Programación de Aplicaciones (API: Application Programming Interfaces) que definen una serie de funciones del ORB y que pueden ser accedidas directamente por el código cliente. Entre ellas están las de convertir las referencias de objetos (información necesaria que un cliente necesita para operar con el ORB y dicho Tarject Object) en cadenas de caracteres o viceversa y las que sirven para crear estructuras de datos y listas de argumentos de peticiones, hechos a través de una invocación dinámica.

1.6.4 Lenguaje de Definición de interfaces (IDL: Interface Definition Language)

Cuando un cliente solicita los servicios de un objeto, este debe conocer las operaciones soportadas por dicho objeto. Las interfaces de un objeto describen dichas operaciones. Una de las ventajas de describir las operaciones de esta forma, es separar los puntos de acceso a un objeto de su propia implementación, lo que permite que los objetos sean implementados en diferentes lenguajes de programación e interactúen entre sí en forma transparente.

La interoperabilidad de componentes CORBA se garantiza con la descripción de interfaces en lenguaje IDL. El OMG-IDL es un lenguaje de especificación parecido en estructura a C++, que permite declarar la interfaz de un objeto con el mundo exterior. Es independiente del lenguaje de implementación y plataforma de ejecución. OMG define la traducción de IDL a los lenguajes más comúnmente utilizados: C, C++, Java, ADA95, Smalltalk, etc.

1.6.5 Cabos y esqueletos IDL

Un stub o cabo también llamado Proxy es un ente encargado de enviar las peticiones de un cliente a un servidor a través de ORB. A esto se le denomina Marshaling o serialización: se convierten las peticiones de un cliente implementado en algún lenguaje de programación en una presentación adecuada para el envió de información a través del ORB.

El skeleton o esqueleto es el encargado en el servidor de colaborar con la recepción de dichas peticiones desde el ORB y enviarlas a la implementación de objetos de CORBA. A esta otra operación se la llama Unmarshaling o deserialización. También a través del esqueleto se envía alguna respuesta a través del ORB y es recibida por el cliente por medio del stub. En este caso se cambian los papeles y es el esqueleto quien serializa y el stub quien deserializa.

1.6.6 Repositorio de interfaces (IR: Interface Repository)

Es una base de datos distribuida, contiene información de las interfaces IDL definidas para los objetos que cooperan en un entorno distribuido, puede ser accedida o sobrescrita en tiempo de ejecución. Se puede pensar en el IR como un objeto CORBA, con una base de datos asociada y que tiene un conjunto de operaciones que se puede utilizar como si fuese un objeto cualquiera. El servicio que ofrece dicho objeto CORBA es permitir navegar sobre la jerarquía de interfaces almacenada en la base de datos, de tal forma que se pueda saber si se quiere, la descripción de todas las operaciones

que un objeto soporta. Una aplicación muy interesante y de mucha utilidad es usar el IR para descubrir interfaces de objetos en tiempo de ejecución, y acceder a ellos mediante invocación dinámica.

1.6.7 Adaptador de Objetos

Es el ente de contacto entre el ORB y la implementación de objetos, acepta peticiones en nombre de los objetos servidores. Se encarga en tiempo de ejecución de activar, ejemplarizar, pasar peticiones y generar referencias de dichos objetos.

Colabora con el ORB para que todas las peticiones que se hagan de múltiples conexiones, sean recibidas sin ningún tipo de bloqueo. El adaptador de objetos tiene tres interfaces asociadas, una al DSI (DSI: Dynamic Skeleton Interface), una al IDL skeleton y otra a la implementación de objetos, siendo las dos primeras privadas y la última publica, con el ánimo de aislar la implementación de objetos del ORB tanto como sea posible.

1.7 JMX (JMX: Java Management Extension)

Esta tecnología puede estructurarse en tres niveles, como sigue: Nivel de Infraestructura o instrumentación, Nivel de agentes JMX, Nivel de servicios distribuidos o Nivel de adaptación o Gestión y el Agente remoto.

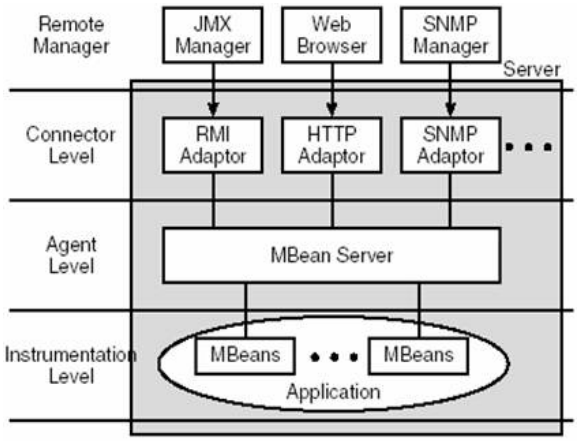


Figura 13. Relación entre los Componentes de la Arquitectura JMX

- **Nivel de infraestructura JMX**

Este nivel o capa incluye todos los recursos o componentes que facilitan la información necesaria para la gestión de aplicaciones, en este se definen los requerimientos para implementar un recurso gestionable (aplicaciones, componentes de los servicios, dispositivos, etc) mediante JMX [27].

Este nivel presenta cuatro tipos de MBean:

MBean Estándar: es un Java Bean simple y definido estáticamente (son los más comunes dentro de JBOSS).

MBean Dinámicos: estos exponen su interfaz en tiempo de ejecución.
MBean Abiertos: estos son una extensión de los anteriores.
MBean de Modelo: estos también son una extensión de los MBean Dinámicos, simplifican la instrumentación de los recursos dándole a estos un comportamiento por defecto. Los XMBean de JBOSS son un ejemplo de este tipo de MBean.

- **Nivel de Agente**

Un agente es responsable de controlar y hacer disponibles los recursos manejados en el nivel de instrumentación, como también de administrar las relaciones entre ellos. En este nivel se proveen los requerimientos para implementar un agente.

Un componente esencial en este nivel es el MBean Server, servidor de MBean, este es un registro de MBean que permite ser accesibles a otras aplicaciones. Un MBean Server ofrece un servicio de consulta para los MBean, tras una consulta retorna los nombres de los MBean. Debido a que sólo se devuelvan los nombres, todas las operaciones en todos los MBeandeben pasar por el MBean Server.

- **Nivel de servicios distribuidos (Nivel de adaptación o Gestión)**

Nivel destinado a permitir una gestión cooperativa de redes de agentes y de sus recursos, en otras palabras ayuda a que las aplicaciones de administración interactúen con los agentes y sus objetos gestionados a través de adaptadores. En general, existe al menos un adaptador específico para cada protocolo de manejo o tecnología requerida para apoyar los diferentes sistemas de gestión, permitiendo a estos componentes ampliarse para una completa aplicación de gestión.

- **Gestión Remota**

A la tecnología JMX se puede acceder de diferentes maneras. Una de estas es que puede ser accedida por los protocolos existentes de gestión como SNMP [28]. El servidor se basa en protocolos para conectase y hacer que un agente JMX sea accesible por aplicaciones de gestión desde fuera de agentes de la Máquina Virtual de Java cuando sea necesario.

1.7.1 Relación entre Componentes JMX

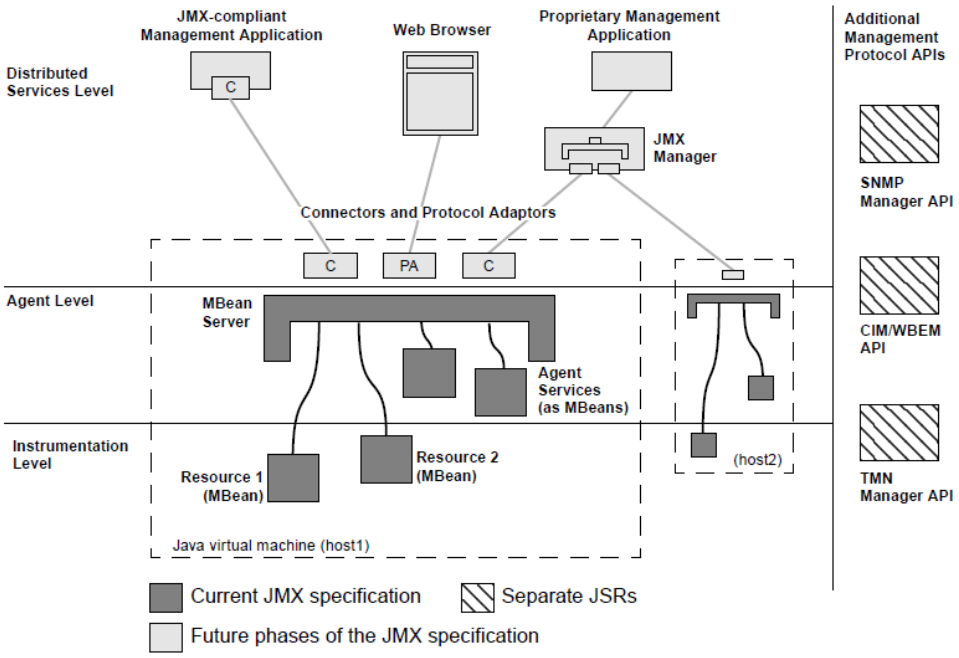


Figura 14. Relación entre Componentes JMX

1.7.1.1 Componentes en Nivel de Gestión

1.7.1.1.1 Aplicaciones de gestión compatibles con JMX.

Es una interfaz que permite el acceso a la aplicación de gestión a través de JMX. Un agente en la máquina virtual Java expone los servicios de agente, los llamados MBean, que pertenecen a los componentes que se ejecutan en la máquina virtual. Una aplicación de gestión compatible con JMX puede conectarse con el agente y tener acceso a los servicios disponibles de una manera estandarizada.

1.7.1.1.2 JMX Browser.

Un navegador o navegador web (del inglés, web browser) es el programa que permite ver la información que contiene la página web, este puede estar alojado en un servidor, dentro de la World Wide Web o en un servidor local.

El navegador interpreta el código, en el que está escrita la página web y lo presenta en pantalla permitiendo al usuario interactuar con su contenido y navegar hacia otros lugares de la red mediante enlaces o hipervínculos.

La funcionalidad básica del navegador web es permitir la visualización de documentos de texto. Los documentos que usualmente son páginas web, que pueden estar en la computadora en donde está el usuario, pero también pueden estar en cualquier otro dispositivo que esté conectado a la computadora del usuario o a través de Internet, y que tenga los recursos necesarios para la transmisión de los documentos. Tales documentos son comúnmente denominados páginas web.

Esta herramienta permite que el desarrollador y el administrador de un servidor J2EE (J2EE: Java 2 Platform, Enterprise Edition) controlen este servidor a través de la API JMX. Es una herramienta que permite ver e interactuar con el espacio JMX a través de una aplicación compatible con JMX. Ofrece las siguientes características:

- Conexión local (basada en la Web).
- Una vista jerárquica de los dominios y los MBean.
- Acceso a los atributos, las operaciones y notificaciones.
- El navegador JMX tiene dos formas:
- Una aplicación basadas en Swing, que se puede conectar remotamente e interactuar con el servidor JMX / aplicación.
- Una aplicación basada en Web, que se puede ejecutar en cualquier J2EE 1.3 Web. Proporciona una forma fácil de interactuar con el servidor JMX.

1.7.1.1.3 Aplicación de gestión Propietaria o privativa.

La expresión software privativo comenzó al ser utilizada por Richard Stallman, desde el año 2003, en sus conferencias sobre software libre, pues sería más adecuada que software propietario. El término privativo significa que causa privación o restricción de derechos o libertades, lo que se pretende es describir la privación a los usuarios de sus libertades en relación al software, esto desde el punto de vista de las organizaciones que apoyan las opciones de software libre.

1.7.1.1.4 Conector.

La aplicación de gestión utiliza los conectores para interactuar con los agentes. La biblioteca JDMK (JDMK: Java Dynamic Management Kit) de JMX soporta conectores para RMI, HTTP y HTTPS. Un conector permite que las clases de JVM remotas, es decir, clientes remotos JMX, los MBean y el MBean Server a través de JMX se traten como si fueran locales. Por ejemplo un objeto conector puede utilizar el protocolo CORBA para conectar un cliente JMX remoto a un MBean Server. De esta manera, en teoría, cualquier cliente con protocolo X debe ser capaz de conectarse a cualquier conector que soporta el protocolo X.

Las aplicaciones de gestión acceden a los MBean remotamente a través del Servidor MBean. La comunicación con el Servidor de MBean se realiza a través de un conector. Este elemento se encarga de transmitir las peticiones al servidor y de conectar los MBean registrados remotamente con el Servidor de MBean. El acceso también se puede realizar a través de los adaptadores de protocolos.

1.7.1.1.5 Adaptador de protocolo.

Los adaptadores de protocolos se utilizan cuando se quiere integrar el recurso representado por MBean en un entorno de gestión ya existente. Por ejemplo JDMK ofrece un conjunto de herramientas que facilitan la creación de agentes SNMP e integra la gestión SNMP en la arquitectura JMX. Para esto debe tener la implementación jar(s). Un adaptador de protocolo es sólo un cliente JMX que está escrito para que los programas no JMX se integren a la interfaz de su agente. El adaptador soporta todas las interfaces no JMX.

1.7.1.1.6 Clientes JMX.

Un cliente JMX es cualquier clase que utiliza un objeto MBean directamente o a través de una interfaz no JMX. Para hacer esto, la clase debe ser instanciada por un MBean, MBean Server o MBeanServerConnection, y debe tener la implementación Jar(s), un cliente JMX no invoca métodos sobre MBeans.

En general el cliente JMX usa instancias del MBean Server, este debería usar una interfaz MBeanServerConnection mientras sea posible, porque este último es portable (se puede invocar localmente o remotamente).

- **Cliente local JMX.** Un cliente local JMX se corre en la misma JVM que el MBean Server.
- **Cliente remoto JMX.** Un cliente remoto JMX es como un cliente local JMX, excepto que debe crear instancias de un cliente conector y obtener una conexión al MBean Server a través del método MBeanServerConnection. A continuación, utiliza el MBeanServerConnection de la misma forma que un cliente local JMX.
- **Cliente de Administración.** Un programa cliente se usa para la gestión de MBean. El cliente de administración no suele ser un cliente JMX, pero podría serlo. El punto esencial es que se trata de un programa de usuario final que de alguna manera inicia comandos que terminan invocando métodos MBean o servicios. Un ejemplo es un navegador web conectado a un adaptador de HTML. Otro ejemplo es una GUI de administración SNMP existente, que puede enviar comandos a un adaptador de Protocolo SNMP. En estos casos típicos, tanto el navegador y el GUI de gestión SNMP, no saben nada de JMX.

1.7.1.2 Componentes en el Nivel de Agentes

1.7.1.2.1 Mbean Server.

El servidor de MBean es el elemento central desde el que se realiza la gestión. Este elemento almacena los MBean creados en el propio servidor, así como representaciones de MBean remotas. Su función es dar información sobre los MBean que tiene registrados y sobre sus servicios, dar acceso a dichos servicios y ofrecer servicios propios (ya sea para la gestión de MBean o para cualquier otra tarea).

1.7.1.2.2 Servicios de agente.

Son objetos que pueden realizar operaciones de gestión sobre los MBean registrados en el servidor MBean. Al incluir inteligencia de gestión en el agente, JMX le ayuda a construir soluciones de gestión más potentes. Los Servicios de agente son a menudo MBean así, permite que ellos y sus funcionalidades sean controlados a través del servidor MBean.

Algunas responsabilidades de Agente son:

- Cargar algunos MBean en el inicio.
- Cargar e inicializar conectores.
- MBean Store para el almacenamiento permanente para que puedan cargarse de nuevo cuando se reinicie la máquina virtual de Java.

- Dar formato y administrar los servicios de agente, tales como carga dinámica de clases, monitoreo, temporización.

1.7.1.3 Componentes en el Nivel de Instrumentación

1.7.1.3.1 MBEAN.

Este es el nombre más común para los Bean gestionados, se trata de objetos que pueden ser almacenados en el repositorio (que es el servidor de MBean). Después que un MBean se almacena en un repositorio de clases, otras clases Java pueden tener acceso a métodos específicos y descripciones del MBean. Hacer un MBean no es tan simple como implementar una interfaz MBean (no hay clases o interfaz denominada MBean en una implementación JMX.

Las Extensiones de Gestión de Java definen una arquitectura, API, servicios y patrones de diseño para aplicaciones, gestión de red, y monitoreo utilizando el lenguaje de programación Java [29]. Aporta mecanismos para crear agentes y gestores en este lenguaje, implementar aplicaciones de gestión distribuida e integrar éstas a aplicaciones de gestión existentes a través de las API que implementan protocolos de gestión estandarizados y ampliamente utilizados en la industria.

Usualmente esta tecnología incluye la configuración de aplicaciones, sigue el comportamiento de la aplicación y notifica cuando la aplicación sufre algún cambio. La especificación JMX es una arquitectura estándar de patrones de diseño y aplicación de servicios, gestión de redes y monitoreo en el lenguaje Java [30]. El recurso de la tecnología JMX es un servicio en tiempo de ejecución que es gestionado por uno o más objetos Java como Bean o MBean. Donde los MBean son registrados en el servidor de MBean. Este servidor se conoce como servidor de objetos, el cual es un agente de gestión que puede ejecutarse en la mayoría de dispositivos con lenguaje java.

1.8 Comparación de tecnologías de Gestión.

La forma en que se ha desarrollado el proyecto es como sigue: se identificó diferentes tecnologías de gestión, en términos de sus características técnicas, servicios y otras cuestiones adicionales. Así se pueden comparar y exponer los puntos a favor y en contra de cada una de ellas, esto se expondrá más adelante en el documento.

Después de estudiar las tecnologías de gestión cabe mencionar sus diferencias, ventajas y desventajas, para así seleccionar la mejor alternativa para este proyecto. Para esto se hace un análisis comparando WBEM, CORBA, y JMX. A continuación se observan las características generales de un sistema de gestión según organismos que regulan estas tecnologías de gestión de redes y servicios.

En el caso del Open Group, un sistema de gestión debe tratar de ser portable, interoperable, transparente, extensible y robusto. Para el TeleManagement Forum un sistema de gestión debe usar sistemas distribuidos, reutilizar componentes, usar un diseño orientado a objetos, manteniendo sistemas heredados, y dando acceso al sistema con herramientas de propósito general y a bajo costo [31]. Lo que se observa es que hasta el momento todas las tecnologías estudiadas tienen algunas o todas estas características, entonces se debe profundizar el análisis.

De CORBA se tiene que unifica el lenguaje de especificación de la información de gestión mediante el uso de IDL, pero ésta tecnología no define cómo interactuar con otros dominios de gestión, por tanto solo se usa tecnologías que posean una interfaz CORBA.

Para resolver el problema de interoperabilidad entre los múltiples marcos de gestión existentes, el DMTF propuso WBEM cuyo elemento central es el CIM Modelo Común de Información, que aporta un lenguaje de modelado de información, como pueden ser SMI o GDMO, basado en UML (Unified Modelling Language, Lenguaje de Modelado Unificado), con el que se trata de modelar toda la información de gestión existente, incluyendo la definida por los lenguajes anteriores. Los esquemas CIM son MIB que tratan de definir varias áreas de la gestión: Sistemas, Dispositivos, Red, Aplicaciones, Inventario, etc. Pero que no tienen una correspondencia exacta con las MIB de los otros marcos de gestión.

WBEM, es la arquitectura sobre la que se sustenta CIM. Busca llevar a cabo gestión integrada de los recursos, en términos de las FCAPS empleando las tecnologías que han dado éxito al web.

De lo anterior se puede deducir que WBEM no sería adecuada ya que como se dijo anteriormente ésta hace gestión de los recursos en términos de FCAPS, lo que implica la Gestión de Fallos, proceso equivalente a la Gestión de Incidentes en el marco ITIL, esto indica que el proceso de Gestión de Fallos tiene sus pasos ya estipulados para poder ser implementado, y estos pasos son diferentes a los propuestos en ITIL – LITE, lo que imposibilitaría llegar al objetivo propuesto.

Por otra parte de las Extensiones de Gestión Java (JMX) se dice que no es una arquitectura de gestión, sino una arquitectura de instrumentación de la gestión. Se menciona que JMX es únicamente un conjunto de bibliotecas de Java que facilitan la instrumentación de aplicaciones de una forma más sencilla, sin importar el protocolo de intercambio de información. Por tanto es a partir de este conjunto de bibliotecas que se podría diseñar una solución de gestión de incidentes. De lo anterior JMX tomaría ventaja con respecto a las demás estudiadas, pero el análisis se profundiza sacando ventajas y desventajas, además también se deben tener en cuenta necesidades específicas del proyecto.

Tabla 2. Ventajas y Desventajas de las Tecnologías Analizadas

TECNOLOGÍA	VENTAJAS	DESVENTAJAS
CORBA	• La portabilidad, transparencia, extensibilidad y robustez. • Reutilización de componentes, diseño orientado a objetos, mantenimiento de sistemas heredados y acceso al sistema de propósito general y bajo costo.	• El IDL es una tecnología que no define cómo interactuar con otros dominios de gestión, por tanto solo se usa tecnologías que posean una interfaz CORBA. • Usa IDL, que es menos potente que CIM para el diseño específico de información de gestión. • Es un estándar que establece una plataforma de desarrollo de sistemas distribuidos.
WBEM	• Unifica todos los posibles modelos de información existentes. Para ello, se hace uso de CIM, un modelo bastante potente y orientado a objetos y basado en UML. • Existe una integración total de las tecnologías web en esta tecnología, cumpliendo las exigencias de reusabilidad y bajo costo.	• Utiliza las FCAPS que implica Gestión de Fallos, proceso equivalente a la Gestión de Incidentes del marco ITIL, ésta emplea diferentes pasos para ser implementada. • Falta de modularidad, no es posible desplegar aplicaciones de forma que un cliente tenga una interfaz de acceso única.
JMX	• Las múltiples bibliotecas definidas en JMX dan la posibilidad de usar cualquier protocolo. • El uso de Java permite su despliegue en cualquier sistema operativo. • La información se puede definir en un lenguaje orientado a objetos, utilizando la estructura de Mbean. • Al ser una arquitectura de instrumentación de la gestión posee un conjunto de bibliotecas que permiten diseñar una solución.	• JMX está centrado en Java, lo que limita su aplicabilidad con otros lenguajes de programación, pero, el uso de IIOP solventa la interoperabilidad entre códigos escritos con distintos lenguajes.

La anterior tabla 2, expone las ventajas y desventajas de las diferentes tecnologías de gestión con respecto al proyecto, no respecto a la tecnología en sí misma.

1.8.1 Porque usar JMX

Después de comparar las diferentes tecnologías de gestión, se define que la tecnología más apropiada para soportar la gestión de incidentes basada en el marco de referencia ITIL, es la tecnología JMX, ya que brinda una serie de herramientas no solo a la hora de escribir código Java sino también a la hora de hacer gestión de aplicaciones, los agentes Java distribuyen la gestión entre directivos y mandos medios. Esto es integrado dentro de los sistemas existentes de gestión y monitoreo. Las razones del porque utilizar JMX se muestran a continuación:

1. Las múltiples bibliotecas definidas en JMX dan la posibilidad de usar cualquier protocolo.

2. El uso de Java permite su despliegue en cualquier sistema operativo.

3. La información se puede definir en un lenguaje orientado a objetos, utilizando la estructura de Mbean.

4. Al ser una arquitectura de instrumentación de la gestión posee un conjunto de bibliotecas que permiten diseñar una solución de gestión.

5. Porque aprovecha e integra las tecnologías y estándares de Java, como JNDI (JNDI: Java Naming and Directory Interface), JDBC (JDBC: Java Database Conecting API), JTS (JTS: Java Transaction Services) soluciones a la gestión.

6. Porque la Máquina Virtual de Java es altamente instrumentada lo cual es una característica secundaria de JMX.

7. Porque las especificaciones de JMX se pueden referenciar con las especificaciones existentes de Java.

8. La tecnología JMX puede crearse desde un módulo del IDE NetBeans.

9. La especificación JMX es una arquitectura estándar de patrones de diseño y aplicación de servicios, gestión de redes y monitoreo en el lenguaje Java.

Capítulo 2. Diseño de la Solución para la Gestión de un Centrex IP, Basado en ITIL-LITE.

2.1 Ambiente del Sistema

Se propone un diseño de referencia para crear una solución que sea capaz de ofrecer funcionalidades para la Gestión de Incidentes con el fin de ayudar a la Centrex en la prestación de un buen servicio a través de un Service Desk. Así, el ambiente del sistema se concibió a partir de los elementos conceptuales y funcionalidades de ITIL–LITE.

El proceso inicia cuando ocurre algún incidente, éstos serán detectados automáticamente por herramientas de monitoreo, o serán reportados. En el Service Desk se llevaría acabo los soportes de primero y segundo nivel, aquí se debe tener conocimiento de los incidentes que tienen que enfrentar los técnicos, también se contaría con niveles de soporte adicionales, por ejemplo, el de nivel tres que proporcionaría soporte por un técnico externo a la empresa, en áreas especializadas tales como hardware, sistemas operativos o aplicaciones específicas de software.

Todos los incidentes se registran en una base de datos. Estos incidentes proporcionan información valiosa para la organización, que puede usar a su conveniencia para tomar decisiones acerca del mejoramiento del negocio. La diferencia con otros Service Desk es que a la información consignada se le aplicarán los pasos recomendados para Gestión de Incidentes de ITIL-LITE. Después de procesada la información, el incidente específico es gestionado para que el administrador de IT examine el caso para decidir si el incidente se ha resuelto o es un problema más global que debe abordar otra área de gestión.

Figura 15. Ambiente del Sistema

Tomando como punto de partida el ambiente del sistema, en la siguiente sección se define un diseño de referencia para la Gestión de Incidentes de un Centrex IP.

2.2 Descripción General

En esta sección se muestra el sistema y sus componentes, relación entre ellos y con el ambiente, principios que guían su construcción y evolución. Se describe la estructura

del sistema, la cual abarca componentes de software, propiedades externas visibles de estos componentes y sus relaciones. El diseño de la solución propuesto consiste de los siguientes elementos:

- Una descripción de los componentes del sistema
- Definición de las relaciones entre los componentes
- Definición de las relaciones entre los componentes del sistema y los elementos externos al sistema

El principal problema que debe abordar el diseño de referencia es definir una estructura general y las relaciones entre los principales elementos funcionales del sistema (procesos), debe indicar el flujo de datos, control entre los procesos e identificar los principales requisitos de desempeño, almacenamientos de datos e interfaces entre los componentes del sistema.

La Figura 16 representa el diseño de referencia propuesto para la Gestión de Incidentes, que se sustenta sobre cuatro niveles: Nivel de Gestión, Nivel de Agentes, Nivel de Instrumentación y Nivel de Sistema Gestionado, estos niveles agrupan los módulos necesarios para llevar a cabo la Gestión Incidentes.

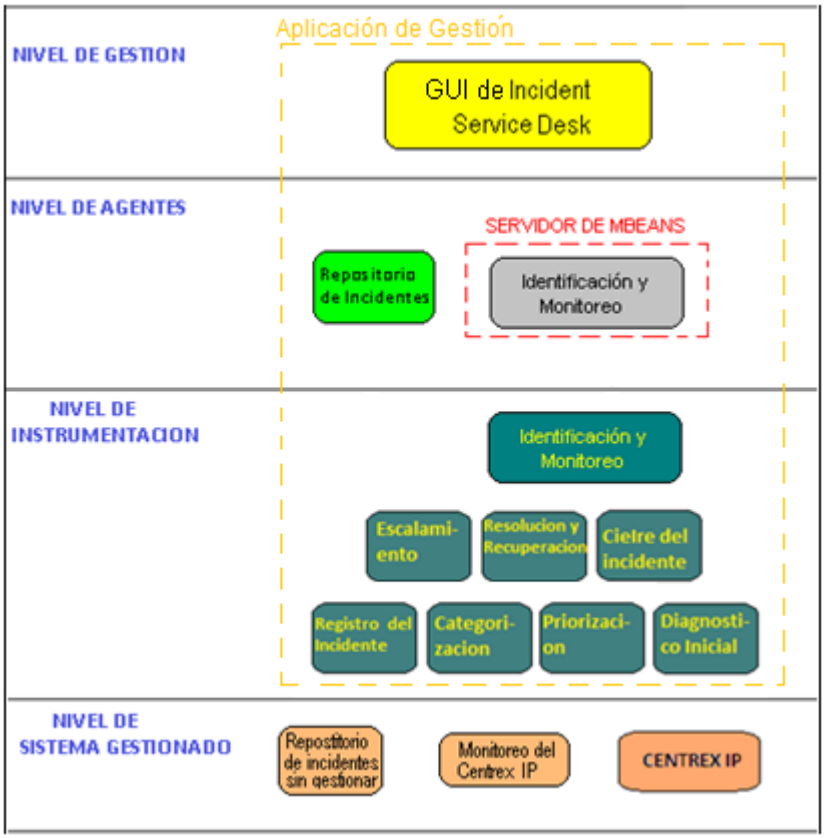


Figura 16. Diseño de Referencia Propuesto

A continuación se describe cada uno de los niveles que conforman el diseño de referencia.

2.2.1 Nivel de Gestión

El nivel superior corresponde a las interfaces administrativas, las cuales proporcionan funcionalidades necesarias para que los usuarios (administrador o técnico) interactúen con los demás niveles. Estas interfaces son necesarias para que los usuarios aprovechen los procesos que expone la Gestión de Incidentes.

Este nivel permite, si fuese el caso, implementar adaptadores de protocolo en el servidor JMX hacia otros protocolos de gestión, implementa los MBean que interactúan con otros protocolos de gestión, exponiendo sus atributos como un recurso gestionable. Los adaptadores de protocolo permiten que las aplicaciones de gestión accedan al servidor JMX y manipulen los MBean que lo conforman.

- **GUI del Incident Service Desk:** es una herramienta de Gestión de Incidentes que se basa en el uso de imágenes y objetos gráficos, su función es dar soporte a la organización, permite visualizar incidentes que se detectan automáticamente provenientes de la Centrex, para luego ser procesados y de ser necesario se notifica a los administradores para tomar las acciones de gestión necesarias. Permite dar mejores soluciones en asuntos de soporte técnico, ayuda a las organizaciones en la administración de su complejidad y de su infraestructura IT.

- **Conector:** este elemento es conectado al Servidor de MBean dela API de JMX, hace que el Servidor sea accesible remotamente o de forma local por un cliente de tecnología basada en java. El cliente final de un conector exporta inicialmente la misma interfaz que el Servidor de MBean.

2.2.2 Nivel de Agentes

En este nivel se encuentra el servidor de MBeans, que agrupa los nombres de los MBean necesarios para realizar la Gestión de Incidentes por parte de las aplicaciones de Gestión, la GUI del Inciden Service Desk es la aplicación que accede a este nivel. El servidor de MBean ofrece un servicio de consulta para obtener los MBean, retornando sus nombres. Su funcionalidad es la de controlar y hacer asequibles los recursos manejados en el nivel de instrumentación, además de administrar las relaciones entre ellos. En este nivel también se encuentra el módulo involucrado en almacenar la información relativa a los incidentes y los módulos mencionados anteriormente.

- **Servidor de MBean:** se asemeja a un directorio telefónico en el cual se buscan los nombres de los MBean que representan a las funcionalidades de gestión ubicadas en el nivel de instrumentación. En este proyecto se cuenta con el MBean de Identificación y monitoreo de los incidentes del Centrex IP.

- **Repositorio de Incidentes:** almacena los incidentes que son de interés al construir la solución, aquí llegan sólo los incidentes que tienen un grado de importancia, la información con la que llegan puede incluir fecha y hora, número único de referencia, priorización, entre otros datos que se consideren relevantes en la construcción del Incident Service Desk. Lo anterior se realiza para mantener un registro histórico de los incidentes ocurridos en los servicios del Centrex IP, de

modo que si el incidente tiene que ser referido a otro(s) grupo(s) de soporte, ellos contarán con la información necesaria para enfrentarlos.

2.2.3 Nivel de Instrumentación

En este nivel se encuentran los MBean y las funcionalidades, las cuales son una abstracción de las recomendaciones del marco de referencia ITIL- LITE, que facilitan la información necesaria para la gestión. En el momento de la implementación pueden aparecer nuevas funcionalidades que podrían ser útiles para mejorar la herramienta de gestión, sin importar que estas estén por fuera de las recomendaciones de ITIL – LITE.

Este nivel cuenta con las siguientes Clases:

- **MBean Identificación y Monitoreo:** se encarga de identificar los incidentes detectados por herramientas de monitoreo que llegan del Centrex IP, como también los que lleguen de otras fuentes, después de identificados guarda esta información en una base de datos que hará las veces de repositorio, la información quedará presta para continuar con los procesos de gestión.

- **Registro de Incidentes:** en este inciso se registra principalmente la fecha y hora del reporte del incidente y fecha y hora del cierre del mismo. Además otros datos que se podrían incluir son: número único de referencia, categoría del incidente, prioridad, entre otros.

- **Categorización:** este paso da inicio con el registro inicial, esta categorización se hace con las siguientes categorías: Software-Freeswitch, Software-Syslog, Software-Zoiper y Hardware.

- **Priorización:** se puede determinar teniendo en cuenta la urgencia del incidente y el nivel de impacto en la organización o haciendo un mapeo de la prioridad asignada por otras herramientas, tanto el impacto como la urgencia serán acordados previamente, finalmente se asigna un código numérico de prioridad que puede ir de uno a cinco (1-5) o una descripción de prioridad (crítica, alta, media, baja, en estudio) según la prioridad se determinaran unos límites de tiempo a la hora de resolver cada tipo de incidente.

- **Diagnóstico Inicial:** este campo brinda la funcionalidad de soporte de primer nivel, el cual busca dar solución al incidente mientras el cliente del servicio de VoIP se encuentra aún en su equipo personal, para realizar esto primero se hace un almacenamiento de la descripción y los pasos que dan la solución de algunos incidentes, esto se podría hacer escogiendo características de un menú que aparecería en la ventana del Incident Service Desk, este diagnóstico permite que el sistema responda con una posible solución que se encuentra predeterminada.

- **Escalamiento:** el sistema sería capaz de permitir realizar el escalamiento que se hace de la siguiente forma: el primer punto de soporte y primer nivel de escalamiento es en el Incident Service Desk, si éste no es capaz de resolverlo pide apoyo al segundo nivel de soporte que en este caso pueden ser técnicos,

cada uno experto en su área, si estos no encuentran la solución se pasa a un tercer nivel de escalamiento que son técnicos externos a la empresa, y si finalmente no hay solución por estos últimos, se escala a un cuarto nivel que sería informar a los directivos de la empresa, quienes son los que toman las decisiones que mejor convenga.

- **Solución y Recuperación:** esta funcionalidad pide al técnico que lleve a cabo actividades que han sido previamente almacenadas, si con ellas no encuentra solución debe buscar pasos que conlleven al restablecimiento del servicio, si dichos pasos funcionan el sistema permitirá guardar este procedimiento.

- **Cierre de Incidentes:** se encargaría de cerrar el registro de incidentes, se puede utilizar un tiempo de cierre específico o automático. El sistema puede tomar la fecha y hora del equipo para asignarla como fecha de cierre del incidente cuando se ha comprobado que éste ha sido resuelto completamente, antes de cerrar también se verifica que la categoría y prioridad sean correctas.

2.2.4 Nivel de Sistema Gestionado

En este nivel se encuentran los servicios de telefonía de VoIP contenidos en la Central PBX IP, también se tiene la herramienta que extrae incidentes de la Central que luego se almacenan en el repositorio que está ubicado en este mismo nivel.

- **Centrex IP:** es una central telefónica (conmutador telefónico) que provee servicios de VoIP que son ofrecidos a las empresas. Puede prestar servicios como: establecimiento de llamada, correo de Voz, conferencia, desvío ocupado, desvío sin respuesta, desvío directo, llamada en espera, recoger llamadas, transferencia, recoger llamadas directas, etc.

- **Monitoreo del Centrex IP:** este módulo permite que se pueda monitorear una central telefónica IP. Es un software que extrae y almacena incidentes automáticamente que se generan en la Centrex IP, esta herramienta almacena en un repositorio información acerca de los incidentes, para luego ser gestionados según las recomendaciones de ITIL- LITE.

- **Repositorio de Incidentes sin Gestionar:** aquí se almacena la información que genera la Centrex IP, en forma de eventos e incidentes que son la materia prima para que luego sea procesada por el siguiente nivel de la arquitectura (Nivel de Instrumentación).

2.3 Relaciones

En este apartado se explican las relaciones entre los distintos módulos de la solución planteada y cómo interactúan con actores externos, también se describe la forma como los módulos colaboran para soportar las funcionalidades básicas (identificación y monitoreo, registro, categorización, etc.). La siguiente figura17presenta el diagrama de relaciones entre los módulos del diseño de referencia.

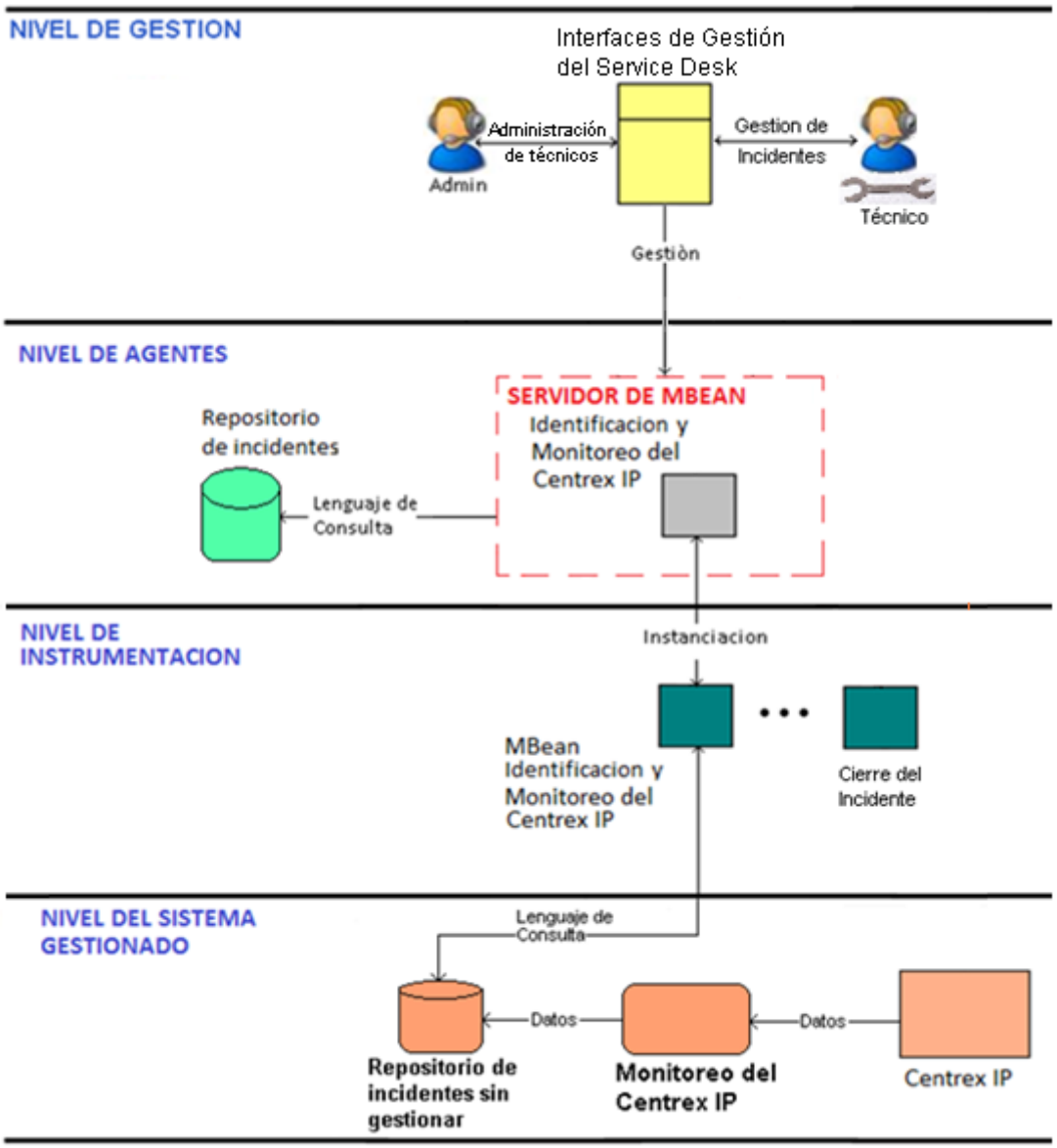


Figura 17. Diagrama Relaciones entre Componentes

Los actores que interactúan con el sistema son dos: el Administrador y el Técnico. El administrador es quien tiene asignado todos los derechos para administrar el Incident Service Desk y también la posibilidad de gestionar técnicos. El técnico es el encargado de la gestión de incidentes como también de actualizar información referente a ellos en las bases de datos.

Las características en la interfaz de entrada al sistema se habilitan de acuerdo al perfil de usuario (administrador o técnico) y como consecuencia los privilegios en el Incident Service Desk, les permite gestionar incidentes y ver tickets.

En los siguientes párrafos se explican las relaciones entre cada uno de los módulos, y la información que intercambian.

- **Interfaces de Gestión – Servidor de MBean:** intercambian los datos requeridos o generados cuando ocurren incidentes. Se aprovechan las facilidades de

presentación contenidas por las interfaces para capturar los datos necesarios para invocar un servicio de gestión y mostrar la información que se genera como resultado después de ejecutar los procesos de gestión.

- **Servidor de MBean–Repositorio de Incidentes:** esta relación representa las consultas que se realizan al repositorio de incidentes con el fin de obtener la información necesaria para realizar la gestión. Elementalmente el repositorio brinda un lenguaje de consulta con el que se pueden realizar las funciones de búsqueda y actualización de datos almacenados en el repositorio.

- **Agente – MBean:** estos componentes tienen una relación que es de instanciación, esto es debido a que los objetos MBean son instancias de los agentes, ya que las extensiones de Java pueden hacer uso de las capacidades ofrecidas por la Máquina Virtual de Java, que en definitiva permite la comunicación interna y de forma transparente entre estos componentes, lo que en alguna literatura llaman el mundo JMX.

- **MBean – Repositorio de Incidentes sin Gestionar:** posee un lenguaje de consulta que permite monitorear incidentes cada vez que llegan a dicho repositorio, con el fin de luego procesar esta información en el siguiente nivel.

- **Repositorio de Incidentes sin Gestionar – Monitoreo de Centrex IP:** el módulo de monitoreo del Centrex se encarga de extraer los incidentes para luego almacenar dichos datos en el repositorio mencionado. Dicha relación se hace con el fin de tener información permanente de los incidentes, de lo contrario se perderían datos para el proceso de gestión, ya que los incidentes generados y almacenados en los archivos de la Centrex se eliminan cada cierto tiempo.

- **Monitoreo de Centrex IP –Centrex IP:** dicha relación permite el intercambio de datos requeridos y generados mientras la Centrex IP está operando, ya que este módulo está constantemente vigilando si se producen incidentes en la Central IP, que puedan degradar o interrumpir el servicio que se ofrece a los clientes de la MIPYME.

2.4 Módulos

En esta sección se describe la estructura interna de los principales módulos que pertenecen al diseño de referencia. Se establecen los componentes y las relaciones que se crean para desempeñar sus labores.

2.4.1 Aplicación de Gestión

Para las aplicaciones de usuario se propone emplear el patrón MVC (MVC: Model View Controller), que de acuerdo a [32] fue introducido como parte del lenguaje de programación Smalltalk. Fue diseñado para reducir el esfuerzo de programación necesario en la implementación de sistemas. Sus características principales son que: el Modelo, las Vistas y los Controladores se tratan como entidades separadas.

MVC es un patrón de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control, en tres componentes distintos [33]. De

esta manera el diseño queda más simple y permite reutilizar mucho más código ya que, las partes a implementar, quedan mucho más definidas y son idénticas a las de otros proyectos.
En la figura 18se muestra el patrón MVC en su forma más general.

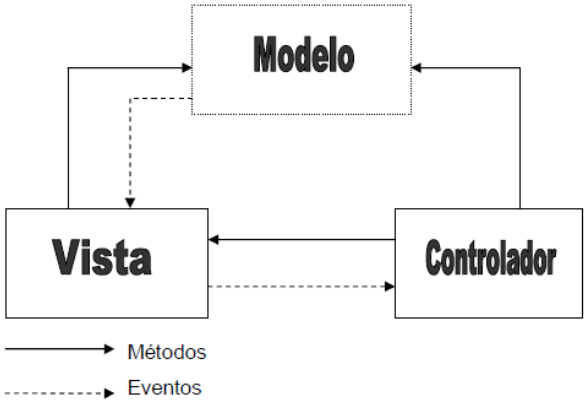


Figura 18. Patrón MVC

Se continúa definiendo los componentes internos y las relaciones de la aplicación de gestión, además se muestra la interacción con los componentes externos, para realizar las funciones de Gestión de Incidentes de la Centrex, como se muestra en la figura 19.

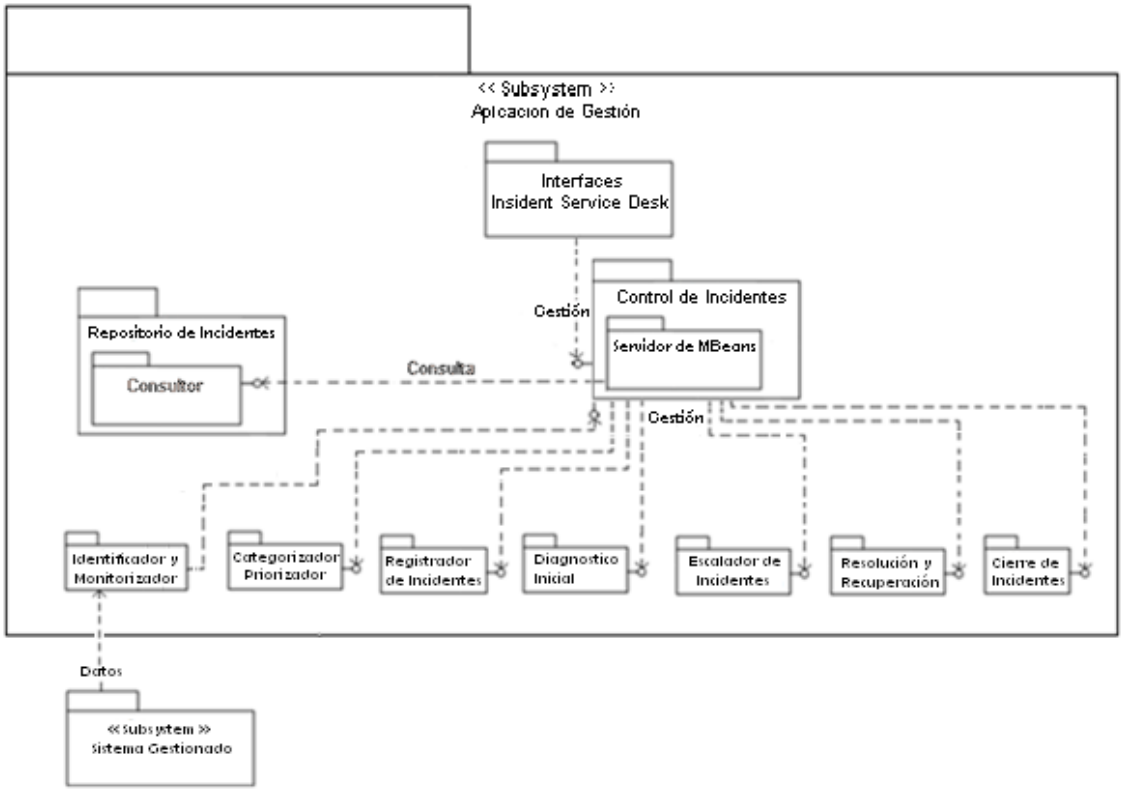


Figura 19. Componentes del Módulo Aplicación de Gestión

Algunos de los componentes incorporados en el módulo Aplicación de Gestión se describen a continuación:

- **Interfaces:** son las ventanas que sirven de mediador entre el técnico o administrador con la aplicación de gestión, haciendo uso de los signos gráficos. Desde estas interfaces también se permite la entrada o denegación de los usuarios al sistema, entre muchas funciones.

- **Control de Incidentes:** se encarga de coordinar o controlar los componentes internos para realizar la Gestión de Incidentes, encierra la lógica que determina los pasos a seguir para realizar esta gestión. Este componente recibe las peticiones provenientes desde las interfaces de usuario, para iniciar el proceso de Gestión de Incidentes basado en lo que requiere el usuario. Una vez terminada la Gestión del incidente, este devuelve los resultados obtenidos a las interfaces de gestión para que se complete el requerimiento.

- **Identificador y Monitorizador:** se encarga de detectar incidentes provenientes de la Centrex IP, este bloque permite saber que ha llegado un incidente, para luego enviar esta información al control, quien decidiría con que paso se debe continuar en la gestión.

- **Registrador:** soporta las funcionalidades para registrar el incidente con su fecha y hora, también puede incluir: número único de referencia, categorización de incidente, priorización del incidente, etc.

- **Categorizador y Priorizador**: tiene la funcionalidad de darle una categoría según sea del tipo hardware, software, aplicación, etc. Toma los datos del incidente, para luego con esta información asignar la prioridad de dicha falla, lo que determinará los tiempos límites de solución.

- **Diagnóstico inicial**: este componente provee soluciones automáticas, sencillas y previamente acordadas, si se puede se dará solución mientras el usuario aún está en el equipo, si al incidente se le encuentra solución pasará al bloque de cierre, de lo contrario se escalará a un nivel de soporte superior.

- **Escalador de Incidentes**: implementa la funcionalidad de cambiar la escala de incidentes según se necesite, este dato se utilizará con otra información adecuada para generar un ticket que se muestra en pantalla. El nivel de escala será de valor 1 cuando la solución que esta previamente almacenada sirva para dar soporte al incidente, de lo contrario tendrá el valor 2 que indica que el técnico interno deberá encontrar los pasos que den solución al incidente y estos deberán ser registrados. Si el técnico interno no lo pudo resolver, a través de una interfaz, él informará al Incident Service Desk aumentando el nivel de escala, si en el nivel 3 no se resuelve, entonces se hará un escalado jerárquico (nivel 4), es decir que se le informara al gerente de la empresa sobre lo ocurrido.

- **Solución y Recuperación**: se encarga de registrar la solución encontrada, de manera que en posteriores incidentes del mismo tipo se obtenga la solución de la forma más rápida, generándose una base de conocimiento. Este incidente se pasa al componente de cierre para terminar el proceso.

- **Cierre de Incidente:** este componente recibe un evento que indica que hubo solución el cual desencadenara una serie de respuestas del sistema, como pueden ser tomar la fecha y hora del equipo para luego adicionarla como un nuevo dato en el campo de fecha de cierre del sistema.

- **Repositorio de Incidentes:** en la descripción del repositorio de Incidentes se establece un diseño común (ver Figura 20) para este repositorio, identificando los componentes necesarios para gestionar los datos de incidentes almacenados.

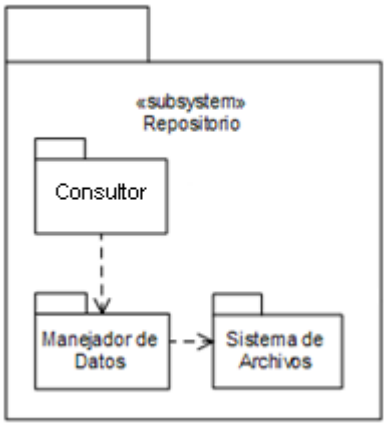


Figura 20. Diseño del Repositorio de Incidentes

Los componentes que hacen parte del repositorio son los siguientes:

- **Sistema de Archivos:** representa a las estructuras de datos contenidas en una unidad de almacenamiento y este permite representar la información de los incidentes.

- **Manejador de Datos:** este componente permite obtener acceso a datos almacenados en el sistema de archivos. Brinda independencia a los niveles superiores del sistema de almacenamiento escogido a través de una implementación determinada. Se puede contar con distintos tipos de almacenamiento como son bases de datos, archivos planos, XML, entre otros. Para tener control de esto se implementa un manejador de datos específico, y este es transparente a niveles superiores, a la hora de manipular datos almacenados.

- **Consultor:** provee de una interfaz que permite acceder a los datos almacenados en el sistema de archivos al software externo, hace parte del núcleo del repositorio. Este elemento contiene una clase para gestionar los datos almacenados, implementando diversas funcionalidades como creación, actualización y eliminación de datos, permite hacer consultas sobre el repositorio para obtener información sobre los incidentes.

La gestión es iniciada cuando se detecta automáticamente un incidente de la Centrex IP a través del Incident Service Desk. El componente Control de Incidentes recibe la invocación e inicia el manejo de los demás componentes para realizar la Gestión de Incidentes. Los objetos se han creado con el fin de cumplir los requerimientos según

las recomendaciones de ITIL-LITE, dentro de otras funcionalidades puede estar el cambio de contraseñas de los técnicos y administrador, llenar tickets de las acciones tomadas por ellos, realizar búsquedas en el repositorio, etc. Finalmente, cuando el incidente es totalmente solucionado, es marcado por el componente de cierre como resuelto en el Incident Service Desk.

2.5 Alcance y Características del Diseño de Referencia

Esta sección describe hasta donde abarca el diseño de la solución de referencia y sus características. Como primera medida se define el alcance del mismo y las decisiones de diseño que se toman en cuenta. Luego, se describe el modelo de despliegue exponiendo algunos de sus diagramas y sus respectivos componentes.

2.5.1 Alcance

El diseño propuesto de referencia busca alcanzar dos objetivos principales, relacionados a los objetivos de esta tesis:

1. Diseñar una solución con las funcionalidades necesarias para soportar la Gestión de Incidentes en un Centrex IP, utilizando los conceptos de ITIL-LITE.

2. Desarrollar un prototipo que permita comprobar experimentalmente el diseño de la solución propuesta.

2.5.1.1 Características

Según los propósitos se puede definir las siguientes características para la implementación de referencia propuesta:

- **Monitorización de la Centrex IP:** la implementación de referencia soportaría las funcionalidades necesarias para realizar la monitorización automática de la Centrex IP.

- **Gestión de Incidentes:** debe incluir las funciones de Gestión de Incidentes basándose en las recomendaciones de ITIL-LITE, aplicadas a la gestión de una Centrex IP.

- **Gestión de Servicios:** la implementación de referencia debe ser probada en un contexto de gestión de servicios de la Centrex IP, esta característica permite realizar la integración de todo el sistema.

2.5.1.2 Restricciones

En esta sección se enumeran las restricciones que se consideraron para realizar el diseño, para luego llevar a cabo la implementación del presente trabajo:

1. Teniendo en cuenta que las recomendaciones de ITIL dentro de la operación del servicio, tiene como proceso la Gestión de Incidentes y ésta a la vez propone ciertas funcionalidades para realizar dicha gestión, se excluye la Investigación y Diagnostico en la construcción del diseño de referencia, ya que este componente nos es viable para ser tomado en cuenta por razones ya expuestas en la sección 1.3.3.7 Investigación y Diagnostico.

2. En el diseño propuesto se excluye el uso de Adaptadores de Protocolos, que permiten comunicar tanto los clientes de gestión como también el sistema gestionado al Servidor de MBean, ya que estos se utilizan para conectar aplicaciones externas e integrarlas al mundo JMX.

2.5.2 Diseño Final

En la figura 21 se muestra el diseño final de la solución de gestión, sustentado en cuatro niveles como se ha planteado anteriormente.

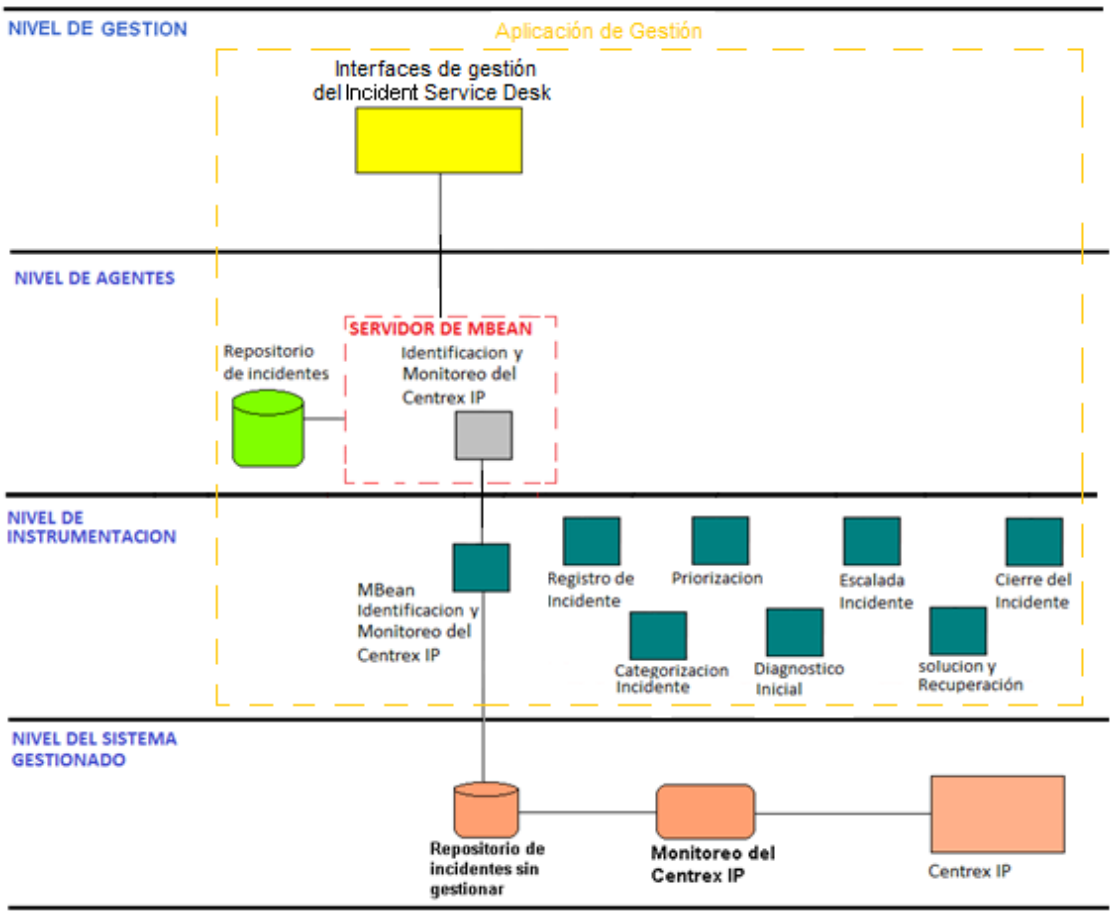


Figura 21. Diseño Final

De la anterior figura se puede observar que en el nivel de gestión se encuentran las interfaces gráficas de Gestión, éstas permiten comunicar al usuario con el interior de la aplicación de Gestión.
El Servidor contiene Agentes que brindan las funcionalidades que abstraen las recomendaciones estudiadas de ITIL, del diseño se puede intuir que en la implementación sería práctico unir algunos de los componentes del sistema, ya que sus funcionalidades están directamente relacionadas y esta unión puede que facilite su implementación. Para la comunicación entre el Servidor de MBean y el Repositorio de Incidentes se usa la API JDBC (JDBC: Java Database Connectivity), que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java. Cabe mencionar la existencia de un consultor del Repositorio de Incidentes que

sirve de mediador entre el Repositorio y el Servidor de MBean, este mediador realiza todo tipo de consultas sobre el repositorio.

2.5.3 Modelo de Despliegue

En este apartado se presentan el diagrama de paquetes y el diagrama de despliegue que se pretende implementar.

2.5.3.1 Diagrama de Paquetes

La figura 22 muestra el diagrama de paquetes del diseño de referencia y se describen cada una de las capas contenidas en el diagrama.

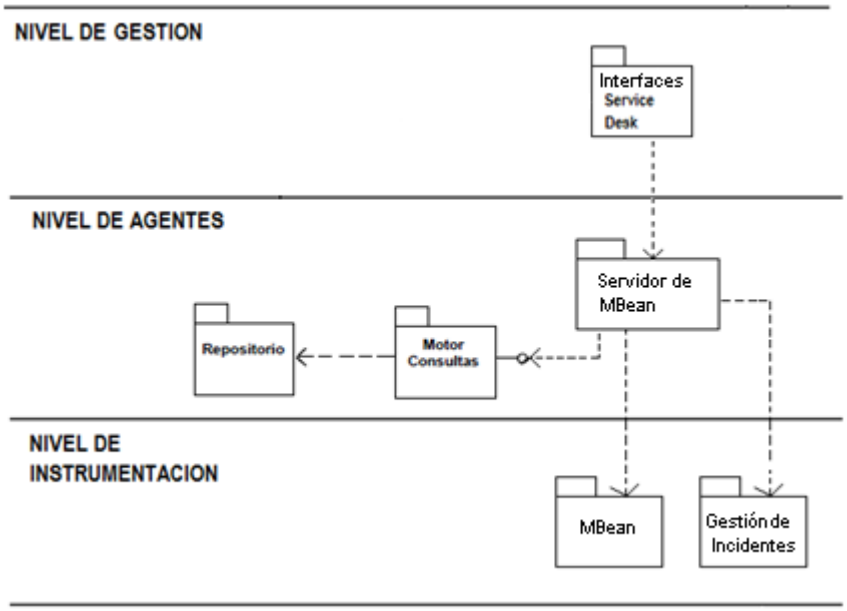


Figura 22. Diagrama de Paquetes de la Aplicación

Nivel de Gestión: se compone de un paquete correspondiente a las interfaces del Incident Service Desk a través de las cuales el usuario interactúa con la aplicación de Gestión de Incidentes. Este paquete se implementa con el lenguaje de programación Java y se soportan sobre el JDK 6.

Nivel de Agentes: abarca los paquetes de Servidor de MBean, un motor de consultas, y también incorpora el repositorio de incidentes. Para realizar las consultar se emplearía el API JDBC, la implementación de los paquetes se soporta sobre el JDK 6. Estos paquetes se implementan con el lenguaje de programación Java.

Nivel de Instrumentación: en este nivel se ubican los paquetes correspondientes tanto a los MBean como a las clases que brindan las funcionalidades de gestión. Estos paquetes se implementan con el lenguaje de programación Java y se soportan sobre el JDK 6. Estas clases son entidades software que permiten realizar la gestión. La implementación de estos recursos significa crear objetos Java.

2.5.3.2 Diagrama de Despliegue

En la figura 23 se muestra el diagrama de despliegue que expone la configuración de la implementación de referencia para su desarrollo en un ambiente real, presentando la

disposición física de los distintos equipos que intervienen en la conformación del sistema y la topología hardware de la implementación.

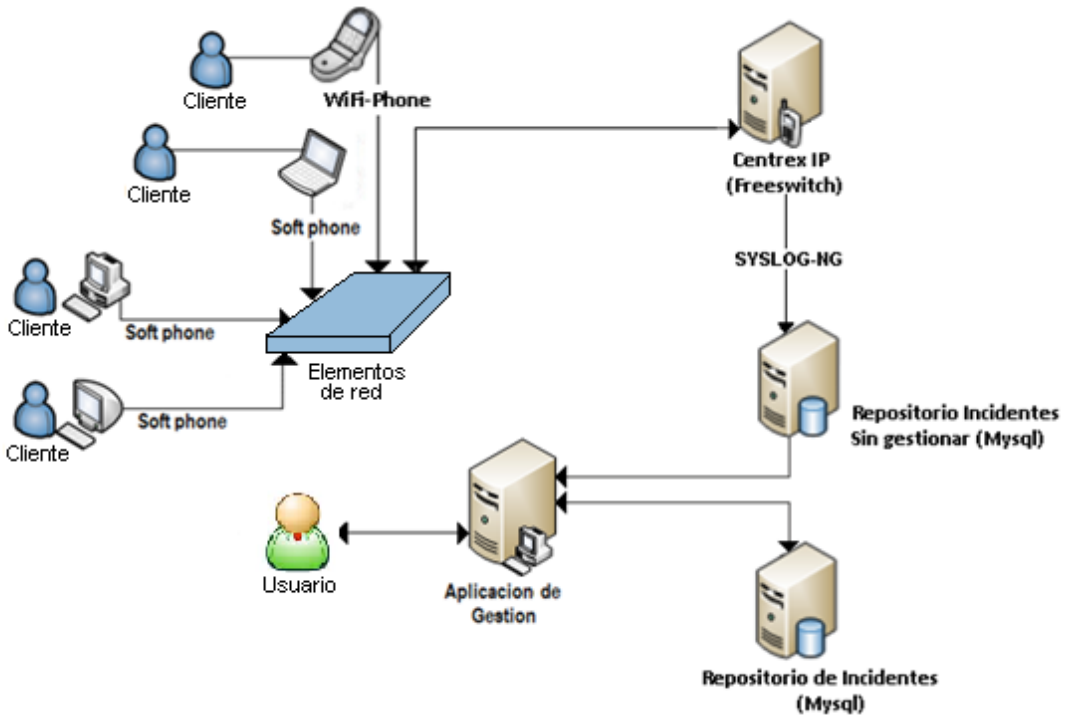


Figura 23. Diagrama de Despliegue

En el Diagrama de Despliegue se observan los clientes que poseen equipos que tienen instalado un SoftPhone, estos equipos pueden ser computadores de escritorio, portátiles, teléfonos IP e incluso un Wifi-Phone, estos SoftPhone permiten establecer llamadas a través de la red, esto es posible ya que se cuenta con una Centrex IP que en este caso es Freeswitch y además de los elementos de red. Esta central ofrece los servicios de telefonía IP los cuales son monitorizados por la herramienta llamada Syslog-ng, a su vez extrae incidentes en forma de Logs que son almacenados en una base de datos Mysql, para luego ser tomados por la Aplicación de Gestión donde se implementan las funcionalidades recomendadas por ITIL-LITE, finalmente se usa una base de datos para almacenar los incidentes ya gestionados y procesados.

Los elementos que componen el Diagrama de Despliegue son:

- **Softphone:** es una aplicación que permite a un computador de escritorio, portátil, o un teléfono móvil, funcionar como un teléfono convencional a través de voz sobre IP, con funcionalidades como por ejemplo, retener llamada, remarcado, directorio telefónico, identificador de llamada, etc. En este caso se instala para esta red el SoftPhone llamado Zoiper, es un software versátil y liviano que tiene las funcionalidades requeridas en el proyecto, es de uso libre.

- **Elementos de red:** lo componen dispositivos físicos de red como son switches, routers, cableado estructurado, conectores, y demás componentes necesarios

para hacer posible el tránsito de datos y la conexión de los diferentes dispositivos referentes al sistema de comunicación y gestión.

- **Centrex IP:** es una central telefónica encargada de proveer la comunicación de VoIP, para su implementación se utiliza la central Freeswitch, que permite el procesamiento de llamadas, y otros servicios. Entre sus funciones se encuentra el registro de extensiones, establecimiento de llamada, la detección de tonos para dar soporte al servicio de transferencia de llamada, realizar la captura de llamada y llevar a cabo la grabación de estas, también da soporte a servicios como correo de voz, IVR, entre otros.

- **Syslog-ng:** es una aplicación de software libre, que en este caso en particular se usa para extraer mensajes de registro en una red informática IP, un mensaje de registro puede contener cualquier información útil para realizar gestión, Syslog también es conocido como un protocolo de red. Este es compatible con UDP y TCP, gracias a sus archivos de configuración se puede aplicar filtros dependiendo de las necesidades del sistema, estos archivos permiten seleccionar las fuentes de las cuales se requiere extraer los log y también se puede establecer el destino en el cual serán almacenados los mensajes de registro, en este caso el destino que se utiliza es una Base de Datos Mysql, la cual tendrá los log de Freeswitch. Todo esto con el objeto de tener la información de manera organizada para una óptima gestión.

- **Repositorio de Incidentes sin Gestionar:** este elemento almacena la información extraída por Syslog-ng en una Base de Datos de Mysql, este repositorio está constantemente actualizándose conforme se esté generando nueva información en el sistema. En este Repositorio quedan almacenados los log que se presentan en la Centrex, estos se almacenan de manera desordenada. Esta información puede ser consultada desde su interfaz de gestión, en el momento que se requiera.

- **Aplicación de Gestión:** contiene las funcionalidades basadas en ITIL-LITE para realizar la gestión de los servicios del Centrex IP, continuamente se encuentra consultando la base de datos (Mysql) a la espera de nuevos incidentes que se presenten en el Centrex IP, con el propósito de ofrecer una solución pronta y eficaz. Parte de la estructura de esta aplicación de gestión se basa en la tecnología de Extensiones de Gestión de Java (JMX).

Protocolos de comunicación y conexión. Los protocolos son el conjunto de reglas que especifican el intercambio de mensajes durante la comunicación entre las entidades que forman parte de una red. Los utilizados por la plataforma son los siguientes:

- **HTTP** (HyperText Transfer Protocol): el Protocolo de transferencia de hipertexto define la sintaxis y la semántica que utilizan los elementos software de la arquitectura web (clientes, servidores) para comunicarse. Es un protocolo orientado a transacciones y sigue el esquema petición-respuesta entre un cliente y un servidor. Al cliente que efectúa la petición (un navegador web) se

lo conoce como agente del usuario. A la información transmitida se la llama recurso y se la identifica mediante un URL. Es el protocolo de comunicación definido para reportar incidentes a la herramienta de monitoreo.

- **JDBC** (Java Database Connectivity): esta API lleva a cabo las consultas sobre bases de datos utilizando el lenguaje de programación Java, gracias a la máquina virtual de java estas consultas se pueden realizar independientemente del sistema operativo donde se ejecute o de la base de datos a la cual se accede. Este protocolo es empleado para consultar el repositorio de incidentes.

Capítulo 3. Construcción de un Prototipo Basado en el Diseño Propuesto.

Para la construcción del prototipo se determina que herramientas software existentes son las más convenientes en su desarrollo, se da una descripción de su funcionamiento, su configuración, y finalmente se describe la aplicación del Insident Service Desk, utilizando algunos diagramas UML.

3.1 Herramientas Software

3.1.1 Freeswitch 1.0.6

Freeswitch es una plataforma de comunicaciones de software libre es decir código abierto para la creación de productos de voz, mensajería instantánea y video. El cual está disponible bajo licencia pública de Mozilla. La biblioteca principal, libfreeswitch, puede ser embebida en programas externos así como en aplicaciones independientes. Según el desarrollador principal, Anthony Minessale (pertenecía al grupo de desarrolladores de Asterisk), Freeswitch es un soft-switch construido sobre una máquina de estado sólida que incluye estabilidad, escalabilidad y abstracción.

Freeswitch incluye varios módulos que proveen aplicaciones por defecto como conferencias, XML-RPC para controlar llamadas en tiempo real, respuesta de Voz Interactiva (IVR), Conversor texto-voz, Reconocimiento Automático de Voz (CTV/RAH), Red Telefónica Conmutada (RTC), la capacidad de interconexión con circuitos analógicos y digitales, protocolos Voz sobre IP como SIP, SCCP, H.323, XMPP(Google Talk), entre otros.

Existe en el mercado otros soft-switch similares los cuales se descartaron por las siguientes razones:

- **Asterisk:** el core no es independiente, hay código necesario disperso en módulos, se puede probar haciendo una llamada sin tener el módulo res_features cargado, entonces la llamada no tendrá éxito. Asterisk genera un solo thread por llamada, por lo que las transferencias son un problema, ya que ocurre un fenómeno llamado 'channel masquerading', en el que los datos de un canal tienen que pasar de la memoria de un thread a otro, sin que ocurra nada. Además haciendo una comparación con el mismo hardware, Freeswitch puede manejar 1000 llamadas concurrentes sin problemas, mientras que en Asterisk estos empiezan a aparecer a partir de las 250 llamadas [34].

- **Bayonne:** cuenta ya con varios años en el mercado sin embargo no tiene tanta aceptación. Su ingreso en el proyecto GNU (proyecto iniciado por Richard Stallman con el objetivo de crear un sistema operativo completamente libre), es reciente. Su autor, David Sugar, lo llama la "navaja Suiza" de los servidores de telefonía. Promete ser un software muy completo en el futuro y soportar todas las funcionalidades requeridas por el usuario. Lastimosamente aún se

encuentra en desarrollo y debido a que su difusión no es amplia, su desarrollo es más lento de lo esperado.

- **OpenPBX:** ha sido desarrollado y diseñado por la compañía Nulit. OpenPBX contiene todas las características encontradas comúnmente en soluciones comerciales al costo de una computadora portátil, pero funciona únicamente en sistema operativo Linux.

Tabla 3. Comparativa entre Centrales IP [35]

	Asterisk	FreeSWITCH	Bayonne	OpenPBX
Protocolo SIP	si	si	si	si
Protocolo H323	si	si	licencia	licencia
Protocolo IAX2	si	si	no	si
Protocolo MCGP	si	si	no	no
Procesador	PII 300Mhz	PIII 1 Ghz	PIII 1.5 Ghz	PIII 2 Ghz
Memoria RAM	128 Mb	256 Mb	256 Mb	512 Mb
Disco Duro	4 Gb	8 Gb	4 Gb	4 Gb
Correo de voz	si	si	si	si
Ruteo	si	si	si	si
Desvío de llamadas	si	si	no	no
Conferencias	si	si	no	si
IVR	si	si	no	si
Texto a voz	si (nuevo)	si (nuevo)	no	No
Control de llamadas	si	si	no	Si
Interfaces	E1, T1, BRI, FXS, FXO	E1, T1, BRI, FXS, FXO	FXS, FXO	E1, T1, BRI, FXS, FXO
Escalabilidad	2000 extensiones	2000 extensiones	500 extensiones	500 exten.
Documentación	Muy amplia	Amplia	Poca	Regular
Soporte	Abundante	Poco	Muy poco	Regular

3.1.1.1 Instalación

Previamente a la instalación de Freeswitch 1.0.6 se instalan los siguientes paquetes para su correcta compilación:

1. Entrando como administrador en Ubuntu 10.04 instalamos el paquete buil-essential:
   apt-get update
   apt-get install build-essential

2. Se sigue con el motor de bases de datos y la aplicación para php. Cada uno de los siguientes comandos instala la última versión de mysql y la versión 5 de php. Para mysql se asigna un password (root) para el usuario.

   apt-get install mysql-client

apt-get install mysql-server
apt-get install libmysqlclient15-dev
apt-get install php5 php5-dev

3. Se prosiguecon el ODBC (ODBC: Open DataBase Connectivity) para el motor de base de datos:

   apt-get install unixodbc
   apt-get install unixodbc-bin
   apt-get install unixodbc-dev

   wget
   http://co.archive.ubuntu.com/ubuntu/pool/universe/u/unixodbc/unixodbc-bin_2.2.11-16ubuntu1_i386.deb

4. Luego se instala el servidor de correo electrónico que soporta los protocolos SMTP (SMTP: Simple Mail Transfer Protocol), POP (POP: Post Office Protocol), IMAP (IMAP: Internet Message Access Protocol).

   apt-get install sendmail dovecot-common

5. Se instala el CURL, el cual es una herramienta para usar en un intérprete de comandos para transferir archivos con sintaxis URL, el principal propósito y uso para CURL es automatizar transferencias de archivos o secuencias de operaciones no supervisadas. Es una buena herramienta para simular las acciones de un usuario en un navegador web. La librería libcurl se usa para proveer capacidades de transferencia de URL a numerosas aplicaciones, tanto libres y open source como privativas.

   apt-get install libcurl3-dev curl libcurl3

6. Para la compilación del módulo jabber se sigue con iksemel y gnutls.

   apt-get install libgnutls26 libgnutls-dev

   cd /usr/src

   wget http://iksemel.googlecode.com/files/iksemel-1.3.tar.gz

   tar -xf iksemel-1.3.tar.gz
   cd iksemel-1.3
   ./configure --prefix=/usr
   make
   make check
   make install

7. Para el modulo FAX se instala spandsp.

cd /usr/src

apt-get install libtiff4 libtiff4-dev libncurses5-dev
wget http://www.soft-switch.org/downloads/spandsp/spandsp-0.0.6pre17.tgz

tar -xf spandsp-0.0.6pre17.tgz
cd spandsp-0.0.6
./configure --prefix=/usr
make
make install

8. Se actualiza las librerías de las instalaciones.

   ldconfig -v

9. Ahora, ya el sistema está listo para instalar Freeswitch su última versión, la 1.0.6

   cd /usr/src

   wget http://files.freeswitch.org/freeswitch-1.0.6.tar.gz
   tar -xf freeswitch-1.0.6.tar.gz
   cd freeswitch-1.0.6

10. Para el trabajo a realizar se activan algunos módulos adicionales descomentariando algunas líneas del archivo modules.conf

    nano modules.conf

    Se quita el # a las líneas:

    endpoints/mod_dingaling
    languages/mod_spidermonkey_odbc
    xml_int/mod_xml_curl

    Los cambios se guardan con Ctrl+O, se acepta con Enter, y para salir Ctrl-X.

11. Una vez modificado esto, se empieza a compilar.

    ./configure --enable-core-odbc-support

    make
    make install

12. Se instala las voces y los archivos para la música en espera.

make sounds-install
make moh-install

13. El programa queda listo para ser arrancado, se hace mediante los siguientes comandos.

cd /usr/local/freeswitch/bin
./freeswitch

Cuando todo termina bien, la consola debe mostrar la siguiente imagen:

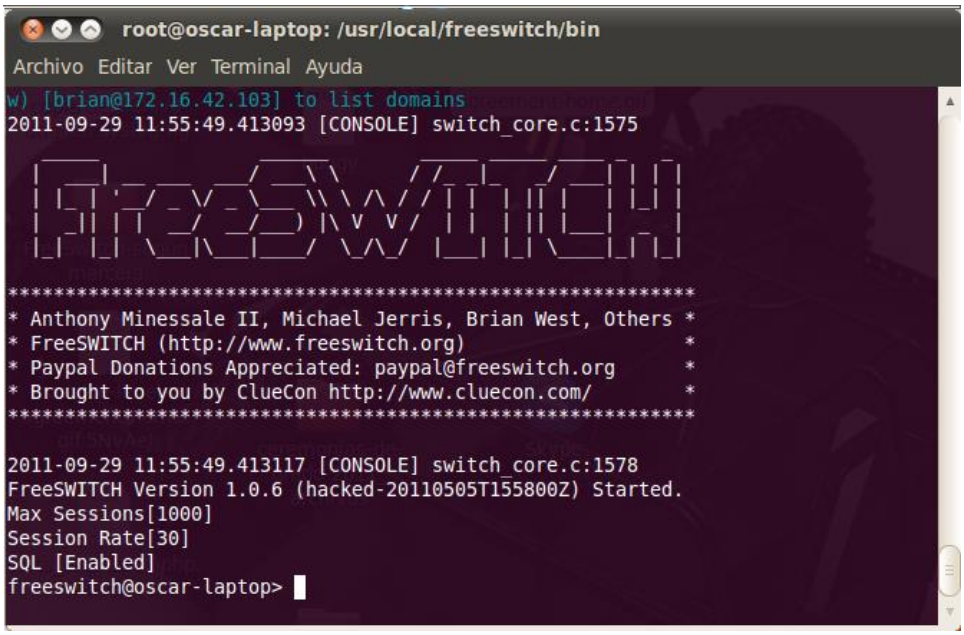


Figura 24. Central Freeswitch Corriendo.

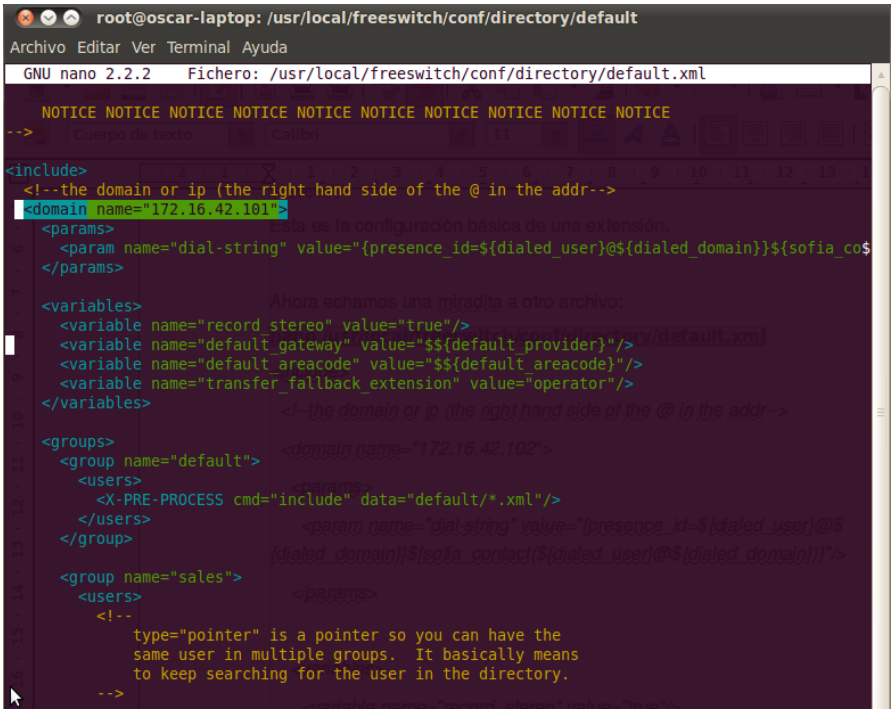
Para abrir la aplicación nuevamente:

cd /usr/local/freeswitch/bin
./freeswitch

3.1.1.2 Configuración

Una vez instalada la Freeswitch se debe configurar las extensiones para poder establecer las llamadas, pero antes de esto se ingresa al archivo default.xml para editar el nombre de dominio que en este caso es la IP del servidor, ver figura 25. Con el siguiente comando se abre el archivo con su ubicación haciendo uso del editor nano.

nano /usr/local/freeswitch/conf/directory/default.xml




```
<include>
  <!--the domain or ip (the right hand side of the @ in the addr-->
  <domain name="172.16.42.101">
    <params>
      <param name="dial-string" value="{presence_id=${dialed_user}@${dialed_domain}}${sofia_co$
    </params>

    <variables>
      <variable name="record_stereo" value="true"/>
      <variable name="default_gateway" value="$${default_provider}"/>
      <variable name="default_areacode" value="$${default_areacode}"/>
      <variable name="transfer_fallback_extension" value="operator"/>
    </variables>

    <groups>
      <group name="default">
        <users>
          <X-PRE-PROCESS cmd="include" data="default/*.xml"/>
        </users>
      </group>

      <group name="sales">
        <users>
          <!--
              type="pointer" is a pointer so you can have the
              same user in multiple groups.  It basically means
              to keep searching for the user in the directory.
          -->
```

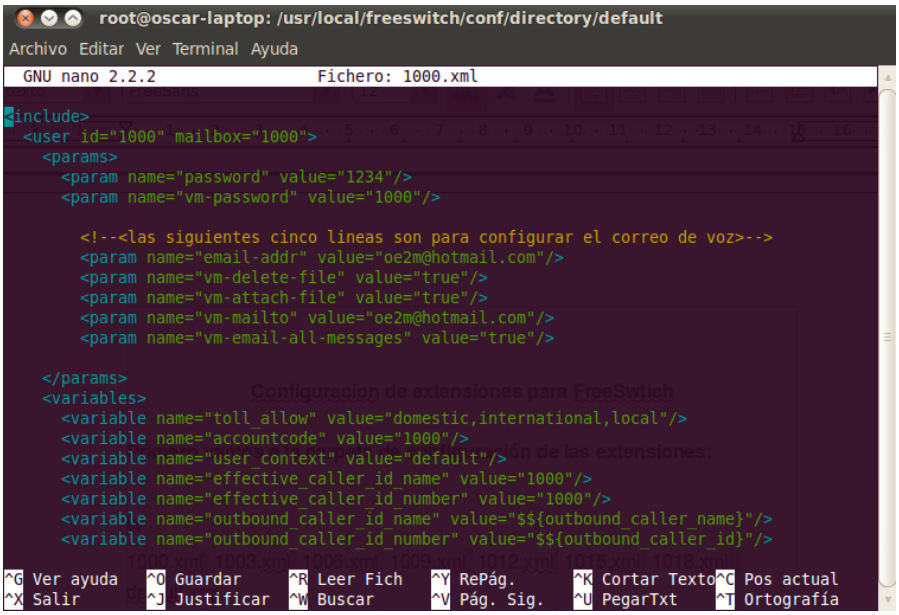
Figura 25. Configuración del Nombre de Dominio

- **Configuración de Extensiones**

Para configurar por ejemplo la extensión 1000, se edita el archivo 1000.xml ubicado en la siguiente ruta:

nano /usr/local/freeswitch/conf/directory/default/1000.xml

El archivo se edita como se ve en la siguiente figura:




```
<include>
  <user id="1000" mailbox="1000">
    <params>
      <param name="password" value="1234"/>
      <param name="vm-password" value="1000"/>

        <!--<las siguientes cinco lineas son para configurar el correo de voz>-->
        <param name="email-addr" value="oe2m@hotmail.com"/>
        <param name="vm-delete-file" value="true"/>
        <param name="vm-attach-file" value="true"/>
        <param name="vm-mailto" value="oe2m@hotmail.com"/>
        <param name="vm-email-all-messages" value="true"/>

    </params>
    <variables>
      <variable name="toll_allow" value="domestic,international,local"/>
      <variable name="accountcode" value="1000"/>
      <variable name="user_context" value="default"/>
      <variable name="effective_caller_id_name" value="1000"/>
      <variable name="effective_caller_id_number" value="1000"/>
      <variable name="outbound_caller_id_name" value="$${outbound_caller_name}"/>
      <variable name="outbound_caller_id_number" value="$${outbound_caller_id}"/>
```

Figura 26. Configuración Extensión 1000

- **Servicios de VoIP Seleccionados**

Freeswitch ofrece varios servicios de VoIP de los cuales se escogió el establecimiento de llamada, el cual funciona al crearse las extensiones, también se escogió el servicio de transferencia de llamadas. Los anteriores servicios se escogen con el fin de generar incidentes de manera controlada en la comunicación, con el propósito de hacer pruebas en el sistema de gestión.

Para realizar el servicio de transferencia de llamada se prosigue por ejemplo de la siguiente forma:

Un usuario con extensión número 1002 marca a un usuario con extensión 1000, en la extensión 1000 se escucha el timbre de su softphone y el usuario contesta, en este momento se establece la llamada, para realizar la transferencia de llamada a una tercera extensión (1001), el usuario de la extensión 1000 marca en su softphome *1, deja un espacio y marca 1001, esto significa que la extensión 1000 transfiere la llamada a la extensión 1001, en este momento timbra en la 1001, esta contesta y ahora se establece una nueva llamada entre los usuarios de las extensiones 1002 y 1001.

3.1.2 Zoiper

Zoiper 2.28 es un cliente para VozIP (softphone) de alta calidad, que permite conectar con la central Freeswitch y es una buena opción de comunicación telefónica por internet. Entre sus características se puede comentar: soporte de protocolos SIP + IAX / IAX 2, soporte STUN, servidor STUN por cuenta, soporte TCP con SIP, soporte TLS con SIP, disponibles codecs: GSM, ulaw, alaw, speex, ilbc, G.729 (solo en versión comercial Zoiper BIZ), soporte multilenguaje, servicio de conferencias nativo, API, auto respuestas, integración Outlook, entre otras funciones también multiplataforma, Linux, Windows y Mac OS X.

3.1.2.1 Instalación

Su proceso de instalación es el siguiente:

1. Se descarga el instalador tanto para Windows como para Linux desde el sitio oficial:
   http://www.zoiper.com/

2. Se Copia la descarga a la carpeta respectiva, desde descargas:
   cd Descargas
   mv zoiper218-linux.tar.gz /usr/src

3. Se habilita permisos y se descomprime
   cd usr/src/
   chmod +x zoiper218-linux.tar.gz
   tar -zxvf zoiper218-linux.tar.gz

4. Con los comandos anteriores se descomprime un archivo zoiper que se ejecuta con la siguiente línea de comando:
   ./zoiper

Este ejecutable despliega la siguiente interfaz:

Figura 27. Softphone Zoiper

3.1.2.2 Configuración

Para la configuración del softphone Zoiper se debe tener en cuenta el número de la extensión, la contraseña, nombre de identificación del usuario y el dominio, éste último debe ser el mismo que se configura en archivo default.xml de Freeswitch, para introducir estos parámetros se hace clic en herramientas desplegándose la interfaz que permite dicha configuración, dando como resultado el registro de una extensión al softswitch utilizado.

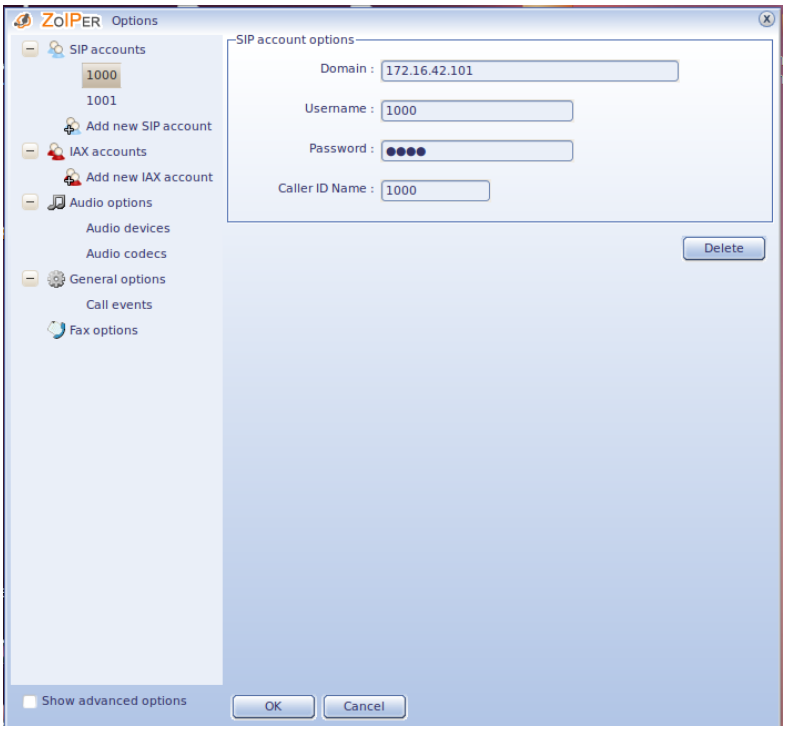


Figura 28. Configuración de una Extensión en Zoiper

3.1.3 Syslog-ng

Esta herramienta permite monitorizar y extraer los incidentes de Freeswitch en forma de Log desde su archivo llamado freeswitch.log ubicado en la siguiente ruta /usr/local/freeswitch/log/freeswitch.log, luego dicha herramienta envía los Log a una base de datos de Mysql llamada también syslog con una tabla nombrada log donde se encuentran los incidentes.

3.1.3.1 Instalación

Para el correcto y completo funcionamiento de esta herramienta se instala un software adicional como lo es Apache, Mysql y PHP, para hacer efectivo esto se digita el siguiente comando desde la terminal:

1. apt-get install apache2 mysql-server php5 php5-mysql

2. Después de esto se instala syslog-ng

   apt-get install syslog-ng

3. También se necesita instalar php-syslog (en este caso fue instalada la versión 2.8) el cual se puede descargar desde la dirección

   http://sourceforge.net/projects/php-syslog-ng/files/

4. Cuando se logra la descarga se descomprime este paquete en
   tar -zxvf phpsyslogng-2.8.tar.gz -C /var/www

   y se mueve de directorio

   cd /var/www && mv phpsyslogng-2.8 phpsyslog

3.1.3.2 Configuración

Ya estando instaladas estas herramientas se configura Mysql para que trabaje con Syslog editando el archivo dbsetup.sql

1. gedit dbsetup.sql
   Ahora se buscan estas lineas en el fichero:

   # create users
   INSERT INTO user (Host, User, Password) VALUES (‘localhost’,'sysloguser’, password(‘PW_HERE‘));
   INSERT INTO db (Host, Db, User) VALUES (‘localhost’,'syslog’,'sysloguser’);

   INSERT INTO user (Host, User, Password) VALUES (‘localhost’,'syslogfeeder’, password(‘PW_HERE‘));
   INSERT INTO db (Host, Db, User) VALUES (‘localhost’,'syslog’,'syslogfeeder’);

INSERT INTO user (Host, User, Password) VALUES ('localhost','syslogadmin',password('PW_HERE'));
INSERT INTO db (Host, Db, User) VALUES ('localhost','syslog','syslogadmin');
COMMIT;
FLUSH PRIVILEGES;

Se edita la parte PW_HERE por una nueva contraseña para la seguridad de la herramienta de monitoreo.
Se guarda los cambios y se cierra el archivo, después se ejecuta la siguiente línea de comandos en el terminal: mysql -u root -p < dbsetup.sql, y se digita la contraseña de root de Mysql

2. Para configurar la tubería que une Syslog con Mysql se edita el siguiente archivo con: gedit syslog2mysql.sh

   En el fichero se busca esta línea:

   mysql -u syslogfeeder –password=PW_HERE syslog < /var/log/mysql.pipe >/dev/null

   Y se edita la parte de PW_HERE. Después de guardar los cambios se ejecuta el archivo para correr la tubería:

   ./syslog2mysql.sh

3. Ahora se prosigue con la configuración de apache, que en este caso es el servidor web, editando el archivo apache2.conf por medio de:

   gedit /etc/apache2/apache2.conf
   se agrega al final del mismo la siguiente línea:
   AddType application/x-httpd-php .html .php

4. En este punto se configura PHP-SYSLOG-NG modificando su contraseña por medio del editor gedit:

   gedit /var/www/phpsyslog/config/config.php

   Para que todo marche bien se reinicia apache.
   # /etc/init.d/apache2 restart

5. Después de esto se abre el directorio/var/www/phpsyslogng-2.8/scripts y se configura el archivo para que Syslog-ng pueda escribir datos en Mysql.

   Se ejecuta como root en la consola

   cd /var/www/phpsyslogng-2.8/scripts

   cat syslog.conf >> /etc/syslog-ng/syslog-ng.conf

   utilizando el editor gedit se abre el archivo /etc/default/syslog-ng

gedit /etc/default/syslog-ng

y se procede a descomentarear esta linea (le quitamos la almohadilla)

CONSOLE_LOG_LEVEL=1

6. Por último se configura el archivo syslog-ng.conf ubicado en /etc/syslog ng/syslog-ng.conf.

   La línea de comando #use_dns(no); se la remplaza por #use_dns(yes); y debajo de esta se inserta la línea dns_cache(yes);

Después de todas estas configuraciones el sistema de monitoreo que da listo para ser usado, entonces en el explorador Mozila se digita http://localhost/phpsyslog

3.1.4 Netbeans 7.0

Este IDE (IDE: Integrated Development Environment) está hecho principalmente para el lenguaje de programación Java, es un producto libre y gratuito. Esta herramienta permite que las aplicaciones sean desarrolladas a partir de un conjunto de componentes de software llamados módulos, los cuales se pueden ir agregando para extender el IDE, dentro de estos módulos se encuentra el denominado JMX que permite la construcción del diseño de la solución de gestión, también se encuentra el modulo Swing que permite el desarrollo de interfaces graficas de usuarios. Para este proyecto se utiliza la última versión que es la 7.0 lanzada en Abril de 2011.

3.2 Implementación del Incident Service Desk

Hasta este punto se ha instalado la central Freeswitch, se han configurado sus servicios, y también se ha instalado y configurado la herramienta de monitoreo de incidentes llamada Syslog-ng. Lo que resta por hacer es exponer la implementación de la herramienta que realiza Gestión de incidentes, para lo anterior se usan algunos de los diagramas que posee UML, dentro de estos están los diagramas de tablas y diagramas de secuencias. Antes de entrar a describir los diagramas de la implementacióndel Incident Service Desk se muestra el diagrama de flujo de actividades de procesos (figura 29), que han sido descritos a través del documento.

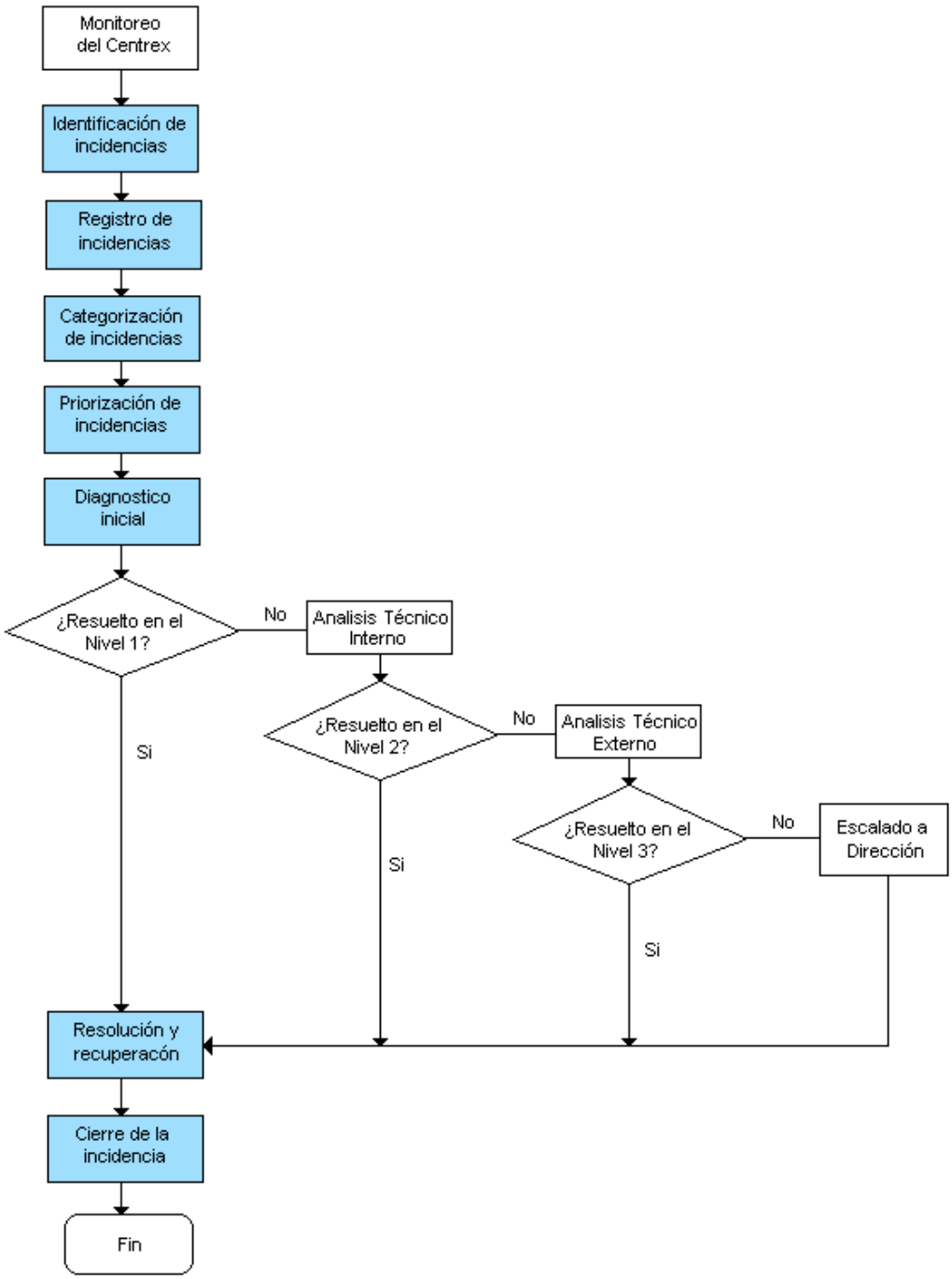


Figura 29. Procesos en la Gestión de Incidentes implementado

3.2.1 Descripción de Actores

Los actores que intervienen en el manejo del sistema para realizar Gestión de Incidentes son:

- **Administrador:** persona que accede al sistema (figura 30) y tiene las tareas de ingresar, modificar, consultar y eliminar los requerimientos que se presenten, también verifica el desempeño del personal que trabaja en el Service Desk.

Otras de las funciones son ingresar, modificar, consultar y eliminar técnicos, proveer recursos adicionales, entre otros.

En el siguiente diagrama de secuencia se describe las acciones que se realizan a la hora de autenticar los datos de usuario ante el sistema, si la identificación de usuario y la contraseña están contenidas en la base de datos UsuarioBD entonces el sistema retorna verdadero, lo que indica el éxito del ingreso a la aplicación.

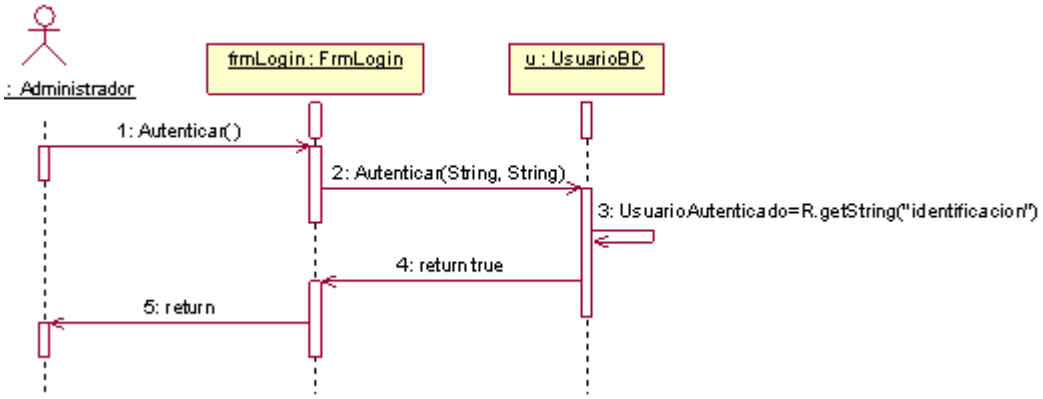


Figura 30. Secuencia Iniciar Sesión

- **Técnico:** es el encargado de atender personalmente los incidentes detectados, si esta persona no logra restablecer el servicio, hace uso del escalado funcional. El técnico tiene las funcionalidades de: iniciar sesión (semejante a iniciar sesión como administrador figura 30), consultar incidentes disponibles (figura 31), seleccionar incidente a resolver, resolver incidente (figura 32). A continuación se detallan las principales funcionalidades por medio de diagramas de secuencia.

  En la figura 31 se expone el diagrama de secuencia consultar incidentes disponibles, que inicia cuando el técnico invoca al método ConsultarIncidentesDisponibles() después de hacer clic en el respectivo botón de su interfaz. El componente FrmTablaIncidentes recibe la invocación que carga las categorías haciéndolas visibles, en este componente también se invoca el método que carga los filtros de búsqueda de incidentes. Luego se llama al método que extrae los Log desde una base de datos para ser procesados y convertidos en incidentes. También se cuenta con un método de monitoreo de incidentes recientes, el cual activa los temporizadores que inician este proceso y el proceso que actualiza la base de datos de incidentes.

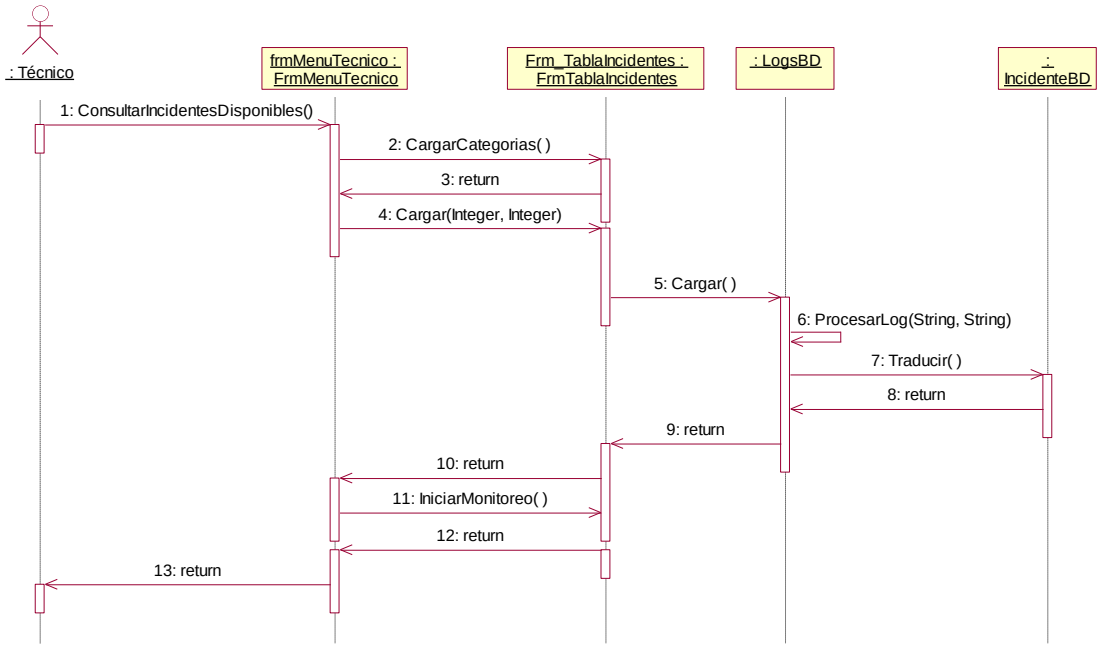


Figura 31. Consultar Incidentes Disponibles

A continuación se describe la secuencia de acciones que conlleva la solución de un incidente, ésta da inicio cuando el técnico escoge un incidente y carga su plantilla o ticket, el cual muestra su interfaz para editar el campo de solución, como también editar escala entre otros, luego de modificar estos campos se procede a cerrar el incidente.

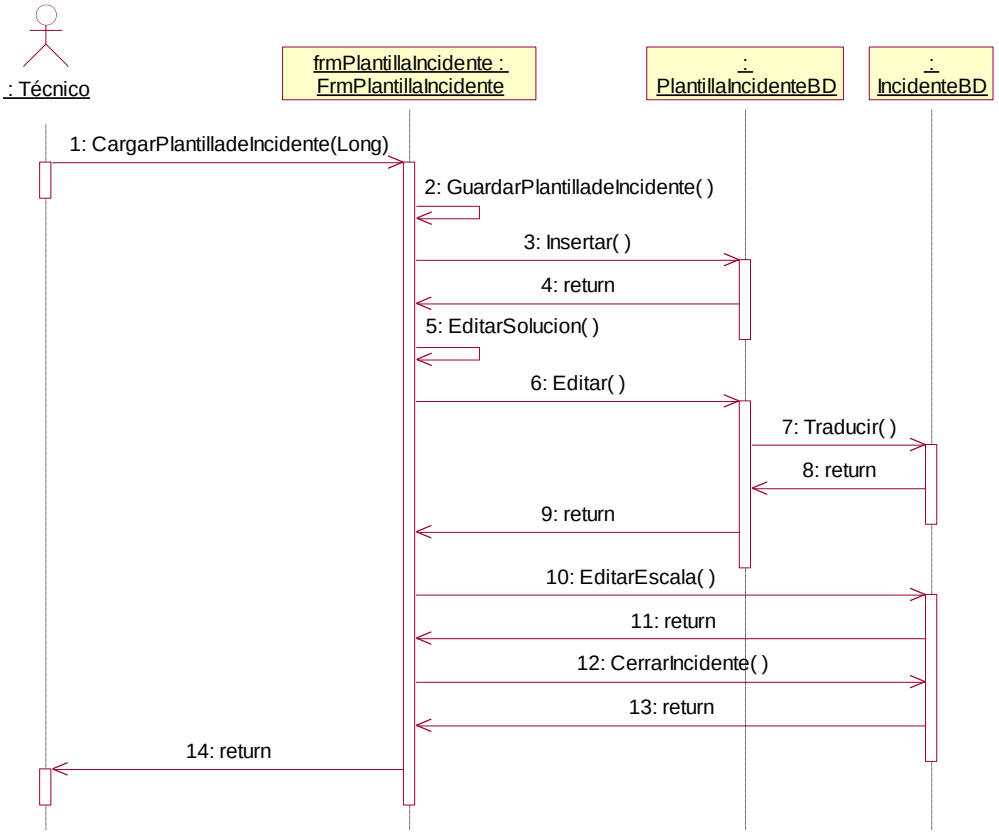


Figura 32. Secuencia Resolver Incidente

3.2.2 Diseño de la Base de Datos

A continuación se muestra la base de datos llamada GDI (GDI: Gestión de Incidentes), la cual fue implementada con Mysql, las razones por las que se usa es que es software libre, se encuentra en el interior del IDE de Netbeans 7.0 en un módulo de servicios o prestaciones. También se detallan las tablas con las que cuenta dicha base de datos y se muestra el siguiente diagrama (Figura 33).

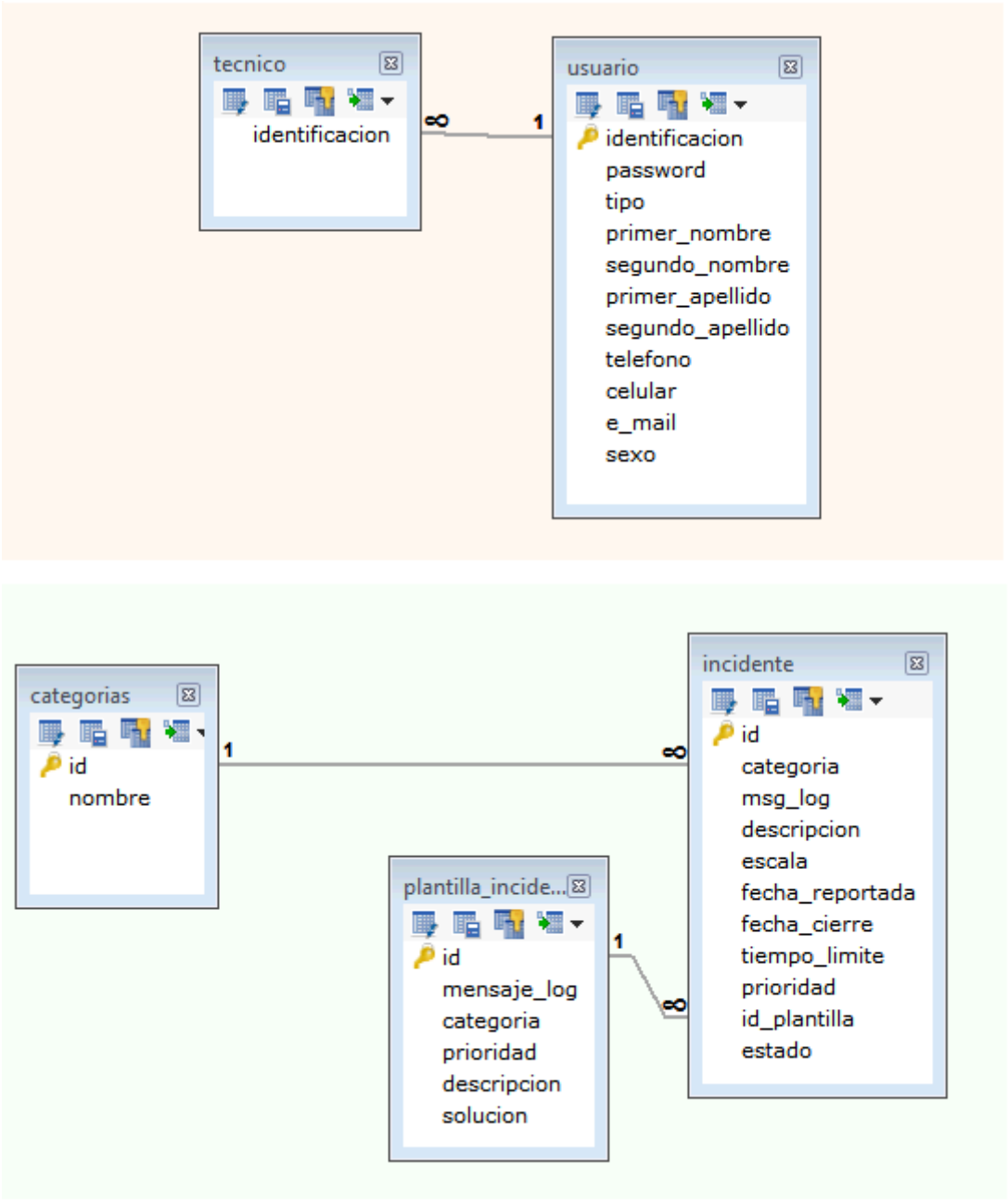


Figura 33. Diagrama de Tablas de la Base de Datos GDI

Se procede a describir cada una de las tablas que conforman la base de datos GDI:

Tabla usuario: esta tabla se crea con el fin de generar un registro de los usuarios que interactúan con la herramienta de gestión Incident Service Desk, estos usuarios

pueden ser tanto un técnico como un administrador. En esta tabla se encuentra un campo llamado Tipo, el cual es relevante a la hora de identificar si el usuario es un técnico o un administrador, a continuación se explican los campos que conforma la Tabla Usuario:

- **identificación:** este campo es la llave primaria de esta tabla, ya que la identificación es única para un técnico o un administrador, también se utiliza como nombre de usuario para ingresar al sistema.

- **password:** en esta columna se almacena la contraseña del usuario que le sirve como segundo dato de identificación para el ingreso al Incident Service Desk, aumentando la seguridad del sistema.

- **tipo:** cuando este campo está en valor 1 el usuario es un administrador, y si el valor es 2 se trata de un técnico.

- **primer_nombre:** es el primer nombre del técnico o administrador, este campo al igual que anteriormente mencionados es carácter obligatorio.

- **segundo_nombre:** se trata del segundo nombre del usuario, este campo no es obligatorio.

- **primer_apellido:** es el primer apellido del usuario ya sea técnico o administrador, este campo es necesario para el sistema.

- **segundo_apellido:** como su nombre lo indica esta casilla almacena el otro apellido del usuario, y es un campo no obligatorio.

- **teléfono:** se trata del teléfono fijo, este campo se utiliza con el fin de tener datos para contactar al usuario.

- **celular:** al igual que el anterior es un dato para contactar al usuario en su teléfono móvil.

- **e-mail:** dato de contacto complementario a la demás información personal.

- **sexo:** permite identificar si el usuario se trata de una mujer o un hombre.

Tabla técnico: esta tabla se crea con el fin de lograr una gestión práctica de los datos de los técnicos, por medio de una interfaz de usuario se puede actualizar, agregar, eliminar y editar información de dichos técnicos. Aquí se encuentra un único campo llamado identificación, ya que los demás campos los abstrae de la tabla usuario.

- **identificación:** campo único que identifica al técnico.

Tabla incidente: contiene la información necesaria para realizar el proceso de registro de un incidente, permite asignarle una prioridad y darle una categoría al mismo, entre otras funcionalidades que son necesarias para la gestión de incidentes. Las columnas que pertenecen a esta tabla son las siguientes:

- **id:** número único de identificación del incidente, este valor se va incrementando a medida que llegan nuevos incidentes.
- **categoría:** este es un campo que contiene un número que identifica la categoría del incidente.
- **msg_log:** en esta columna se almacena el mensaje log de cada incidente, los cuales son extraídos por la herramienta de monitoreo.
- **descripción:** en este campo se escribe una interpretación del mensaje log, para mejor comprensión del técnico.
- **escala:** aquí se almacena un número entre el 1 y el 4 que representa en nivel de escalado del incidente.
- **fecha_reportada:** campo en el cual se guarda la fecha en la cual se ha detectado el incidente.
- **fecha_cierre:** casilla que sirve para almacenar la fecha en la cual el incidente ha sido solucionado.
- **tiempo_limite:** es el número de horas que el técnico dispone para solucionar el incidente, este valor se asigna dependiendo la prioridad.
- **prioridad:** lugar en el que se almacena la palabra que describe la prioridad del incidente.
- **id_plantilla:** como su nombre lo indica es el campo donde está el número que identifica la plantilla que tiene asociado el incidente.
- **estado:** campo que guarda información que permite identificar si el incidente está en estado abierto o cerrado.

Tabla plantilla_incidente: esta plantilla representa al ticket de incidente, contiene información que permite hacer modificaciones de algunos de los campos del incidente, para que se pueda lograr un correcto cierre del mismo. Esta tabla cuenta con los siguientes campos:

- **id:** campo que identifica con un número a la plantilla, pueda darse el caso de que varios incidentes tengan asociada una plantilla.
- **mensaje_log:** columna que guarda parte del msg_log con el fin de convertirlo en mensaje estándar para identificar incidentes del mismo tipo, que pueden ocurrir en un futuro.
- **categoría:** campo en el que se almacena el número que representa la categoría, este dato se puede corroborar antes de cerrar el incidente.
- **prioridad:** al igual que el anterior es un campo modificable antes del cierre del incidente, y representa el grado de prioridad del mismo.

- **descripción:** en este campo se escribe una interpretación del mensaje log, para mejor comprensión del técnico.

- **solución:** en esta columna se registra los pasos que conllevan a la solución del incidente, este campo puede ser editado por el técnico.

Tabla categorías: fue creada con el fin de logra facilidad en la asignación de las categorías seleccionadas, también permite crear una nueva categoría en caso de que el personal de Service Desk lo considere necesario. Los campos que contiene esta tabla son:

- **id:** valor numérico que identifica la categoría, este va del 0 al 5.

- **nombre:** columna donde se almacena el nombre de la categoría, este nombre está asociado al campo identificación anteriormente mencionado, las categorías que se estimaron en este proyecto son categoría desconocida, software-freeswitch, software-syslog, software-zoiper.

Capítulo 4. Pruebas y Funcionamiento

En este capítulo se describe y se muestra la herramienta Incident Service Desk, la cual tiene implementadas las funcionalidades de ITIL-LITE expuestas en capítulos anteriores. Para probar el adecuado funcionamiento del sistema se presenta un grupo de imágenes donde se observa las distintas opciones que permite la herramienta.

4.1 Seguridad del Sistema

La seguridad del sistema permite que no ingrese personal ajeno al grupo del Service Desk, que puedan causar algún daño o inconsistencias a la información ahí contenida, y solo pueda ingresar el personal al cual se le haya asignado una identificación de usuario y su respectiva contraseña.

4.1.1 Intento Errado

Al dar inicio a la herramienta aparecerá la primera interfaz gráfica (figura 34), en la cual se ingresa la identificación del respectivo usuario (Administrador o Técnico) y su contraseña.

Figura 34. Interfaz de Inicio

Si al ingresar los datos requeridos estos son incorrectos el sistema bloqueará la entrada a quien lo esté intentando y se muestra el siguiente mensaje (figura 35).

Figura 35. Notificación de Usuario o Contraseña Incorrecta

4.1.2 Ingreso Exitoso a la Aplicación

La herramienta trae por defecto una identificación de usuario (1234) y una contraseña (1234) que permite el ingreso al sistema por primera vez como usuario administrador.

4.1.2.1 Ingreso al sistema como Administrador

El administrador podrá cambiar su contraseña en el campo de seguridad (figura 36), además de hacer uso de todas las funcionalidades asignadas al administrador las cuales abarcan: ingreso de un nuevo técnico, editar técnico, eliminar técnico (figura 37), entre otras.

Figura 36. Interfaz de Administrador

Figura 37. Ventana de Gestión de Técnico.

4.1.2.2 Ingreso al Sistema como Técnico

Desde la interfaz de inicio (figura 34) también se puede ingresar como técnico, ya que el sistema puede determinar comprobando los datos de usuario y contraseña de qué tipo de usuario se trata. Luego de verificar los datos del técnico la herramienta despliega la interfaz de técnico (figura 38) en la cual se encuentran las opciones de Incidentes Disponibles, Cambiar Contraseña y Volver.

Figura 38. Interfaz Menú Técnico.

- Cuando se escoge la opción Incidentes Disponibles, la herramienta despliega una interfaz (figura 41) que permite visualizar los incidentes detectados por el sistema, estos incidentes provienen tanto de Syslog, Zoiper como de Freeswitch, ya que previamente se realizó un filtro en el archivo de configuración llamado syslog-ng.conf de Syslog (herramienta de monitoreo de logs).

- Si el técnico desea cambiar su clave de ingreso lo puede hacer desde el botón Cambiar Contraseña, el cual le pedirá que ingrese la contraseña actual, también que digite la nueva clave y que finalmente la confirme, para que el sistema pueda comprobar los datos y realizar adecuadamente los cambios. Lo anterior se puede observar en la siguiente figura 39.

Figura 39. Cambio de Contraseña del Técnico.

- La interfaz de Menú Técnico cuenta con un botón Volver, éste permite desplazarse hacia la interfaz inicial (ver figura 34), con el propósito de poder ingresar con otro perfil de usuario.

4.2 Funcionalidades del Incident Service Desk

Del marco de gestión ITIL-LITE se extrajo las funcionalidades que se implementaron en el sistema, dentro de estas funcionalidades se encuentran la identificación del incidente, priorización, categorización, registro, diagnostica inicial, escalado, resolución y cierre de la incidencia.

4.2.1 Detección del Incidente

Para efectos de prueba del sistema se genera un incidente en el servicio de establecimiento de llamada, el incidente de ejemplo se produce al llamar a un usuario que aún no está registrado como extensión en los archivos de configuración de la Centrex IP. Como primera medida se detecta el log por medio de la herramienta de monitoreo Syslog como se puede apreciar en la siguiente figura 40. Aquí vale notar que los campos que utiliza esta herramienta son solamente: el mensaje log, fecha y hora, prioridad, host, y el campo seq (secuencia).

En cuanto al campo de prioridad que asigna Syslog, se analizó que no es acorde a la prioridad real que se quiere representar en el Incident Service Desk, esto es en cuanto

a la diferencia en las palabras utilizadas en la priorización y también en cuanto a la gravedad del incidente.

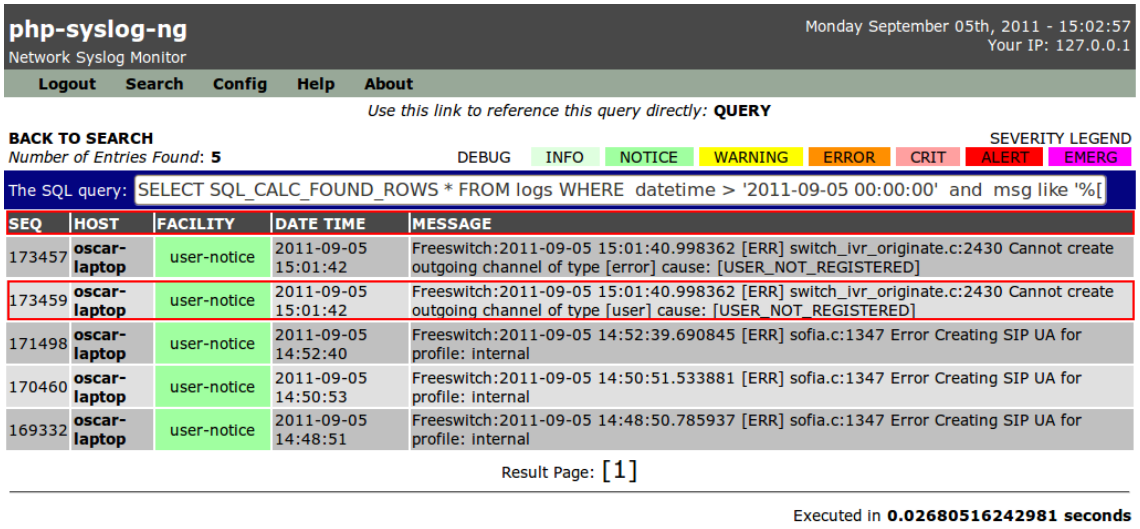


Figura 40. Detección del Log desde Syslog.

El log se encuentra almacenado en una base de datos llamada syslog, de la cual se extrae para ser almacenado como incidente en una nueva base de datos llamada gdi agregándole nuevos campos lo cuales son: ID, Fecha Reportada, Tiempo Límite, Categoría, Prioridad, Escala, Descripción, Mensaje Log, y Fecha de Cierre. Los campos mencionados permiten una eficiente gestión de incidentes. Finalmente el log, más los nuevos campos son presentados como un incidente en una interfaz gráfica como se puede ver en la figura 41.

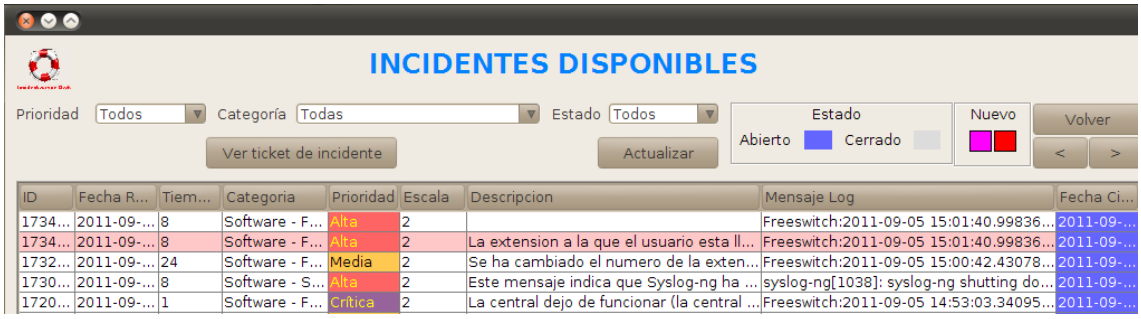


Figura 41. Detección del incidente desde el Incident Service Desk.

4.2.2 Identificación del Incidente

La identificación del incidente se lleva a cabo cuando el Incident Service Desk asigna una ID a cada incidente que lo hará único en todo el sistema, con el fin de facilitar su gestión (ver figura 42).

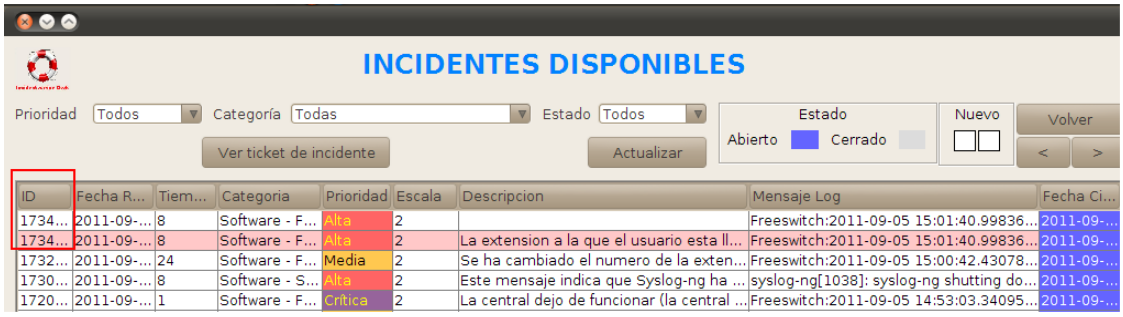


Figura 42. Identificación del Incidente.

4.2.3 Filtrado de incidentes

Además de las funcionalidades que recomienda ITIL, la herramienta Incident Service Desk cuenta con un sistema de búsqueda de incidentes por medio de una serie de filtros, los cuales son: filtrado por prioridad, por categoría, por estado, y la combinación de estos (ver figura 43).

- **Filtrado por Prioridad:** este filtro permite buscar los incidentes que se encuentran registrados en el sistema por medio de su prioridad (Critica, Alta, Media, Baja, Planeada o Todos), esto facilita al técnico la búsqueda rápida y eficaz de los incidentes que requieren de una pronta solución.

- **Filtrado por Categoría:** esta opción da facilidad a la hora de buscar incidentes de acuerdo a la categoría, en este caso se escogieron las categorías: Software Freeswitch, Software Syslog, Software Zoiper y Hardware.

- **Filtrado por Estado:** con este filtro el técnico se puede dar cuenta de que tantos incidentes han sido solucionados (Estado Cerrado) y cuales faltan por solucionar (Estado Abierto).

- **Combinación de Filtros:** además de poder utilizar un filtro de forma individual, se puede utilizar la combinación de estos, por ejemplo el técnico puede buscar incidentes de Prioridad Alta en la Categoría Software-Freeswitch y con Estado Abierto (ver figura 43).

Figura 43. Filtrado de incidentes.

Como se puede observar de la figura 43, se implementó un código de colores para identificar la prioridad y el estado de los incidentes. En cuanto a la prioridad se asignaron colores de la siguiente forma: para los incidentes críticos color magenta, para los de prioridad alta el color rojo, la prioridad media se identifica con el color

naranja, el color amarillo es para identificar la prioridad baja, y para los de prioridad planeada se identifican con el color verde iluminado. Los anteriores colores fueron mapeados del código de colores de la herramienta de monitoreo Syslog. Para identificar el estado del incidente se maneja otro código de colores que es el siguiente: para observar los incidentes de estado abierto se utiliza el color azul marino, y para identificar los incidentes de estado cerrado se utilizó el color gris.

También se implementa un sistema de alarmas para indicar al técnico el momento en el cual se ha generado un incidente de tipo crítico o de prioridad alta, que son los que necesitan ser atendidos por el técnico con mayor prontitud. Cada alarma es monitoreada por medio de un MBean, estas alarmas se pueden observar en la figura 43 (campo Nuevo) en la parte superior derecha de la interfaz. Cuando llega un incidente de tipo crítico parpadea el color magenta del cuadro que está más a la izquierda debajo de la palabra nuevo, y en el momento en que aparezca un incidente de tipo alto, el cuadro que está más a la derecha parpadea de color rojo.

4.2.4 Ticket del Incidente

El Ticket de Incidente (ver figura 44) cuenta con una serie de botones, combos y campos de textos que representan las funcionalidades tomadas de las recomendaciones del marco de referencia ITIL.

- **Mensaje Log:** este campo de texto muestra el mensaje log del incidente, el cual se debe modificar quitándole algunos componentes del mensaje, estos pueden ser: fecha y hora, dirección IP, entre otros. Lo anterior se hace con el fin de que el mensaje modificado sirva como mensaje estándar para lograr identificar incidentes del mismo tipo que pueden producirse a futuro.

- **Descripción:** en este campo se escribe una traducción en cuanto al idioma y a la interpretación del mensaje log, con el fin de facilitar al técnico la comprensión del incidente y evitar gastos de tiempo innecesarios.

- **Solución:** cuando se encuentra solución a un incidente, en este campo se escriben los pasos que condujeron a dicha solución, esto se realiza con el objeto de que cuando llegue un incidente del mismo tipo, el técnico aplique los pasos previamente almacenados y de ser necesarios nuevos pasos para la resolución, el técnico los debe escribir debajo de los anteriores con el fin de enriquecer la base de datos. Lo anterior se puede realizar si se activa el campo de texto solución por medio del botón editar solución.

- **Tiempo Restante:** en este componente se visualiza el tiempo que le queda disponible al técnico para resolver el incidente, este tiempo se asigna según la prioridad del incidente (ver tabla 1), este tiempo empieza a correr una vez el incidente llega a la base de datos del Incident Service Desk.

- **Escala:** este combo permite asignar un valor de escala al incidente que va de 1 a 4. El sistema permite cambiar el valor de escala, 1 cuando el incidente tiene almacenada previamente la solución y esta permite dar soporte al incidente ; si el incidente no tiene una solución almacenada o la solución que trae por

defecto no sirvió para dar solución, el incidente será escalado al segundo nivel; si el técnico interno de la microempresa no encontró una solución al incidente y los tiempos de resolución se agotan, se debe escalar a nivel 3 que significa que el incidente debe ser tratado por personal externo a la empresa; finalmente si aún no hay solución o hay que comprar equipos para lograr restaurar el servicio, se debe escalar el incidente a nivel 4 donde se informa de esto a los directivos de la entidad, para que tomen decisiones administrativas.

- **Cerrar Incidente:** esta funcionalidad se implementó por medio de un botón que lleva su mismo nombre, este permite tomar la fecha y hora en la cual el incidente ha sido cerrado, llenando el campo Fecha Cierre de la tabla Incidentes Disponibles (ver figura 43). Según lo recomendado por ITIL-LITE antes de cerrar un incidente se comprueba que las estimaciones de la categoría y la prioridad fueron las adecuadas, de lo contrario se puedan modificar de acuerdo a la realidad del incidente.

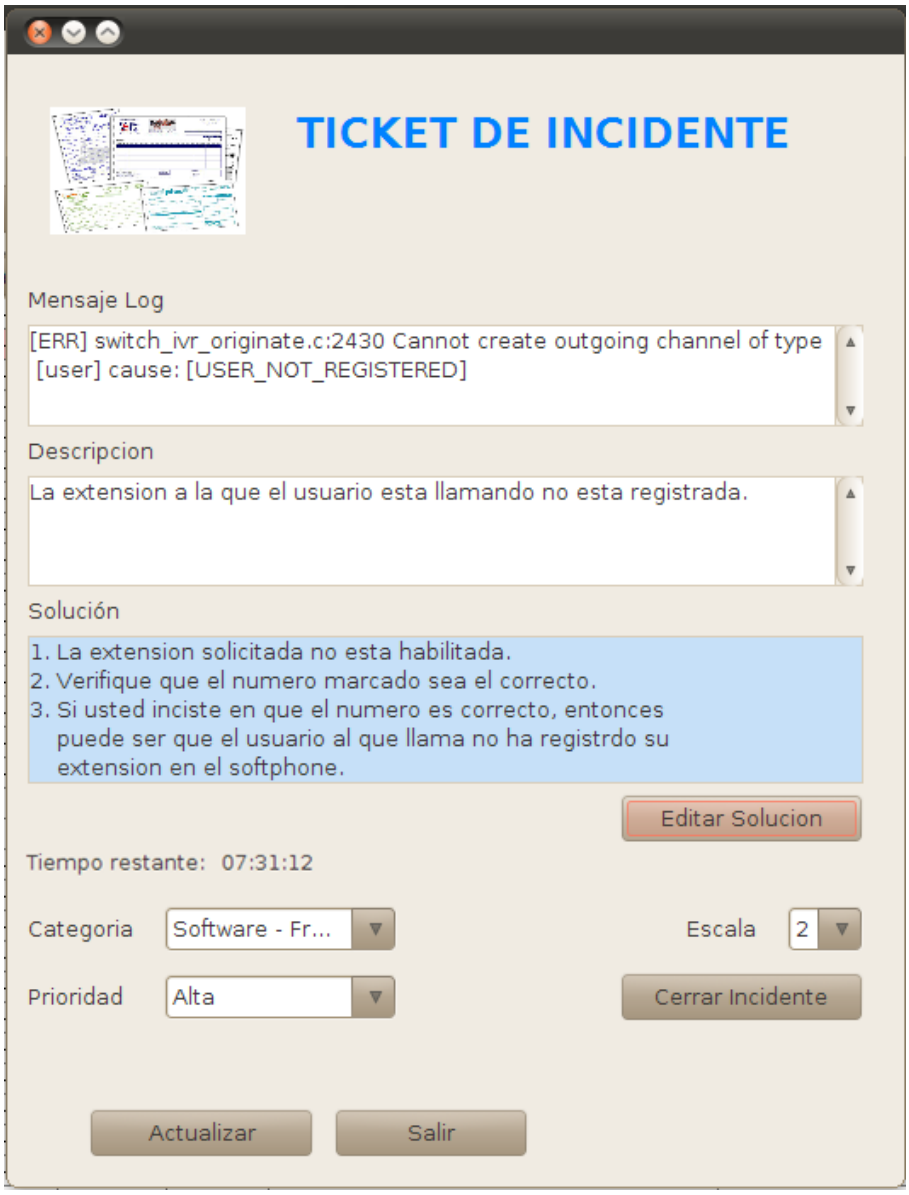


Figura 44. Ticket de Incidente

Capítulo 5. Conclusiones

- La tecnología JMX contiene niveles que permiten la construcción de arquitecturas desde sus inicios, a diferencia de otras tecnologías que sirven más para adaptarlas o adoptarlas, ya que traen marcos de gestión especificados y establecidos por organismos reguladores. Además JMX no es realmente una arquitectura de gestión, sino de instrumentación de la gestión, esta tecnología es un conjunto de bibliotecas de Java que hacen posible la instrumentación de aplicaciones.

- De las funcionalidades adoptadas de ITIL se observa que el diseño del componente Resolución y Recuperación brinda un gran valor de eficiencia a la gestión, ya que cuando se da una solución a un incidente ésta se almacena en una base de datos, con el fin de que cuando llegue un incidente del mismo tipo retorne automáticamente la solución previamente almacenada. Lo anterior hace que se reduzca tiempo y costos en la operación de un sistema de gestión.

- Este proyecto de grado hace uso del registro de eventos de un sistema, los denominados log, en este caso se registran los eventos que produce la Centrex IP, estos se guardan en un fichero de texto en un formato estándar. Cada log generado por la Centrex es interpretado o leído por el Insident Service Desk. El registro de dichos log es la materia prima para realizar la Gestión de Incidentes en la implementación de dicho sistema, por lo que muchas soluciones de gestión se pueden implementar a partir de la toma de los registros entregados por los elementos o por un sistema.

- Durante la construcción de la aplicación de gestión de incidentes bajo las recomendaciones de ITIL-LITE, se pudo plasmar en la practica lo que esta escrito en la teoría. Entonces con este proyecto de grado se puede concluir experimentalmente que si es posible implementar una versión LITE de ITIL, en cuanto a la gestión de incidentes en el entorno de las MIPYME, esto es posible gracias a la utilización de software libre en la construcción y el funcionamiento dela aplicación de gestión.

- Después de desarrollar este trabajo de grado, se logró un mayor conocimiento en cuanto a la gestión de incidentes, y fue enriquecedora la experiencia de haber conocido el marco de referencia ITIL, ya que dicho marco no solamente contiene recomendaciones sobre esta área de la gestión, sino que también aborda otras áreas referentes a los servicios IT, como son la gestión de problemas, gestión del nivel del servicio, entre otras.

5.1 Aportes

En esta tesis de pregrado se diseñó una solución para construir una Aplicación de Gestión de Incidentes de una Centrex IP, utilizando las recomendaciones de ITIL-LITE.

Para lo anterior se hizo un estudio de algunas de las tecnologías existentes en la gestión de servicios IT, del cual se dedujo que JMX es la más adecuada para este proyecto, ya que no es realmente una arquitectura de gestión, sino de instrumentación de la gestión. El diseño de la solución se planteó en cuatro niveles, nivel de sistema gestionado, es donde se encuentra la Centrex IP de la cual se extraen los log por medio de la herramienta de monitoreo Syslog-ng, estos log se almacenan en una base de datos de MySQL; el nivel de instrumentación, este agrupa las funcionalidades abstraídas de ITIL-LITE que se implementan en MBeans; nivel de agentes, es donde se encuentra el servidor de MBeans, este trabaja como un directorio para la búsqueda de MBeans; nivel de clientes de gestión, en este nivel se encuentra las interfaces gráficas de la aplicación de escritorio que son empleadas por el usuario para interactuar con el sistema y realizar las tareas de gestión.

Se diseñó con MySQL una base de datos que almacena incidentes, esta base de datos es controlada a través de un gestor que permite las funcionalidades de actualizar, eliminar, agregar y guardar tanto usuarios como incidentes. Después de diseñada esta solución se implementó el Incident Service Desk, el cual es un prototipo que abstrae las funcionalidades recomendadas por ITIL-LITE para una eficiente Gestión de Incidentes, algunas de las funcionalidades son: identificación del incidente, aquí se le asigna al incidente una identificación única, fecha y hora de apertura como datos más relevantes; priorización, que se encarga de dar un valor cuantitativo y cualitativo a la gravedad del incidente según se haya acordado; cierre de la incidencia, aquí se cierra el incidente con datos de fecha y hora del momento de solución, niveles de escalado, entre otros datos.

Finalmente se construyó un prototipo que como se ha mencionado anteriormente abstrae las recomendaciones de ITIL-LITE, fue implementado bajo el lenguaje de programación Java, utilizando el IDE Netbeans y haciendo uso de su módulo JMX. Siendo los aportes principales de esta tesis de pregrado los siguientes:

- **Base de conocimiento de la Gestión de Incidentes del marco de referencia ITIL.** El resultado de este estudio fue encontrar una técnica apropiada para llevar a cabo la Gestión de Incidentes la cual se traduce en funcionalidades como: identificación de la incidencia, registro, categorización, priorización, diagnóstico inicial, escala de incidentes, resolución y recuperación, y por ultimo cierre del incidente.

- **Diseño de una solución para realizar Gestión de Incidentes de un Centrex IP.** Para realizar ésta gestión se propuso un diseño de referencia que proporciona las funcionalidades de la gestión de incidentes recomendadas por ITIL-LITE. Para alcanzar este objetivo se toma la arquitectura en niveles de la tecnología JMX, y ya que ésta es escalable se incorpora un nuevo nivel denominado nivel de sistema gestionado, en donde se encuentran la Centrex IP, la herramienta de monitoreo Syslog-ng, y el repositorio de incidentes sin gestionar. Este diseño está pensado como base en la construcción de soluciones en el entorno empresarial de las MIPYME.

- **Construcción del prototipo.** Para la realización de este prototipo se realizó una implementación del diseño final de referencia, delimitado para realizar Gestión de Incidentes en un Centrex IP en entornos de redes LAN. La implementación de referencia se llevó a cabo con el fin de realizar pruebas del funcionamiento del sistema.

- **Componente de Solución y recuperación.** Se implementa un campo que permite editar o agregar a una base de datos los pasos que conducen a restaurar el servicio, de esta manera se mejora la eficiencia al dar solución a un incidente, ya que cuando se genere un incidente del mismo tipo el sistema entrega una solución automática, que permite reducir el tiempo y el costo en investigar y diagnosticar que falló, sino que se procede a solucionarlo más rápidamente y con mayor acierto.

5.2 Recomendaciones

- Ya que el Incident Service Desk no cuenta con la seguridad informática apropiada que resguarde su información y correcto funcionamiento, se recomienda implementar un módulo que garantice que los recursos informáticos de la empresa estén muy bien resguardados y protegidos de circunstancias y factores externos.

- Cuando se tenga una nueva iniciativa de negocio en el campo de la gestión de servicios IT utilizando el marco ITIL, se recomienda dar marcha con la Gestión de Incidentes y posteriormente implementar la Gestión de Problemas ya que estas dos son áreas muy relacionadas, luego de esto de manera progresiva continuar con la implementación de las demás áreas de gestión.

- Se recomienda hacer un estudio minucioso de la veracidad de la prioridad de algunos de los Logs que genera Freeswitch y monitorea Syslog-ng, ya que algunos Logs son representados con prioridad baja por estas herramientas y estos por el contrario, en la práctica generan alto impacto negativo en el negocio. Por ejemplo cuando la Centrex IP deja de funcionar se emite un Log de prioridad tipo NOTICE, lo que indica que es simplemente un aviso, pero en la práctica esto tendría una prioridad muy alta, pues si se detiene la Centrex IP es parar el motor del negocio.

5.3 Trabajos Futuros

Esta tesis de pregrado ha aportado soluciones al problema de la Gestión de incidentes en un Centrex IP, basándose en las recomendaciones de ITIL-LITE. Así, de acuerdo al entorno de estudio de este proyecto de grado se proponen los siguientes trabajos futuros:

- El presente proyecto abordó el área de la Gestión de Incidentes del marco de referencia ITIL, como trabajo futuro se propone la implementación del módulo de Gestión de Problemas, ya que este da soporte a la Gestión de incidentes, proporcionando información y soluciones temporales o parches.

- Diseñar e implementar una solución que permita realizar la Gestión del Nivel del Servicio, según indicadores que sean acordados dentro de un SLA, para garantizar calidad del servicio en los servicios de un Centrex IP.

- Con la llegada al mercado de la telefonía móvil de cuarta generación se hace indispensable proporcionar mecanismos de Gestión de Incidentes basándose en las recomendaciones de ITIL, ya que estas recomendaciones son ampliamente aceptadas en organizaciones proveedoras de servicios IT.

- En este proyecto se reportan los incidentes a través de tickets que se generan en una aplicación de escritorio y son visualizados en su interfaz de usuario. De lo anterior se puede observar que un trabajo futuro seria lograr enviar dichos tickets en forma de mensajes de texto al móvil del técnico o directivos, según sea el caso.

REFERENCIAS BLIBLIOGRÁFICAS

[1] Patricia MARCU, Larisa SHWARTZ, Genady GRABARNIK, "Model for Incident Ticket Correlation for Inter-Organizational Service Delivery",IIMC International Information Management Corporation, 2009.

[2] Lynn A. DeNoia, Michael G. Carper, William Hanczaryk "HOW TO FIND SELF-INFLICTED TROUBLES", Winthrop University, Indiana State University, University of South Carolina, 2009.

[3] Anónimo, "ITIL – LITE A Right-sized Approach to IT Support Services for Small to Medium Business", 2009.

[4] Osiatis, "ITIL – Gestión de Servicios IT: Gestión de Incidentes", Pagina web disponible en: http://itil.osiatis.es/Curso_ITIL/Gestion_Servicios_TI/gestion_de_incidentes/vision_general_gestion_de_incidentes/vision_general_gestion_de_incidentes.php
Última consulta: Mayo 3 de 2010.

[5] Anónimo, "Política y Procedimientos para Documentar, Manejar y Resolver incidentes técnicos", Universidad de Puerto Rico en Bayamón, Oficina de Sistemas de Información, 2007.

[6] Anónimo, "ITIL Glossary v01", Acronyms, Glosario de Términos ITIL, Definiciones y Acrónimos, Mayo de 2006

[7] Rafael Fernández Calvo, "Glosario básico inglés-español para usuarios de Internet", 2003.

[8] Ángel Hernán Tamayo Rodríguez, Orlando Patricio Valencia Pozo, "DESARROLLO DE UNA PROPUESTA PARA LA IMPLEMENTACION DE LA MESA DE AYUDA BASADA EN EL FRAMEWORK DE ITIL Y APLICADA A LA INFRAESTRUCTURA DE LA ESPOCH", Escuela Superior Politécnica de Chimborazo, Riobamba Ecuador, 2009.

[9] Adoración Marín Soler," Análisis de la Calidad Experimentada en Aplicaciones de Voz Sobre IP de libre Distribución", Escuela Técnica de Ingeniería de Telecomunicaciones, Universidad Politécnica de Cartagena, Cartagena – Colombia, 2008.

[10] Ana Quevedo Val, "Implementación de una metodología de procesos para la mejora de TI en una empresa", Universidad Politécnica Cataluña, Barcelona, 2009.

[11] The ITIL Community Forum, "Basics: Frequently Asked ITIL Questions Part 1", Página web disponible en: http://www.itilcommunity.com/
Última consulta: Mayo 12 de 2010.

[12] Jesús Dextre Tuya, "Information Technology Infrastructure Library", Página web disponible en: http://itilunfv.net16.net/
Última consulta: Junio 2 de 2010.

[13] Giovanny Javier Jiménez Cadena, Giovanny David Ramíres Gallegos," Desarrollo de un Sistema Para la Gestión de Cambios en la Infraestructura de TI. Aplicado a un Caso de Estudio", Escuela Politécnica Nacional, 2008.

[14] itSMF International "OGC anuncia la retirada gradual de ITIL v2 exámenes en itSMF Internacional de Conferencias" 2009, Pagina web disponible en:
http://www.itsmfi.org/
Última consulta: 19 de Octubre de 2010.

[15] Aileen Cater-Steel, Wui-Gee Tan, Mark Toleman, "itSMF Australia 2009 Conference: Summary Report of ITSM Standards and Frameworks Survey", University of Southern Queensland, Toowoomba Australia, 2009.

[16] Malcolm Fry "The ITIL – LITE Book", Página web disponible en:
http://www.theitillitebook.com/
Última consulta: Octubre 26 de 2010.

[17] Elena Orta, Mercedes Ruiz, Miguel Toro "Aplicación de las Técnicas de Modelado y Simulación en la Gestión de Servicios TI", Departamento de Lenguajes y Sistemas Informáticos Escuela Superior de Ingeniería, Chile, 2009.

[18] Osiatis "ITIL – Gestión de Servicios IT: Gestión de Problemas", Pagina web disponible en:
http://itil.osiatis.es/Curso_ITIL/Gestion_Servicios_TI/gestion_de_problemas/introduccion_objetivos_gestion_de_problemas/introduccion_objetivos_gestion_de_problemas.php
Última consulta: Junio 15 de 2010.

[19] Natalia Rodríguez, "ITIL – Mejores Prácticas", Centro de Investigación de las Telecomunicaciones (CINTEL), 2008.

[20] Osiatis "ITIL – Gestión de Niveles de Servicio", Pagina web disponible en:
http://itil.osiatis.es/Curso_ITIL/Gestion_Servicios_TI/gestion_de_niveles_de_servicio/vision_general_gestion_de_niveles_de_servicio/vision_general_gestion_de_niveles_de_servicio.php
Última consulta: Julio 4 de 2010.

[21] Gerardo Garrido Aguilar," Gestión de Desempeño de una Red ATM en Internet 2 Utilizando la Especificación MPLS, "Instituto Politécnico Nacional, Centro de Investigación y Desarrollo de Tecnología Digital, Tijuana, México, 2003.

[22] Anónimo, "Administración de Redes", Redes de Computadoras II.

[23] Olga Alexandra Rosero, Diego Alejandro Proaño," Estudio y Desarrollo de una Metodología para la Implementación de un Modelo de Gestión y Administración de Red Para la Universidad Técnica Estatal de Quevedo", Escuela Politécnica Nacional, Facultad de Ingeniería Eléctrica y Electrónica, Quito, 2009.

[24] Luisa Carolina Nieto H," Modelo de Gestión de Redes de Datos a Través de Web
", Universidad Centro occidental Lisandro Alvarado, Barquisimeto, 2009.

[25] Jorge Enrique López de Vergara Méndez," Especificación de Modelos de Información de Gestión de Red Integrada Mediante el Uso de Ontologías y Técnicas de Representación del Conocimiento", Universidad Politécnica de Madrid, Departamentos de Sistemas Telemáticos, Madrid, España.

[26] Jorge Enrique López de Vergara Méndez," Diseño e Implementación de un Sistema para la Gestión de una Aplicación Distribuida de intermediación electrónica", Universidad Politécnica de Madrid, 1998.

[27] Washington Rodrigo Almeida Moyano, Daniel Fernando Imbacuán López, "Benchmark Para el Uso de Tecnologías Relacionadas a Servidor de Aplicaciones Glassfish y Jboss", Escuela Politécnica Nacional, Quito – Ecuador, Mayo de 2009.

[28] Bhuiya, Badruduja, "Unified Hosting Environment for the Container Monitoring and the Adaptation", Dresden University of Technology, Marzo de 2010.

[29] Nono Carballo Escalona, "Presencia de la Tecnología Java en la Gestión (Parte I)", REVISTA TELEM@TICA, Departamento de Telemática Facultad de Ingeniería Eléctrica, Instituto Superior Politécnico José Antonio Echeverría, Habana, Cuba, 2003.

[30] Bhuiya, Badruduja,"Unified Hosting Environment for the Container Monitoring and the Adaptation", Dresden University of Technology, Department of Computer Science, 2010.

[31] Jorge López de Vergara, Víctor Villagrá, Juan Asensio, Julio Berrocal, "Análisis y Comparativa de las Alternativas Propuestas para la Gestión Basada en Web", Departamento de Ingeniería de Sistemas Telemáticos, Universidad Politécnica de Madrid.

[32] Kleber Eduardo Ayala León, Arturo Javier Hinojosa Alvear, "Herramienta Generadora de Aplicaciones Web con Arquitectura MVC en Java", Escuela Politécnica Nacional, Quito – Ecuador, 2007.

[33] Alejandra María Narváez Camayo, Omar Albeiro Trejo Narváez, "Arquitectura para Aplicaciones Web Orientada a Aspectos Basada en los Conceptos de Separación de Incumbencias", Universidad del Cauca, Popayán – Cauca.

[34] Anders Brownworth, "Asterisk vs. FreeSWITCH", Thoughts and commentary on Technology, Mayo de 2008.

[35] Fernando Álvarez Marín, "Diseño de una red telefónica IP interna entre los colegios San José – La Salle de Guayaquil y Hno Miguel – La Salle de Quito e implementación de un prototipo, usando como central telefónica servidores con Sistema Operativo libre y Software libre", Escuela Superior Politécnica del Litoral, Guayaquil – Ecuador, 2006.