



UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA  
Escuela de Ingeniería Informática



**Ingeniería Superior en Informática**

# **Análisis y Diseño de un sistema software para el registro y seguimiento de las incidencias de los clientes de la empresa Kernel Informática SL.**

**Autor:**

- **Francisco Javier Montero Vega**

**Tutores:**

- **Francisco José Santana Pérez**
- **Yeray Tejera León**
- **Norberto Navarro Espósito**

**Las Palmas de Gran Canaria, Julio 2017**

## **Agradecimientos**

*Mi más sincero agradecimiento a todos los que me han apoyado. En especial a mi Familia.*

## Contenido

Introducción .....	4
Estado del Arte y Objetivos .....	5
Historia .....	5
Motivación .....	8
Objetivos .....	14
Recursos necesarios .....	17
Entorno de desarrollo y pruebas .....	17
Entorno de producción .....	18
Análisis, Diseño e Implementación .....	19
Requisitos funcionales .....	24
Requisitos no funcionales .....	48
Diseño de la arquitectura del sistema .....	61
Diseño detallado .....	64
Implementación .....	92
Temporalización .....	109
Pruebas y mantenimiento .....	110
Pruebas unitarias y funcionales .....	112
Pruebas de Integración .....	112
Pruebas de sistema .....	113
Implantación .....	114
Instalación, configuración y carga inicial .....	115
Simulación .....	124
Personalización de Asterisk .....	132
Resolución de problemas .....	141
Análisis de costos .....	146
Conclusiones y trabajos futuros .....	149
Referencias .....	154

## Introducción

El objetivo de este Proyecto Final de Carrera es la realización de un sistema software que capture y gestione las llamadas de los clientes de la empresa Kernel Informática S.L y optimice el proceso de atención, registro y seguimiento de sus incidencias.

La terminología ITIL define que: *“La Gestión de Incidentes no debe confundirse con la Gestión de Problemas, pues a diferencia de esta última, no se preocupa de encontrar y analizar las causas subyacentes a un determinado incidente sino exclusivamente a restaurar el servicio.”* ([ITILv3])

En una empresa el sistema informático juega un papel fundamental al posibilitar la integración entre todos sus individuos, acercando la información y haciendo la comunicación más fácil. En los procesos que se llevan a cabo dentro de la empresa, hay que disponer de conexión y coordinación para asegurar el buen funcionamiento de todas las partes. La disponibilidad de un buen sistema de comunicación puede ser de gran ayuda, incrementando la eficiencia y la rentabilidad de la empresa.

Kernel Informática S.L es una empresa canaria, ubicada en Las Palmas de Gran Canaria, con una amplia experiencia en el sector de la informática.

Su parque de clientes es extenso. Abarcan servicios técnicos software y hardware, desarrollo de aplicaciones, robótica y autómatas.

El sistema software actual de Kernel informática S.L, posee varias deficiencias funcionales en el sistema de recepción de llamadas y gestión de incidencias que provocan el malestar de los clientes y el sobreesfuerzo de los empleados.

En resumen, se pretende la mejora de calidad de los procesos de atención al cliente. Hay una cita de un experto que dice:

*“Fallar en la atención al cliente es una forma bastante efectiva de dinamitar el futuro de una compañía.”* ([Servicio])

Los riesgos de desarrollar procesos de atención ineficientes o sin el control adecuado, pueden ser tan determinantes para la evolución y el futuro de una compañía, que es tan imperioso como vital, fortalecerlos y evaluarlos.

Es decir, que si la empresa fracasa en la atención al cliente, 6 de cada 10 clientes le darán la espalda, con una consecuencia dramática en los resultados de negocio.

# Estado del Arte y Objetivos

## Historia

Kernel Informática S.L es una empresa canaria con una amplia experiencia en el sector de la informática.

Su parque de clientes es extenso. Abarcan servicios técnicos software y hardware, desarrollo de aplicaciones, robótica y autómatas.

El servicio de captación, atención, seguimiento y registro se ha visto renovado en varias ocasiones para ofrecer la mejor calidad de servicio a sus clientes

Kernel Informática S.L desarrolló hace muchos años un paquete software de escritorio PC que atiende y gestiona a sus clientes. Este software era capaz de realizar desde la captación de la llamada entrante por medio de una centralita telefónica física, en aquel entonces estaba ubicada dentro de la oficina, que posteriormente procesaba para obtener de una agenda contenida en una base de datos local, el nombre del cliente. Posteriormente todos los puestos internos de la oficina leían la información del servidor local y mostraban en un breve pop-up de escritorio, quien estaba llamando.

Conjuntamente se implementó una solución software completa para la generación de notas de incidencias para su posterior seguimiento, registro y generación de partes.

Debido al auge de las nuevas tecnologías de la comunicación en centralitas telefónicas virtuales en la nube, Movistar ofreció un paquete de soluciones para empresas denominado "Movistar Fusión".

Las características de este pack de soluciones, en palabras de los comerciales de Movistar, convencieron a los responsables de Kernel Informática S.L, por lo que contrataron dicho servicio, prometiendo que mejoraría significativamente la calidad de los servicios, algo en el que siempre la empresa se ha esforzado y comprometido a mejorar durante mucho tiempo.

Se exponen los casos con más detalles:

1. Reunión con comerciales que proponen el producto de movistar.
2. Kernel informática contrata los servicios
3. El servicio tarda mucho en ser implantado, teniendo múltiples visitas de los técnicos a los largo de varios meses.
4. Los técnicos de movistar instalan terminales IP en todos los puestos de la oficina, les asigna una extensión. Solo instalan un softphone en el puesto de un técnico del servicio técnico.
5. Surgen problemas con el servicio a causa de la configuración. Los técnicos tienen que configurar los terminales y la centralita virtual.
6. Pasado un tiempo siguen con problemas a la hora de identificar las llamadas, tal como les prometieron los comerciales.
  - El software de PC tiene notificación por pop-up pero no se utiliza porque hay que configurar el servidor virtual y el software. Tras la configuración sigue sin funcionar, salvo para llamadas internas. Los técnicos se desentenden y en la actualidad, dos años más tarde, siguen con el mismo problema.

- La solución que ofreció Movistar es que hay que insertar en la agenda de cada puesto de trabajo, el nombre de cada teléfono de todos los clientes individualmente, ya que no existe una agenda cooperativa en la centralita Movistar en la nube. Esto es una gran desventaja y un paso atrás frente a lo que Kernel Informática S.L tenía anteriormente. Tener que replicar la base de datos de clientes, de forma descentralizada, con el consiguiente mantenimiento de tener al día en todos los puestos voz-ip la información de los clientes, es inviable.

La centralita al estar en la nube cambia el escenario y el software implementado por Kernel Informática S.L, encargado de capturar las llamadas queda obsoleto, al no poder acceder a las llamadas entrantes como hacía anteriormente porque está en la nube, no en local como antiguamente.

Al carecer del reconocimiento de clientes durante las llamadas entrantes, se ven obligados a improvisar y realizar un sobreesfuerzo. Actualmente miran el número que llama en pantalla, siendo un parque de clientes muy amplio y diverso, es muy difícil poder prestar un servicio en condiciones, mermando la calidad en general.

Por otro lado, el software para poder realizar la petición del cliente, realizar el seguimiento, registro y generar los partes aún no se ha actualizado ni mantenido en mucho tiempo:



**Ilustración 1 – Ventana de seguimiento de llamadas e incidencias**



Ilustración 2 – Ventana del menú principal. Programa de partes 2011

Fecha	Hora	Parte	Cliente	Incidencia (1)	Priorid	Zon	Op
22/04/2013	09:00	020438	LABORATORIO ALICIA NAVARRO E	- Sustituir discos Jaz de Copia, probar a	*****	04	04
/ /	:	013578	FARM JUAN ANTONIO GARCÍA	Explicar el funcionamiento del programa de	2968	01	
/ /	:	013681	JUEGOS SAN JOSE S.A. (ADMON)	Intentar recuperar la información del programa	2951	04	
/ /	:	014744	JOSE JUAN MATEOS PEREZ	Configurar el Delta para imprimir las Etiquetas	2635	01	
/ /	:	015270	FARM DIEGO CALVO VIERA	Quedar con ellos para prestarles la PDA y	2432	04	11
/ /	:	015283	FARM GUSTAVO SANCHEZ	Tiene problemas con el envío del pedido.	2426	07	
/ /	:	015592	FARM JAIME LÓPEZ	Cursillo Receta Electronica	2337	04	
/ /	:	015607	FARM JUAN ANTONIO GARCÍA	Cursillo Receta Electronica	2335	01	
/ /	:	015635	FARM BERNABE HDEZ CORRALES	Cursillo Receta Electronica	2328		
/ /	:	016222	FARM CARLOS MUÑOZ SANZ	Ha tenido problemas con el cierre, le impedía	2126	01	
/ /	:	017094	GEDETEC, S.L.- OFICINA TÉCNICA	- Cambiar contraseñas en los correos	1809	04	
/ /	:	017237	APOTHEKA	- Según le indican a Jorge, tienen problemas a la	1763		14
/ /	:	017233	VARIOS	Llaman porque no pueden dispensar a través del	1763		14
/ /	:	017245	VARIOS	Llama Elena para indicarme que el otro día no me	1761		05
/ /	:	018319	CUADRADO SIETE ASESORES	- Configurar los dos servidores en la red de la	1470	04	
/ /	:	020268	LOGELCAN / ÁRBORE AGUA S.L.	- Problema con los prorrateos en los documentos de	1116	02	04
/ /	:	020538	BENITO ALVAREZ LOPEZ, S.L	Recuperar el funcionamiento de DELTA	1040	02	06
/ /	:	020619	LOGELCAN / ÁRBORE AGUA S.L.	Descuadre contable en saldos de apertura del	1007	02	
/ /	:	020819	AGENCIA CARLOS VEGA ROMERO	Problemas para entrar en la web y firmar en la	902	04	
/ /	:	021088	AGENCIA JOSE LUIS RODRIGUEZ	- Problemas con el mini portatil de la madre de	761	04	15

- Sustituir discos Jaz de Copia, probar a reformatar los existentes.  
 - Realizar copia de seguridad y restaurarla en equipo auxiliar.  
 - Regenerar índices del equipo auxiliar.

Tipo incidencia: SOPORTE  
 Operario: [ ]

Pesos: [ ] Cliente: [ ] Incidencia: 1 Fecha Alta: 19/04/2013 Hora: 12:15 Días: 1088

Ilustración 3.-Ventana de partes de trabajos pendientes

Por otro lado el aplicativo para captura y seguimiento de incidencias debido a su antigüedad, Kernel Informática S.L ve la necesidad de renovarlo para aportarle mayor funcionalidad y estabilidad.

En resumen:

- Problema de captura de llamadas: No tienen una agenda corporativa, de tal manera que cuando un cliente llama, la centralita muestra el número en la pantalla de los teléfonos. Los comerciales prometieron que identificarían las llamadas, pero no que pudieran estar igual que antes, con la posibilidad de gestionar dicho número para obtener información del cliente y poder mostrarlo en la pantalla de los empleados.
- Necesidad de actualización: El software se divide en varios aplicativos donde cada ventana tiene su función, tal como muestra las capturas por lo tanto:
  - o Pérdida de rendimiento general al perder tiempo entre las pantallas.
  - o El sistema está desarrollado en una plataforma Visual Fox Pro, que está obsoleta y no recibe mantenimiento de Microsoft.
  - o No ha habido más actualizaciones ni ampliaciones de funcionalidad.

## Motivación

### Captura de llamadas

Por parte de Movistar no llega la solución que solvente la necesidad de capturar el número de teléfono cuando un cliente, de Kernel Informática S.L, llame a la oficina. Lo cual impulsa la búsqueda de soluciones que permitan volver a gestionar las llamadas de los clientes. Tal como hacían antes de contratar la solución Movistar Fusión (Centralita telefónica virtual).

Tras analizar el mercado y actualizar los conocimientos sobre las centralitas telefónicas virtuales, se puede entender una serie de conceptos básicos y necesarios para encontrar una solución.

Se realizan las siguientes operaciones para intentar capturar, dentro de la oficina, el número de llamada entrante, que es la principal prioridad:

**1. Capturar el número entrante mediante el único softphone instalado:**

- a. Los técnicos de telefónica instalaron un solo softphone. Está en un puesto dentro del departamento técnico. Se presenta un problema importante, es un software cerrado y de pago. Tras multitud de pruebas ha sido imposible obtener una solución para la captura del número.

**2. Sniffer:**

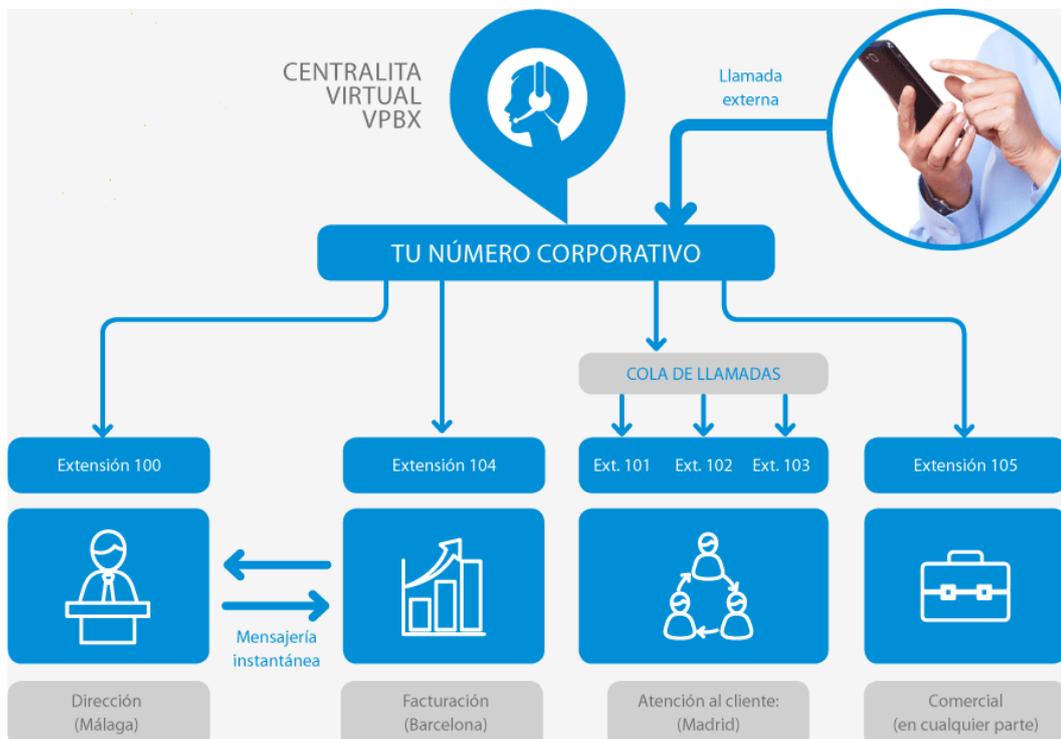
- a. Se prueba a conectar un PC a la red de la oficina, concretamente al cable de red de uno de los teléfonos IP. La idea es capturar los datagramas del protocolo SIP ([RFC-SIP]) y volcarlo en un fichero. Las conclusiones son las siguientes:
  - i. Se pueden capturar los datagramas.
  - ii. El problema es que los campos de datos contenidos en los datagramas están encriptados, por lo que no es legible la obtención del número telefónico y por lo tanto no es una solución.

**3. Acceder a la centralita o a los terminales IP:**

- a. Se analiza toda la documentación para el acceso a la centralita telefónica en la nube y a los teléfonos ip de la oficina, pero no hay solución viable.

Ante la falta de opciones, hubo que buscar soluciones alternativas. Entre estas alternativas, se consideró analizar el mercado de las centralitas telefónicas software.

En el mercado actual destaca con gran diferencia la solución de **Asterisk** y **FreePBX**.



**Ilustración 4: Ejemplo de centralita virtual en la nube**

## Estudio de Asterisk y FreePBX

### Asterisk

Es el software de código libre (bajo licencia GPL) que hace posible el funcionamiento de un servidor como **centralita de telefonía**. Se encarga de toda la parte funcional de recepción, emisión de llamadas, configuración de extensiones, colas de agentes y todo tipo de funcionalidades estándar de PBX tradicionales. Un sistema Asterisk puede manejar extensiones y líneas que pueden ser IP (si el servidor está en una red informática), o combinarse con sistemas de telefonía digital RDSI y primarios, o analógica RTB, según las tarjetas y salidas que incorpore el equipo servidor.

Actualmente las soluciones con servidor Asterisk están muy pensadas para las comunicaciones VoIP, ya que la mayoría de las instalaciones son virtualizadas en un servidor cloud (como el caso de Movistar), donde la única salida es IP.

Asterisk se configura mediante ficheros de texto de configuración y programación de dialplan específica ([Dialplan]). Es por tanto algo complejo de programar, y necesita conocimientos avanzados, por lo que realizar cambios en una extensión o cualquier tarea requerirá reescribir códigos de programación.

### Freepbx

Es un panel de configuración web, open source, para Asterisk ([PBX]). Fue creado para poder realizar de una forma gráfica y sencilla la configuración de entradas, salidas, dialplan y funciones específicas de Asterisk y que no requiriera de un alto conocimiento de programación del dialplan de Asterisk.

Examinando la web oficial ([Asterisk]) se encuentra AsteriskNow. Es una solución que integra a Asterisk y FreePBX en un sistema Linux CentOS.

Por lo tanto, se concluye que AsteriskNOW es la mejor solución para intentar capturar una llamada, a través de la programación del dialplan. A pesar de la dificultad que presenta aprender a programar Asterisk, se presenta como la única opción viable. Es una centralita telefónica software de buena reputación y muy extendida en el mercado. Tiene una alta flexibilidad y funcionalidad, permitiendo además de comportarse como centralita, la capacidad de conectarse vía protocolo Sip-Trunk (Basado en protocolo SIP) a otras centralitas telefónicas que cumplan con los estándares actuales.

Tras analizar Asterisk, se propone una solución. Programar su dialplan, de manera que se obtenga durante una llamada entrante, el número de teléfono. Llegados a este punto, un aplicativo se encargará de procesar el número, obtener información del cliente y transmitir toda la información a los puestos de trabajo dentro de la oficina de Kernel Informática S.L. Esta propuesta, guarda cierto parecido a como se hacía antes de implantar la centralita virtual Movistar.

Por lo tanto hay que realizar la integración entre el sistema que captura la llamada y el sistema de gestión de incidencias.

## Sistema de seguimiento y registro de incidencias de clientes:

Ante la necesidad de cambiar el software de gestión de incidencias, a continuación se analizará el mercado para contrastar las soluciones existentes en el campo de la captura y seguimiento de incidencias.

## Análisis de productos software

### JIRA

Jira es una herramienta de ticketing creada para **entornos con un gran despliegue de tickets desglosados por proyectos** que necesiten diferentes portales de acceso.

Jira surgió como un software de gestión de proyectos y actualmente permite incorporar distintos módulos: para el seguimiento de incidentes, IT, service desk empresarial, etc.

Hoy día, es una de las herramientas con mayor reputación del mercado. Cuenta con una **gran variedad de módulos** útiles y complementarios integrados en su núcleo, y permite la instalación de plugins (muchos de pago).

Pero no todas las prestaciones de Jira son consideradas excelentes. Un sistema de ticketing registra cada incidencia con un número identificativo (un ticket) para optimizar su localización en el sistema, por lo que lo ideal es que un software de este tipo permita buscar tickets de forma directa, a través del número de incidencia.

La principal ventaja que tiene es que está considerado ampliamente como el mejor software de ticketing.

### Desventajas

1. En Jira, el sistema de **búsqueda de tickets tiende a crear confusión** y no permite buscar directamente por número de ticket, aunque si, permite guardar las búsquedas de forma personalizada.
2. Tampoco ofrece la posibilidad de edición rápida de tickets, y es necesario actualizar manualmente las listas de tickets, pues **no dispone de una opción de autorefresco**.
3. Jira es una herramienta muy completa que a base de instalaciones de plugins puede cubrir muchas funcionalidades necesarias para grandes empresas. Su parte negativa principal es el enfoque que tiene en grandes empresas, su complejidad y el coste de la adquisición de los diferentes módulos o plugins.
4. Las licencias son gratis para proyectos OpenSource. Si no es así, ofrecen con licencia para 3 miembros un gestor de incidencia por 10 dólares/mes.

## INTEGRIA IMS

### Ventajas:

1. Integria **permite buscar tickets de forma directa**, gracias al número de incidencia (sólo Integria y Zendesk soportan esto)
2. Destaca por su precio económico en función de sus prestaciones.
3. Búsquedas precisas sin necesidad de visualizar en detalles los tickets.
4. Permite controlar varios tickets al mismo tiempo.
5. Posibilidad de **autorefresco periódico de la búsqueda**.
  
6. En la edición de un ticket, muestra un claro resumen de la incidencia, permitiendo añadir comentarios, ficheros, hacer seguimiento del ticket y ver su inventario asociado.
7. Permite el seguimiento de los tickets y ver su inventario asociado.
8. En **la base de conocimiento de Integria se pueden añadir los tickets junto con la información de artículos**, lo que permite centralizar la base de conocimiento más fácilmente y agilizar el servicio. Por ejemplo, Jira y Zendesk no permiten convertir un ticket en un elemento de la base de conocimiento.
9. Permite automatizar tareas organizando por categorías independiente de si la tarea ha sido creada o modificada. Ejecutar tareas como cambio de estado de un ticket, añadir un comentario, cambiar las fechas, o mandar un email.
10. Preparada para servicio completo a pymes, sin módulos adicionales, integrando todas las tareas relacionadas con la gestión de clientes, proyectos y todo lo relacionado con desarrollo de servicios.
11. Otra ventaja es su fácil manejo.

### Desventajas

1. Punto negativo: Tiene una pequeña comunidad.

## Zendesk

### Características principales:

1. Orientado principalmente para entornos de gestión de tickets en entornos sencillos.
2. Contiene un plan básico de ticketing y posteriormente se le puede ir añadiendo funcionalidad de pago.
3. Potente herramienta por su dashboard y generación de informes.
4. Punto fuerte es su extenso servicio de ayuda con asistentes y videos.
5. Tiene una funcionalidad llamada "play" que permite la gestión de las incidencias de una en una, el sistema en el responsable de dar la prioridad a las incidencias y no el gestor.
6. Es un software muy visual, organizado y fácil de utilizar.
7. Herramientas mediante widget que permite al cliente buscar rápida y directamente tickets.

## Desventajas

1. No dispone de una actualización automática de las vistas de los tickets (solo integra lo tiene).
2. La automatización de tareas es confusa.
3. La edición de los tickets es tan simple que no permite hacer un seguimiento detallado, asociar objetos de inventario.
4. Los tickets no se pueden convertir en artículos para la base de datos, es necesario primero escribir un artículo y luego asociarle un ticket, ralentizando el proceso.
5. No permite la opción de seguir un ticket o ver su inventario.
6. Complejidad y división en las opciones que presenta respecto a Integria ims

## Wrike

Se autoprocama como alternativa al Jira pero está más orientado a la gestión de proyectos para gran y mediana empresa.

## RT (Request Tracker)

De esta aplicación cabe señalar que permite a los usuarios configurar los campos de sus incidencias así como crear incidencias a partir de emails y generar respuestas automáticas.

Este tipo de aplicativos tiene una enorme diversidad en el mercado actual y todas entre ellas difieren en diferentes tipos de soluciones.

Se observa los aplicativos más demandados son de tipo web, debido en gran parte a la portabilidad y otras ventajas asociadas a la web.

Se espera que el aplicativo contemple una solución completa que se ajuste a las necesidades propias de Kernel informática S.L.

## Objetivos

Los objetivos del proyecto son:

1. Análisis y diseño de un sistema de gestión de tickets para el Seguimiento y Registro de las Incidencias de los Clientes (en adelante, SRIC) de la empresa Kernel Informática S.L.
2. Conexión con la centralita telefónica Movistar en la nube para conseguir que las llamadas estén conectadas con el sistema de gestión de tickets.

Se pretende hacer un sistema de gestión de incidencias por tickets conectado con la centralita telefónica de la empresa. Esta centralita telefónica es de Movistar y está situada en la nube.

La captura de incidencias de los clientes de Kernel Informática S.L es importante y por lo tanto hay que realizar un estudio de esta necesidad. Teniendo en cuenta las diferentes vías de entrada de peticiones.

Una vez captada la incidencia, es primordial realizar su registro y seguimiento. Es necesario facilitar el acceso a todos los empleados. Otro objetivo sería la creación de un portal web mediante usuario y contraseña.

Con el fin de mejorar el seguimiento de las incidencias hay que generar una base de conocimiento de incidencias y técnico de los clientes, para que pueda acceder el servicio técnico y demás empleados.

Actualmente, el software que los empleados de Kernel Informática S.L utilizan en la oficina (ver página 7), posee todos los datos de sus clientes. Sin embargo, el nuevo sistema de incidencias inicialmente no tendrá datos. Por lo tanto, otro objetivo es estudiar las diferentes vías de entrada de datos de los clientes, considerando la opción de crear una herramienta que permita importar de forma automática a los clientes, direcciones y contactos.

Para llevar a cabo la gestión de incidencias, es necesario disponer de toda la información relacionada con los clientes y sus recursos materiales. Otro objetivo es llevar a cabo la administración de los clientes y sus dispositivos.

Los directivos regularmente tienen que monitorizar la actividad de la empresa. Por lo que necesitan obtener información de sus clientes e incidencias. Agrupando sus incidencias por tipo y por estado, otro objetivo es facilitar una herramienta estadística.

Hay que mejorar la accesibilidad, flexibilidad y movilidad, por lo que se estudia la opción de ser un aplicativo web, pudiendo dar el salto a dispositivos móviles a través de una programación responsive. Permitiendo además el acceso multiplataforma de escritorio y móvil.

Se pretende realizar un estudio de cómo se configura y gestiona la centralita telefónica en la nube de Movistar, contratada por Kernel Informática S.L, ya que debido a la falta de información por parte del departamento técnico de Movistar, la información técnica es muy escasa. Sobre todo la posibilidad de capturar una llamada y como acceder al histórico de llamadas de la centralita.

Para conseguir estos objetivos se persigue los siguientes puntos:

1. Los usuarios de Kernel informática S.L accederán al sistema de incidencias según su rol. Se pretende distinguir los distintos usuarios del sistema, comenzando por distinguir entre empleados y clientes.
2. Adaptar el sistema de incidencias para que atienda las diferentes vías de entrada de peticiones. Siendo una necesidad que el cliente pueda gestionar y hacer peticiones, se estudiarán inicialmente dos vías: web y telefónica.
3. Generar una base de conocimiento de incidencias. Los empleados podrán acceder a consultar el histórico de incidencias y realizar consultas avanzadas.
4. Con el fin de mejorar la comunicación y la coordinación, para mejorar el flujo de información entre los empleados de Kernel Informática S.L, se pretende estudiar la forma de que el sistema de incidencias pueda avisar y notificar.
5. Con el fin de informar, organizar y registrar la evolución de una incidencia, se estudia la necesidad de un sistema de mensajería interno por cada incidencia, donde los empleados puedan registrar sus mensajes ante eventos, información relacionada con la incidencia, etc.
6. Al capturar una incidencia se necesita acceder a la información de clientes, dispositivos y a sus múltiples direcciones y contactos. Por lo que es una necesidad administrar: clientes con múltiples direcciones y contactos, usuarios y dispositivos. Además la configuración de algunas reglas del sistema.
7. Se pretende importar datos del software que utilizan actualmente en la oficina. Hay que estudiar la manera de establecer conexión con su sistema basado en Visual Fox Pro v9.0.
8. Es necesario monitorizar la actividad de los clientes, se pretende obtener informes que faciliten información del estado y tipo de incidencias de los clientes. Además será necesario otros informes para ciertas tareas de seguimiento y resolución de incidencias.
9. El sistema de incidencias ha de estar informado en tiempo real de cualquier llamada a la oficina. Este es otro objetivo que cubrirá una necesidad prioritaria en Kernel Informática S.L. Se estudiará los métodos que puedan facilitar el sistema de captación de llamadas para informar y hacer un registro de llamadas.
10. En la página 9, se ha realizado un breve estudio que pretendía localizar una posible solución que permitiera la captura y registro de las llamadas de los clientes de Kernel Informática S.L. La conclusión obtenida fue que se realizará a través de un servidor PBX Asterisk. Por lo que, otros de los objetivos es que Asterisk, como centralita software, escuche las llamadas provenientes de la centralita telefónica virtual de Movistar situada en la nube, contratada por Kernel Informática S.L. Se pretende lo siguiente:
  - 10.1. Tras el análisis previo, hay que estudiar la programación del dialplan de la centralita PBX Asterisk ([Dialplan], 2013) para capturar el número de teléfono en el momento que llamen a la oficina de Kernel Informática S.L.

10.2. Una vez capturado el número de teléfono, hay que consultar si se trata de un cliente y en caso positivo, obtener toda su información del sistema de incidencias, siendo otro objetivo importante la centralización para toda la empresa de los números de teléfono, o sea, la creación de una **agenda corporativa**.

10.3. Además surge otro objetivo, identificar el aplicativo encargado de procesar las llamadas ya que Asterisk por sí misma no puede realizarla, por lo tanto:

Hay dos sistemas definidos. Primero, el sistema Asterisk que captura de la línea telefónica el número de teléfono y el segundo es el sistema que ha de procesar dicho número.

Cuando se busca un aplicativo que aporte **interoperabilidad entre dos sistemas diferentes**, como este caso un Sistema Linux con servidor Asterisk ([Asterisk]) y un aplicativo realizado en Webdev sobre Windows, estamos definiendo un aplicativo de servicios web o **webservice** ([WS]).

10.4. Otro objetivo es detectar como y cuando ha de llamarse al webservice de modo que procese en el menor tiempo posible la llamada. Disponiendo de los datos de la llamada del cliente en la menor brevedad.

10.5. Otra necesidad es la comunicación dentro de la oficina de Kernel Informática S.L de los datos generados por el webservice al sistema de incidencias. Se propone el estudio de los siguientes métodos:

10.5.1. Compartir la base de datos del webservice y del sistema de incidencia en el mismo servidor de la oficina. La consulta se realiza directamente sobre base de datos.

10.5.2. La posibilidad de **implantar un sistema de RSS** (Sistema para compartir información en la web de manera muy simple) ([RSS]). Se basaría en consultar el fichero RSS dentro de la red de la oficina.

10.6. En resumen, se pretende conseguir un sistema de captación de llamadas entrantes parecido al que Kernel Informática S.L tenía antes de la implantación de Movistar Fusión (página 5). Pero en este caso, mediante centralita telefónica software (vía protocolo SIP ([RFC-SIP])).

## Recursos necesarios

### Entorno de desarrollo y pruebas

#### Hardware:

- Portatil HP pavilion g6 + monitor Sony x72
  - 4GB ram 800 mhz.
  - Procesador intel I3-2350M.
  - SSD 240gb
  - Pantalla de 15'4 pulgadas. Resolución: 1366x768..
  - Segunda pantalla, Sony x72. Resolución: 1280x1024.
  - Teclado Querty integrado.
  - Ratón Logitech M210 óptico. Resolución: 1000dpi.
  - Controlador de sonido: 1-IDT High Definition.
  - Wifi Intel tipo n
- Samsung S4 GT-I9505
  - Android 5.0.1.
  - Pantalla AMOLED de 5". Resolución: 1920x1080.
  - 2GB ram.
  - Capacidad 16GB.
  - Conectividad: Wifi, bluetooth, nfc, gps y 4g.
  - Cámara: trasera de 13Mpx/frontal 5Mpx.
- Servidor en la nube para pruebas.
  - Facilitado por Kernel Informática S.L

#### Software:

- Windows 7 profesional 64bits
  - Licencia Original (HP pavilion g6)
- Programación en WebDev 21
  - Licencia obtenida por Kernel Informática S.L.
- USB Network Gate Versión 7.0 (build 7.0.1370)
  - Licencia obtenida por Kernel Informática S.L.
- WampServer 3.0.4 64bits
  - Licencia GPL.
  - Apache 2.4.18 - PHP 5.6.19 - MySQL 5.7.11.
- Navegador Chrome Versión 58.0.3029.110 (64-bit)
  - Licencia de software libre.
- Avira Free Antivirus Versión 15.0.26.48.
  - Licencia Free
- Microsoft Office
  - Licencia obtenida por Kernel Informática S.L
- Umbrello UML Modeler 2.18.2
  - Licencia GNU versión 2
- Dia (diagramas estructurados) 0.97.2
  - Licencia de software libre

- Virtualbox 5.1.10
  - Código abierto bajo los términos GPL 2
  - Contiene una máquina virtual con:
    - Centos 6.5 (Final); Free software, GPL
      - Ejecutando: AsteriskNOW (Servidor PBX) v10.13
- Xmind 7 free
  - Licencia open source (LGPL v3 y EPLv1)
- Google Drive 2.34.5075
  - Código abierto

## Entorno de producción

### Hardware:

- Equipo Acer Aspire XC-605
  - 4GB ram 1066 mhz.
  - Procesador intel I5-4440.
  - SSD 240gb.
  - Monitor ACER 19" V193W. Resolución: 1440x900
  - Teclado y ratón originales de Acer.
  - Ethernet Gigabit
- Multifunción Lexmark X3550
  - 24 ppm negro y 17 ppm color
  - Impresión máxima 4800x1200 ppp
  - Resolución óptica 600 x1200 ppp

### Software:

- Windows 7 profesional 64bits
  - Adquirido con el equipo ACER Aspire XC-605
- Webdev 21 application server
  - Licencia obtenida por Kernel Informática S.L.
- HiperFileSQL server
  - Libre distribución
- HiperFileSQL control center
  - Libre distribución
- WampServer 3.0.4 64bits
  - Licencia GPL.
  - Apache 2.4.18 - PHP 5.6.19 - MySQL 5.7.11.
- Virtualbox 5.1.10
  - Código abierto bajo los términos GPL 2
  - Contiene una máquina virtual con:
    - Centos 6.5 (Final); Free software, GPL
      - Ejecutando: AsteriskNOW (Servidor PBX) v10.13

# Análisis, Diseño e Implementación

## Adquisición de información

La primera toma de contacto, es mantener una reunión con el máximo responsable de la empresa Kernel Informática S.L. El propósito es obtener una vista global del proyecto.

## Primera entrevista

La entrevista con D. Norberto Navarro Espósito, director de la empresa Kernel Informática S.L, se orienta a la necesidad de desarrollar un software que tengan como objetivo mejorar la calidad del servicio de atención al cliente, el seguimiento y gestión de sus incidencias.

La solución software la utilizará internamente el propio personal de la empresa y de forma externa, los clientes. Aportará una mejora en la eficiencia de la actividad rutinaria de los empleados y en la atención del cliente.

Se ha estudiado múltiples razones por el cual se decide desarrollar una solución software a medida. La principal es que tras estudiar soluciones software que hay actualmente en el mercado, estas no se ajustan plenamente a los requerimientos de la empresa.

Las necesidades que actualmente no tienen son:

- Captura y procesamiento de la llamada entrante. Identificando a los clientes antes de descolgar.
- Integración del registro de llamadas entrantes conjuntamente con el sistema de registro y seguimiento de incidencia de clientes.
- Reducir el número de llamadas de los clientes solicitando información del estado de su incidencia vía telefónica. Los clientes tendrán acceso al aplicativo donde podrán consultar y a su vez enviar incidencias.
- Optimizar el seguimiento de incidencias (temporización de tareas y alertas)
- Mejor organización, monitorización, consulta y acceso a los registros de incidencias y documentos mediante un sistema de conocimiento. Todos los empleados estarán informados del seguimiento y estado de las incidencias, reduciendo las interrupciones y llamadas internas entre ellos.
- Permitir acceder al aplicativo desde el exterior.
- Control de acceso según los diferentes roles de usuario.

## Entorno de la solución

Se describe un entorno web principalmente por la portabilidad necesaria para acceder e interactuar remotamente, tanto a empleados como a clientes.

La integración del registro de llamadas conllevará a un esfuerzo adicional donde será necesario buscar una solución para interceptar las llamadas entrantes de la oficina, filtrarlas y procesarlas para obtener, en caso de que sea un cliente, toda la información para que el personal pueda prestar un servicio de calidad.

## ¿Hay alguna limitación o aspecto especial de rendimiento que vaya a afectar a la forma en la que se enfoque la solución?

Antes de la primera reunión de grupo se plantean aspectos generales a tener en cuenta. Sin tener definidos los requisitos del proyecto, se enfocan los siguientes riesgos:

- Al tratarse de una aplicación que va a tener comunicación con varios puestos de trabajo de usuarios, se plantea que un bajo rendimiento en la comunicación de red pueda repercutir, y de manera global, con un nivel no óptimo de rendimiento.
- La navegación tiene que ser rápida, siendo un interface minimalista.
- Desde que se captura la llamada hasta que informa al aplicativo y por ende a los empleados, tiene que ser en los primeros tonos.
- La respuesta del sistema al estar muchos usuarios conectados no debe suponer un problema.
- El hardware de la máquina donde corra tiene que tener suficiente potencia para garantizar un trabajo fluido.

## Establecimiento de reuniones en grupo mediante TFEA

Mediante la técnica para facilitar las especificaciones de la aplicación ([Press10], 2010), se pretende involucrar a todos los usuarios del aplicativo. Las reuniones tendrán como fin ahorrar tiempo, consensuar conjuntamente los requerimientos y minimizar los riesgos. Se tomarán en conjunto las medidas para la resolución de problemas, negociación y especificación.

Los participantes en la reunión fueron:

Representante del grupo administrativo:

<b>Participante</b>	<b>Norberto Navarro</b>
<b>Organización</b>	Kernel Informática S.L
<b>Rol</b>	Directivo
<b>Es desarrollador</b>	Si
<b>Es cliente</b>	No
<b>Es usuario</b>	Si
<b>Comentarios</b>	<ul style="list-style-type: none"><li>• Director de Kernel Informática S.L.</li><li>• Analista y programador con 30 años de experiencia.</li><li>• Tutor del proyecto</li></ul>

Representante del grupo técnico:

<b>Participante</b>	<b>Armando</b>
<b>Organización</b>	Kernel Informática S.L
<b>Rol</b>	Administrador de sistemas
<b>Es desarrollador</b>	No
<b>Es cliente</b>	No
<b>Es usuario</b>	Si
<b>Comentarios</b>	<ul style="list-style-type: none"><li>• Jefe del servicio técnico.</li></ul>

Representante de administración:

<b>Participante</b>	<b>Susana</b>
<b>Organización</b>	Kernel Informática S.L
<b>Rol</b>	Recepcionista
<b>Es desarrollador</b>	No
<b>Es cliente</b>	No
<b>Es usuario</b>	Si
<b>Comentarios</b>	<ul style="list-style-type: none"><li>• Encargada de recepción de llamadas de clientes.</li></ul>

Representante de cliente:

<b>Participante</b>	José
<b>Organización</b>	Empresario
<b>Rol</b>	Cliente de Kernel Informática S.L
<b>Es desarrollador</b>	Si
<b>Es cliente</b>	Si
<b>Es usuario</b>	Si
<b>Comentarios</b>	<ul style="list-style-type: none"><li>• Asesor de Kernel Informática</li><li>• Cliente de Kernel Informática</li></ul>

Se define como moderador a Norberto Navarro Esposito, Director de Kernel Informática S.L y tutor. Expone ante el grupo presente su visión general del proyecto, su necesidad y justificación del nuevo producto.

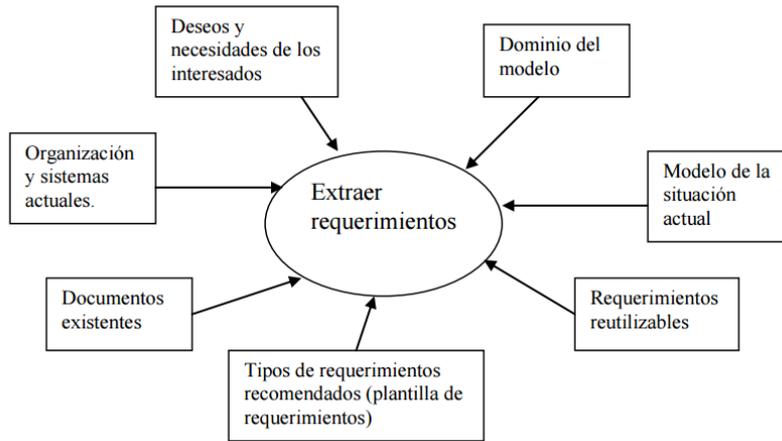
Posteriormente, los representantes de forma coordinada exponen sus propuestas de refinamiento, puntos de vistas y pasos concretos hacia el desarrollo de la aplicación. Como resultado se obtiene las listas consensuadas.

Cada representante, posteriormente a la visión general del proyecto expuesta por Norberto, analiza sus prioridades y necesidades esenciales, confeccionando individualmente una lista de objetos que se encuentren en el entorno que rodea al sistema, operaciones que manipulen o interactúen, restricciones y criterios de rendimiento. Obteniendo un conjunto de mini especificaciones.

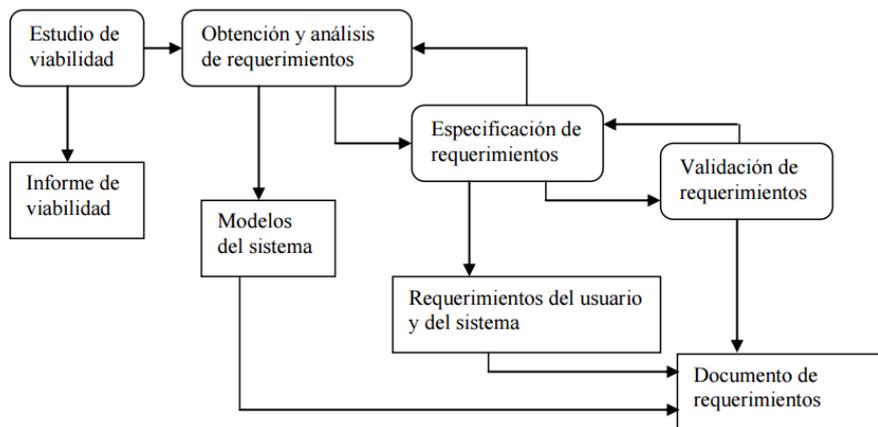
Finalmente, las mini especificaciones se presentan a todo el grupo para pasar a analizarlas. Con ello se obtuvo que añadir correcciones, adiciones y eliminaciones de requerimientos, servicios y restricciones en las listas originales.

El análisis de todas las mini especificaciones pasó por una siguiente etapa de validación, donde se tiene en cuenta repasar los riesgos y la viabilidad del conjunto de requisitos.

La primera reunión finaliza con la obtención de un borrador inicial. Tras analizar todas las mini especificaciones y validar las ideas de equipo se redacta un borrador en conjunto que plasme todos los aspectos consensuados y revisados en la reunión.



**Ilustración 6 . Proceso de ingeniería de requerimientos ([Somm05], 2005)**



**Ilustración 5: Obtención y análisis de requerimientos ([Somm05], 2005)**

Los aspectos que no pudieron ser resueltos en la reunión se conservaron para posteriores reuniones, donde se trataron y resolvieron. En algunos casos puntuales con técnicas de observación y entrevistas con empleados durante la jornada de trabajo.

Las necesidades nombradas de mayor a menor prioridad son:

1. El sistema SRIC tiene que realizar los procesos de gestión de incidencias, cumpliendo con las reglas de negocio establecidas, respetando los roles, aportando mayor funcionalidad y facilidad en general.
2. Otra prioridad es la búsqueda de una solución al reconocimiento de las llamadas entrantes a través de la centralita virtual Movistar.
3. De cara al desarrollo del proyecto, es importante poder empezar a utilizarlo en la menor brevedad, comenzando por la tarea de captura y registro de la llamada. Progresivamente ir implementando otras necesidades.

Como resultado de los documentos generados en las reuniones establecidas mediante TFEA, se obtiene como salida el siguiente documento de requisitos funcionales y no funcionales.

## Requisitos funcionales

### Identificación inicial del Aplicativo Web

Inicialmente se identifica la composición del sistema software, compuesta por los siguientes sistemas:

**1. Sistema de administración:** El administrador o supervisor es la máxima autoridad, tiene control total sobre el sistema, incluyendo la gestión. El resto de empleados, según sus diferentes perfiles de usuario, podrán administrar de forma limitada.

**1.1 Gestión de usuarios:** Consulta y administración de usuarios.

**1.2 Gestión de Clientes:** Consulta y administración de clientes.

- **Gestión de direcciones**
- **Gestión de Contactos**
- **Importación desde Visual Fox Pro**

**1.3 Gestión de dispositivos-aplicativos:** Administración de dispositivos y elementos software de los clientes.

**1.4 Gestión de informes:** Se encargará de generar informes administrativos y estadísticos.

**1.5 Gestión de reglas:** Orientado al control del comportamiento de ciertas tareas o eventos del sistema. El administrador podrá configurar un conjunto de reglas.

- **Calendario lectivo**
- **Control de tiempos (tipo incidencia-prioridad)**

**1.6 Gestión de incidencias:** Todos los empleados tendrán acceso al sistema de incidencias.

**2. Sistema de incidencias:** Encargado del registro, documentación, seguimiento, notificación y resolución de incidencias.

- **Base de conocimiento (Histórico de incidencias):** Provee de funciones que facilitan el seguimiento de las incidencias además de los medios para la recolección, organización y recuperación de conocimiento.
- **Sistema de Notificación:** La base de conocimiento notificará sobre la caducidad de las fechas de solución y resolución para un correcto seguimiento de las incidencias.
- **Comentarios internos:** La comunicación e información de eventos y estados son relevantes para un buen seguimiento.
- **Portal web del cliente:** Funciones que puede realizar el cliente.
- **Procedimiento de captura, seguimiento y cierre:** Requisitos para el procesamiento de las incidencias.

### 3. Sistema de control

- **Control de acceso por usuario:** El sistema tiene que gestionar el control de acceso al sistema, mediante usuario y contraseña.
- **Control de estimación de tiempos (tipo incidencia-prioridad):** Gestiona la estimación de tiempo durante la creación de una incidencia, determinado las reglas del calendario laborable y la tabla de tiempos en función del tipo de incidencia y prioridad.
- **Control de llamadas entrantes:** Capturar las llamadas entrantes de la centralita telefónica virtual en la nube, mostrarlas con información detallada del cliente, facilitando al usuario de atención al cliente saber su situación antes de descolgar.

### 4. Sistema de mantenimiento

- **Gestión de mantenimiento software:** Se generará logs e información que facilite el mantenimiento y depuración.

## Informe detallado

A continuación se explica en detalle los sistemas identificados:

### 1. Sistema de administración

El sistema de administración tiene como fin la gestión total de los recursos y control sobre la actividad de la empresa en su conjunto. Por ello contiene una gestión y accesibilidad de consulta global al cual tiene acceso exclusivo el actor “supervisor”. El resto de usuarios tendrán un acceso limitado.

#### 1.1 Gestión de usuarios

- 1.1.1. El supervisor tendrá acceso a dar de alta, dar de baja y modificar a usuarios. Los campo del fichero de usuarios será:

##### Datos del usuario

- Usuario-ID (identifica al usuario)
- Rol (Tipo de usuario)
- Nombre + Apellidos
- Documento identificativo
- Dirección-ID (Identifica a la dirección del cliente)
- Cliente-ID (Identifica al cliente)
- Teléfono
- Usuario (Dirección email)
- Contraseña
- Validación por el administrador

- 1.1.2. Los empleados de Kernel Informática S.L con rol de jefe del servicio técnico y supervisor podrán consultar y editar los usuarios, exclusivamente.
- 1.1.3. Los usuarios que sean clientes, una vez logueado únicamente podrán acceder a la modificación de su usuario y contraseña. Cualquier cambio de sus datos de alta ha de ser por personal de Kernel Informática S.L. con rol de supervisor.
- 1.1.4. Al eliminar un usuario este no eliminará a su cliente ni todo identificador de dirección o contacto vinculados al mismo.
- 1.1.5. Puede existir usuarios sin cliente, pero siempre con un rol de empleado.

#### 1.2. Gestión de Clientes

- 1.2.1. El cliente a través del portal web podrá darse de alta en el sistema:
- El cliente rellena el formulario de alta.
  - El sistema registra el alta con Rol de Cliente de forma automática.
  - El sistema envía un email a los supervisores, notificando el nuevo alta.
  - El cliente accede al portal y puede enviar su incidencia.

- El cliente aparecerá para los empleados de Kernel Informática S.L sin contrato y no verificado en el sistema, hasta que el supervisor valide el usuario una vez verifique que el cliente y sus datos son correctos.
- 1.2.2.** El sistema tiene que alertar visualmente a los empleados cuando seleccionan en una captura de incidencia, como en una consulta de cliente, a aquel cliente que no tenga su contrato de mantenimiento en vigor (si está al día en las mensualidades). Tiene que acompañar la descripción: “Contactar con el supervisor” en caso de estar capturando una incidencia vía telefónica.
- 1.2.3.** El sistema tiene que administrar correctamente la base de datos cuando se elimina un cliente. De modo que si este tiene alguna incidencia en el sistema no podrá ser eliminado.
- 1.2.4.** El supervisor ha de mantener siempre actualizada la información de los clientes que estén al día en las mensualidades. En caso de no estar al día, o bien, suspender su contrato por alguna razón administrativa, tiene que marcarlo en el sistema. Si hubiera algún aviso importante o restricción en su servicio lo tiene que indicar en el campo descripción para informar a los empleados.
- 1.2.5.** Los empleados de Kernel Informática S.L con rol inferior al del supervisor, podrán acceder a la consulta de las fichas de clientes, direcciones y contactos. El supervisor además podrá administrarlas.
- 1.2.6. Gestión de Direcciones**
- El cliente tiene como mínimo una dirección, pudiendo tener múltiples donde una sería la dirección de la central y el resto las sucursales.
  - Toda dirección tiene como mínimo un contacto.

Datos de dirección (emplazamientos)

- Dirección-ID (identifica una dirección de empresa)
- Calle, número, población, provincia, Zip, país
- Edificio, planta, departamento, zona (puesto de trabajo)
- Teléfono
- Plano (mapa)

**1.2.7. Gestión de Contactos**

- Todo contacto se vincula con un cliente y una dirección.

Datos de contacto

- Contacto-ID (Identifica al contacto)
- Cliente-ID (Identifica al cliente)
- Dirección-ID (identifica una dirección de empresa)
- Nombre + apellidos
- Teléfono
- Cargo
- Descripción

### **1.2.8. Importación de clientes desde Visual Fox Pro**

- Los clientes actualmente se encuentran registrados en la base de datos de Visual Fox Pro v9.
- El sistema tiene que facilitar al rol supervisor la importación de clientes desde Visual Fox Pro 9, de tal manera que se automatice este proceso el cual crearía la empresa, dirección, contacto y el usuario por cada cliente importado.
- En la importación tiene que mostrar los clientes que ha procesado.
- Durante la importación verificará que ya no existan, en caso de existir, no lo registra.
- Debe almacenarse el identificador de cliente que tenía en Visual Fox Pro v9.

## **1.3. Gestión de dispositivos-aplicativos**

- 1.3.1.** Los clientes poseen dispositivos y elementos (hardware y software). Se pretende hacer un registro de todos ellos por cliente, de manera que agilice la detección de las incidencias y la localización de los mismos.
- 1.3.2.** Solo un técnico, jefe técnico y supervisor podrán administrarlo.
- 1.3.3.** Los dispositivos podrán ser identificados por una o varias imágenes.
- 1.3.4.** Se podrán hacer relaciones entre dispositivos y elementos (software), creando una relación que será mostrada en el listado de dispositivos.
- 1.3.5.** Los dispositivos pueden ser parte de otros dispositivos, por lo que hay que incluir el identificador del dispositivo padre si lo tuviera. (Ejemplo: Una tarjeta gráfica pedida al proveedor incluirá el identificador del PC donde se colocará)
- 1.3.6.** Datos de los ficheros para su gestión:

### **Datos de dispositivos (hardware)**

- Tipo de dispositivo
  - Servidor
  - Equipo de sobremesa
  - Portátil
  - Impresora laser
  - Impresora
  - TPV
  - Pantalla
    - CPU
    - Cajón portamonedas
    - Impresora Ticket
    - Escáner
    - Visor de precios
  - Switch
- Icono
- Imagen

- Emplazamiento
- Sistema operativo
- Contacto-ID
- Número de serie (del fabricante)
- Dispositivo-ID (identificador único interno)
  - Normalmente el dispositivo tiene una etiqueta adhesiva pre-impresa a la vista.
- Teamviewer-ID
- Teamviewer-Clave

#### **Datos de elemento**

- Elemento-ID
- Denominación del elemento

#### **Datos de enlace entre Dispositivos y Elementos**

- Dispositivo-ID
- Elemento-ID

#### **Datos de Aplicativos**

- Aplicativo-ID
- Denominación (Novophar, Delta, Stocklight, Antivirus, etc.)
- Módulo
  - Compras
  - Ventas
  - Informes
  - Otros

### **1.4. Gestión de informes**

- 1.4.1.** Todos los informes son accesibles por todos los empleados.
- 1.4.2.** Tal como se especificó en el PFC-1, hay que generará una estadística por cliente y tipos de incidencias en formato CSV.
- 1.4.3.** Informe estadístico sobre la cantidad de incidencias por Cliente, clasificado por los estados propios de la incidencia. En formato CSV
- 1.4.4.** Obtención de los informes sobre las encuestas de satisfacción de los clientes. Hay que indicar el identificador único de la incidencia para generarlo y descargarlo en formato PDF. También está accesible accediendo directamente a la ficha de la incidencia, buscando en el sistema de conocimiento por el Identificador de la incidencia.
- 1.4.5.** Informe de llamadas: Obtener por pantalla, aplicando un filtrado de fecha de inicio y final, el histórico de llamadas, pudiendo exportarlo a PDF.
- 1.4.6.** Partes de trabajo:

- Los partes de trabajo los generan los técnicos, jefes técnicos y supervisores, en la ficha de la incidencia. En formato PDF.

## 1.5. Gestión de reglas.

### 1.5.1. Los supervisores podrán habilitar y deshabilitar las siguientes reglas de negocio:

- **RN-01.** Seleccionar al agente o empleado responsable directo de toda incidencia que se genere en el portal web del cliente. Si el usuario fuera eliminado, el sistema lo reemplazará por un técnico, jefe técnico, o bien supervisor. Esto sería de forma automática.
- **RN-02.** Limitar el número de usuarios no validados por un supervisor. Se pretende garantizar la seguridad del sistema ante ataques masivos de registros, provocaría muchos problemas y pérdidas de tiempo. Si el número de usuarios no validados es igual o inferior al indicado en esta regla, se podrá crear un registro nuevo.
- **RN-03.** Habilitar el aviso de encuesta de satisfacción: Si está habilitado, podrá seleccionar que empleados recibirán en su email (nombre de su usuario) un email con la encuesta del cliente, adjunto en pdf.

### 1.5.2. La combinación de las reglas “Calendario lectivo” y “Control de tiempos (Tipo incidencia-Prioridad)” darán un tiempo en horas que será el dato de entrada para el apartado 3.2 de la página 36 (Estimación de tiempos (Tipo incidencia-Prioridad) )

### 1.5.3. Calendario lectivo

- Se establecerá un calendario de lunes a domingo
- Cada día tiene dos turnos. El de mañana y de tarde.
- Se establece hora de inicio y final para cada turno.
- Habrá un marcador indicando si es o no día festivo por cada día.
- Habrá dos opciones: inicializar el año e inicializar la semana.
- Si inicializa el año, reiniciará todos los horarios, pero no los días festivos.
- Si inicializa la semana, todo día que tenga un horario diferente se almacenará como día personalizado y será el que tenga validez ese día.

### 1.5.4. Control de tiempos (Tipo incidencia-Prioridad)

- Se establece una tabla con los tiempos mínimos de respuesta para dar una solución o resolución de una incidencia.
- De la regla “Calendario lectivo” se obtiene el tiempo lectivo y desde esta tabla el tiempo necesario para una respuesta o resolución. Estos datos los utilizará el sistema de control de fecha del apartado 3.2 de la página 36 (Estimación de tiempos (Tipo incidencia-Prioridad)).
- **Ejemplo: [tipo de incidencia , Prioridad , Hora Mínima solución] = 1 hora**

## **1.6. Gestión de incidencias:**

- 1.6.1.** Todos los empleados de la empresa Kernel Informática S.L tendrán acceso a la captura y gestión de incidencias.
- 1.6.2.** La captura se puede hacer por las siguientes vías: telefónica o web.
- 1.6.3.** En la captura de incidencia vía telefónica el empleado visualizará en el aplicativo la información relacionada con el cliente que llama antes de descolgar y atenderle. Como resultado de la atención obtiene la información que necesita para registrar en el aplicativo su incidencia.
- 1.6.4.** En la captura de incidencia vía web, el cliente tendrá que acceder mediante el usuario y contraseña, asociado a la localización de un cliente, dirección y contacto. Posteriormente, redactar su problema y registrarlo en el aplicativo. Dicha incidencia se le asigna un agente responsable de su resolución. Este agente se asigna por el administrador del sistema mediante una regla de negocio, donde selecciona a que miembro del personal hace responsable.
- 1.6.5.** Los usuarios, mínimo técnicos, podrán gestionar las incidencias que aún no estén resueltas, con el fin de resolverlas.
- 1.6.6.** Dentro de las gestiones internas de una incidencia se encuentra documentar mediante comentarios todas las acciones técnicas, eventos, mensajes o documentación relevante durante las tareas de búsqueda de soluciones. Además de poder ajustar las fechas, estado, agente responsable o descripción general de la misma.
- 1.6.7.** Como mínimo, el usuario técnico puede cerrar una incidencia y generar una previsualización del parte de trabajo. Sólo un jefe técnico o supervisor podrá además entregar el parte al cliente, su función es verificar que es correcto.

## **2. Sistema de incidencias**

- 2.1.** Las incidencias de los clientes emitidas desde el portal serán asignadas a un agente responsable, seleccionado por una regla de supervisor.
- 2.2.** Todas las incidencias son accesibles por todos los empleados de Kernel Informática S.L mostrando información de: Fecha mínima y máxima para aportar una solución o respuesta y resolución del problema, respectivamente.

### **2.3. Base de conocimiento (Histórico de incidencias)**

- 2.3.1.** Provee de funciones que facilitan el seguimiento de las incidencias además de los medios para la recolección, organización y recuperación de conocimiento.
- 2.3.2.** Los empleados tendrán acceso al histórico de incidencias.
- 2.3.3.** El acceso y búsqueda en el histórico facilitará la labor a los empleados que necesiten información sobre cualquier incidencia o temática de problemas.
- 2.3.4.** La búsqueda será muy asociativa por todos los campos de información de incidencias y clientes. De tal manera que devuelva como acierto a toda subcadena de búsqueda contenida en cualquiera de sus campos. Dando mucha potencia a la hora de localizar incidencias.

- 2.3.5. Para localizar una incidencia en concreto se puede indicar el identificador de la incidencia o número de ticket directamente.
- 2.3.6. Todas las incidencias podrán contener múltiples marcadores personalizados (Tags) que faciliten la categorización e identificación de las incidencias. Con la intención de remarcar y facilitar el acceso a aquellas incidencias (cerradas o no) y documentadas con soluciones sobre problemas específicos, ayudando a generar documentación técnica en base a todos los comentarios y documentos adjuntos a los mismos, más la redacción de la resolución técnica al cerrar la incidencia.
- 2.3.7. Mínimo el rol de técnico pueden acceder a la sección de resolución de una incidencia donde podrán redactar la resolución de cara al cliente y además añadir o quitar marcadores o tags, que estarán visibles en una lista.
- 2.3.8. Toda búsqueda de conocimiento además podrá aplicar filtros.

## **2.4. Sistema de notificación**

- 2.4.1. Posee notificación visual de caducidad de fechas y notificación de mensajes en el sistema de conocimiento (cambios, o bien, nuevo comentario interno en una incidencia)
- 2.4.2. El sistema de conocimiento informará con color amarillo sobre la fecha de estimación para dar una solución al cliente si esta se caducara.
- 2.4.3. El sistema de conocimiento informará con color rojo si ha sobrepasado la fecha para resolver la incidencia.
- 2.4.4. El sistema de conocimiento informará marcando toda la incidencia de gris si el responsable de la incidencia (que no es el empleado que la observa) no ha accedido a ver algún comentario nuevo que ha escrito un compañero/a dentro de su incidencia, o bien, acceder a cualquier evento sucedido.
- 2.4.5. Si un usuario con rol mínimo de técnico cambia el estado de una incidencia, sea o no suya, el sistema notificará al supervisor responsable de la misma.
- 2.4.6. Estos avisos serán visible por todos los empleados.

## **2.5. Sistema de comentarios internos por cada incidencia**

- 2.5.1. Por cada incidencia, los usuarios de rol técnico como mínimo pueden añadir comentarios personales en la ficha de la incidencia.
- 2.5.2. Por cada comentario además del mensaje, permite adjuntar cualquier fichero que quedará adjunto al comentario.
- 2.5.3. Los comentarios serán visibles por todos los empleados, pero si no son técnicos como mínimo no podrán adjuntar ficheros al comentario.
- 2.5.4. Todos los empleados desde la vista del gestor de conocimiento pueden seleccionar una incidencia y observar los comentarios y descargar los ficheros adjuntos de sus comentarios si tuvieran.
- 2.5.5. En caso de cambio de estado de una incidencia el sistema registrará en la base de datos la transición de un estado a otro. Además añadirá un comentario de

diferente color dentro de la incidencia informando del evento ocurrido y el nombre del usuario que lo realizó.

## **2.6. Portal web del cliente**

- 2.6.1.** Los clientes pueden consultar el registro de sus incidencias, así como el estado de cada una de ellas, por medio del portal web.
- 2.6.2.** El cliente desde su portal web puede enviar una incidencia. El sistema le asignará un número de ticket que podrá consultar en la información de su incidencia, además mostrará el estado y la fecha de registro.
- 2.6.3.** Como parte de los informes generados, el cliente puede rellenar y enviar un formulario de satisfacción, incluida su opinión personal, por cada incidencia que tenga en su portal. Quedando registrado en el sistema. Esta acción envía al instante un email a los empleados, previamente seleccionados mediante una regla de supervisión, donde adjuntará el documento en formato pdf.

### **Datos de tipo de incidencia**

- Incidencia-ID
- Denominación
  - Hardware
  - Software
  - Operativo
  - Antivirus
  - Comunicaciones
  - Otras

### **Datos de histórico de incidencia**

- Incidencia-ID
- Fecha y hora
- Cliente-ID
- Dirección-ID
- Incidencia-ID (Tipo)
- Prioridad-ID (Tipo)
- Dispositivo-ID
- Aplicativo-ID
- Descripción de la incidencia manual
- Estado
  - Abierta
  - En resolución
  - Pendiente pieza del fabricante o proveedor
  - Cancelada
  - Cerrada
- Fecha último cambio estado
- Agente-ID (persona asignada para su resolución)
- Tiempo estimado resolución
- Tiempo real de resolución

### **Datos de prioridad**

- Prioridad\_ID
- Tipo:
  - Normal
  - Baja
  - Alta
  - Urgente

### **Datos de comentarios en incidencias**

- Incidencia-ID
- Fecha y hora
- Contacto-ID / Agente-ID
- Comentario-ID (Tipo)
- Comentario

## **2.7. Procedimiento de captura de incidencia, seguimiento y cierre**

**2.7.1.** Los procesos de captura son los siguientes:

### **Datos requeridos en la captura de una incidencia telefónica**

- El agente de atención al cliente crea una nueva incidencia.
- Localiza el cliente, dirección y contacto
- Selecciona un técnico que supervisará la incidencia.
- Selecciona el dispositivo involucrado en la incidencia.
- Selecciona el aplicativo involucrado.
- Fecha y hora de la creación
- Minutos de atención telefónica.
- Tipo de problema y la prioridad de la misma.
- Fecha mínima y máxima para la solución (calculado automáticamente mediante el control de fechas **(RF 3.236)**)
- Inserta el título y redacta el problema en detalle.
- Si es una incidencia breve y queda resuelta se creará una incidencia rápida o nota, siendo el usuario creador de la incidencia el mismo que el supervisor. La incidencia pasará a estado cerrado. Si no quedara resuelta, se crea con normalidad y pasaría a la atención del técnico supervisor seleccionado quedando la incidencia en estado “abierta”.

### **Datos requeridos al cliente para enviar una incidencia**

- Estar logueado identifica cliente, dirección y contacto.
- Escribir el título de la incidencia
- Redactar la incidencia y enviar. El sistema crea la incidencia.
- El agente encargado de recibirla revisará los datos si es necesario.

- 2.7.2.** Si la incidencia está abierta el usuario, que como mínimo es un técnico, irá resolviendo la incidencia e informando de su estado además de redactar la información relevante que considere tanto para los demás compañeros como para registrar algún conocimiento al respecto.
- 2.7.3.** Si fuera necesario otro usuario de rol técnico como mínimo, podrá participar en la resolución de dicha incidencia. Pudiendo modificar el responsable de la incidencia si fuera necesario.
- 2.7.4.** Una vez resuelta la incidencia. Si el usuario es un técnico, marcará el estado de “cerrada”. El sistema mostrará un recuadro donde tiene que redactar la solución de cara al cliente. Igualmente se mostrará la lista de marcadores o tags para que facilite, si ve que es necesario, información. Tendrá la opción de previsualizar el parte en formato pdf, cuando considere que está correcto tendrá una opción de “informar al jefe técnico” que es el responsable de revisar y verificar el parte y el encargado de enviar el parte al cliente vía email.
- 2.7.5.** Todo evento de cerrar la incidencia y el envío, por parte del jefe técnico, del parte de trabajo al cliente, quedará registrado en comentarios automáticos dentro de la incidencia para información de todos.
- 2.7.6.** Los únicos que pueden enviar un parte de trabajo son los usuarios que como mínimo tienen el rol de jefe técnico.
- 2.7.7.** El parte se podrá previsualizar y reenviar siempre.
- 2.7.8.** A pesar de que una incidencia esté cerrada, seguirá siendo accesible y cualquier empleado podría añadir un comentario.
- 2.7.9.** Toda incidencia cerrada puede ser reabierta. Mínimo hay que ser usuario con rol de técnico.

## **3. Sistema de control**

### **3.1. Control de acceso de usuario**

- 3.1.1.** El sistema tiene que gestionar un control de acceso al sistema mediante usuario y contraseña.
- 3.1.2.** En el proceso de alta en el sistema, el usuario será el nombre del email y su clave la que haya definido.
- 3.1.3.** En caso de olvidar su contraseña, en el login, podrá solicitar “recordar contraseña” y el sistema le enviará un email recordándoselo, a su nombre de usuario

- 3.1.4.** La contraseña tendrá que tener un tamaño mínimo de 5 caracteres alfanuméricos.
- 3.1.5.** Con el fin de facilitar el acceso, el sistema utilizará una cookie que recuerde el último usuario y contraseña logueado con éxito.

### **3.2. Estimación de tiempos (Tipo incidencia-Prioridad)**

- 3.2.1.** En el proceso de creación de una nueva incidencia, el sistema facilitará el registro mediante el cálculo automático de la fecha estimada para la solución y resolución de dicha incidencia. Se calcula tomando los datos de la regla de supervisión Calendario lectivo (requisito funcional 1.5.3) y Tiempos en función de tipo de incidencia y prioridad de la misma (requisito funcional 1.5.4) donde se obtiene el calendario lectivo y el tiempo estimado por la empresa (o por contrato de mantenimiento). El sistema localiza el tiempo mínimo y máximo para una solución y resolución. Estas fechas se incluyen en la incidencia y el sistema de conocimiento hará uso de ellas para notificar si están en vigor o han caducado, para organizar por prioridad o tiempo de caducidad, etc.

### **3.3. Captura e identificación de llamadas entrantes**

- 3.3.1.** Hay que capturar e identificar el número de llamada entrante proveniente de la centralita telefónica virtual (PBX) contratada por Kernel Informática S.L para su posterior procesamiento.
- 3.3.2.** El número debe de ser localizado en la base de datos, para identificar a un posible cliente.
- 3.3.3.** Cuando detecta que es un cliente tiene que extraer información relevante del mismo para que el usuario de atención al cliente esté informado de su estado.
- 3.3.4.** La información será: Nombre del cliente, resumen de incidencias no cerradas y descripción o notas informativas redactadas en la ficha del cliente.

## **4. Sistema de mantenimiento**

### **Gestión de mantenimiento software**

- 4.1.1.** Los empleados exclusivamente tendrán acceso a esta gestión.
- 4.1.2.** El sistema software generará logs que faciliten el mantenimiento y depuración:
- 4.1.2.1. Mantenimiento:
- Se controlará el login de los usuarios de modo que los administradores del sistema puedan controlar la entrada y salida del sistema de los usuarios.
- 4.2.1.2. Se registrará eventos del sistema que faciliten la depuración, el log lo podrá reiniciar el usuario que como mínimo tenga rol de técnico:
- Comportamientos erróneos del software.
  - Registro de eventos críticos o relevantes.

## Definición de los Actores

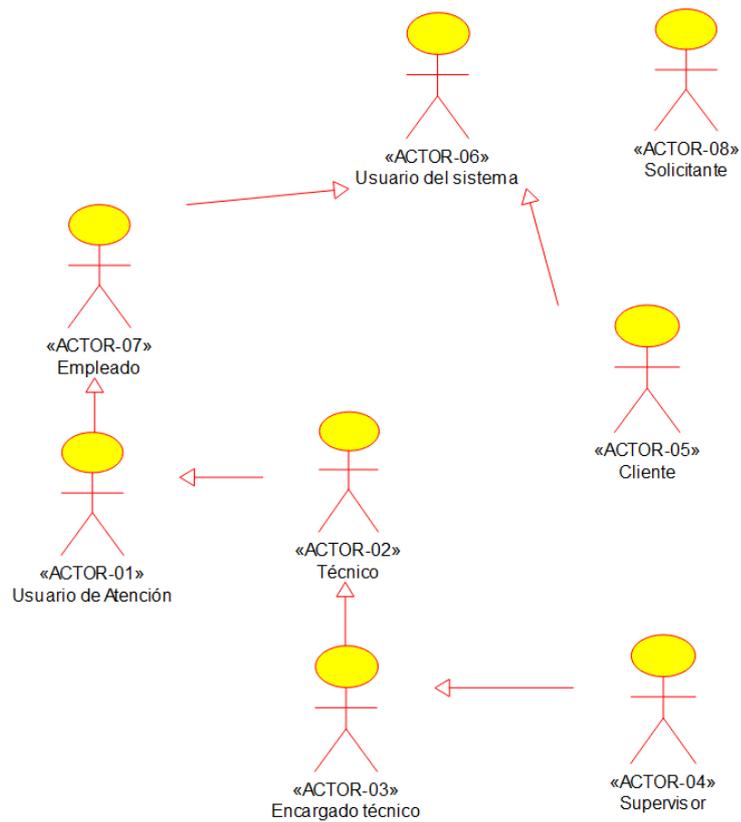


Ilustración 7: Jerarquía de actores

<b>ACTOR-01</b>	<b>Usuario de atención</b>
<b>Versión</b>	2.0 ( 04/11/16 )
<b>Autores</b>	Francisco Javier Montero Vega
<b>Fuentes</b>	Norberto Navarro Esposito
<b>Rol</b>	Hace referencia al empleado de Kernel Informática S.L que captura, atiende la llamada del cliente y realiza gestiones básicas.
<b>Comentarios</b>	<ul style="list-style-type: none"> <li>• Tiene que tener correctamente cumplimentado sus datos de empresa.</li> <li>• Tiene habilidad comunicativa de cara al cliente.</li> <li>• Su prioridad es cuidar la imagen de la empresa, ofreciendo el mejor servicio.</li> <li>• Tiene la posibilidad de resolver pequeños problemas cotidianos, de fácil resolución. Esto permitirá la mejora de gestión de tiempo.</li> </ul>

<b>ACTOR-02</b>	<b>Técnico</b>
<b>Versión</b>	2.0 ( 04/11/16 )
<b>Autores</b>	Francisco Javier Montero Vega
<b>Fuentes</b>	Norberto Navarro Esposito
<b>Rol</b>	Hace referencia a los empleados de Kernel Informática S.L que heredan todas las funciones del ACTOR-01. Ejecutan las tareas adecuadas para la resolución de incidencias. Bajo la supervisión del ACTOR-03. Administran los recursos del servicio técnico.
<b>Comentarios</b>	<ul style="list-style-type: none"> <li>• Tiene que tener correctamente cumplimentado sus datos de empresa.</li> <li>• Dispone de conocimientos avanzados de resolución de problemas informáticos, tanto software y hardware.</li> <li>• Su lugar de trabajo son: su puesto del departamento técnico dentro de Kernel Informática S.L y sus visitas a los clientes.</li> </ul>

<b>ACTOR-03</b>	<b>Encargado Técnico</b>
<b>Versión</b>	2.0 ( 04/11/16 )
<b>Autores</b>	Francisco Javier Montero Vega
<b>Fuentes</b>	Norberto Navarro Esposito Armando
<b>Rol</b>	Hace referencia a los empleados de Kernel Informática S.L que heredan todas las funciones del ACTOR-02. Su gestión del departamento está supervisado por el ACTOR-04. . Las funciones adicionales con respecto a un técnico son: <ul style="list-style-type: none"> <li>▪ Controlar el flujo de trabajo dentro del departamento, ordenando tareas al resto de técnicos.</li> <li>▪ Es el responsable directo del departamento.</li> <li>▪ Encargado de verificar y enviar el parte de trabajo al cliente.</li> </ul>
<b>Comentarios</b>	<ul style="list-style-type: none"> <li>• Tiene que tener correctamente cumplimentado sus datos de empresa.</li> <li>• Solo puede haber un encargado técnico por departamento.</li> <li>• Tiene que estar informado de toda la actividad del departamento para facilitar su gestión.</li> </ul>

<b>ACTOR-04</b>	<b>Supervisor</b>
<b>Versión</b>	2.0 ( 04/11/16 )
<b>Autores</b>	Francisco Javier Montero Vega
<b>Fuentes</b>	Norberto Navarro Esposito
<b>Rol</b>	<p>Hereda todas las funciones de gestión y administración del ACTOR-03.</p> <p>Hace referencia a los directivos de Kernel Informática S.L, encargados de gestionar y administrar todos los recursos humanos, materiales y documentales.</p>
<b>Comentarios</b>	<ul style="list-style-type: none"> <li>• Tiene que tener correctamente cumplimentado sus datos de empresa.</li> <li>• Ejemplo de supervisor: El gerente y el Director.</li> <li>• Constantemente monitorizan la salud de la empresa.</li> <li>• Son los responsables de todas las decisiones importantes.</li> <li>• Tienen acceso a toda la funcionalidad del sistema software.</li> </ul>

<b>ACTOR-05</b>	<b>Cliente</b>
<b>Versión</b>	1.0 ( 04/11/16 )
<b>Autores</b>	Francisco Javier Montero Vega
<b>Fuentes</b>	Norberto Navarro Esposito
<b>Rol</b>	Hace referencia a los clientes registrados en Kernel Informática S.L
<b>Comentarios</b>	<ul style="list-style-type: none"> <li>• Tienen que tener correctamente cumplimentado sus datos en el sistema.</li> <li>• Todo cliente tiene un contrato de mantenimiento con Kernel Informática S.L</li> </ul>

<b>ACTOR-06</b>	<b>Usuario</b>
<b>Versión</b>	1.0 ( 04/11/16 )
<b>Autores</b>	Francisco Javier Montero Vega
<b>Fuentes</b>	Norberto Navarro Esposito
<b>Rol</b>	El actor hace referencia a todo usuario del aplicativo. Pudiendo ser: ACTOR-01, ACTOR-02, ACTOR-03, ACTOR-04, o bien, ACTOR-05.
<b>Comentarios</b>	Ninguno

<b>ACTOR-07</b>	<b>Empleado</b>
<b>Versión</b>	1.0 ( 04/11/16 )
<b>Autores</b>	Francisco Javier Montero Vega
<b>Fuentes</b>	Norberto Navarro Esposito
<b>Rol</b>	Hará referencia exclusiva a todo empleado de Kernel Informática S.L, que esté registrado en el sistema.
<b>Comentarios</b>	<ul style="list-style-type: none"> <li>• Los empleados se distribuyen en los diferentes departamentos de Kernel Informática S.L.</li> </ul>

<b>ACTOR-08</b>	<b>Solicitante</b>
<b>Versión</b>	1.0 ( 04/11/16 )
<b>Autores</b>	Francisco Javier Montero Vega
<b>Fuentes</b>	Norberto Navarro Esposito
<b>Rol</b>	Hace referencia a un cliente de Kernel Informática S.L que aún no tiene un usuario para acceder al portal web, pero tiene intención de hacerlo poniéndose en contacto con Kernel Informática S.L.
<b>Comentarios</b>	<ul style="list-style-type: none"> <li>• El solicitante tiene que ser cliente de Kernel Informática S.L</li> <li>• El Solicitante no es un usuario del sistema.</li> <li>• El solicitante puede solicitar el alta de su usuario por dos vías: Registro a través de la web, o bien, por solicitud directa a un administrador.</li> </ul>

## Casos de uso



Ilustración 8: Casos de uso del sistema SRIC

## Tarjetas CRC ([Booch06], 2006)

ID Caso uso:	<b>CU-02</b>		
Nombre Caso uso:	<b>Solicitar alta</b>		
Creado por:	Francisco Javier Montero Vega	Última actualización por:	Francisco Javier Montero Vega
Fecha Creación:	10-11-16	Fecha última actualización:	23-03-17

Actores:	Solicitante, Supervisor	
Descripción:	Este proceso genera una nueva ficha de usuario en el sistema.	
Se inicia cuando:	<ul style="list-style-type: none"> <li>• Un cliente solicita mediante el sistema web, su registro en el sistema</li> <li>• El supervisor tiene la información del futuro usuario. La analizará y decidirá si va a ser registrada en base a las reglas de negocio que estime oportunas.</li> </ul>	
Precondición:	<ol style="list-style-type: none"> <li>1. El formulario de alta, tiene que estar correctamente cumplimentado. Siendo obligatorio tener cumplimentado los datos de carácter obligatorio: Nombre, apellido, DNI y correo electrónico.</li> <li>2. El nombre de su email tiene que contener solo caracteres válidos para el formato de email.</li> <li>3. El solicitante tiene que facilitar un correo electrónico que esté operativo, al cual tenga acceso y pueda leer los mensajes.</li> <li>4. La contraseña tiene que tener un tamaño mínimo de 5 caracteres. Exclusivamente alfanuméricos.</li> </ol>	
Postcondición:	<ol style="list-style-type: none"> <li>1. El solicitante será registrado como usuario del sistema.</li> </ol>	
Flujo normal:	SOLICITANTE	SUPERVISOR
	<ol style="list-style-type: none"> <li>1. El solicitante accede al portal web desde su navegador. Solicita el alta en el sistema rellenando el formulario web.</li> </ol>	
	<ol style="list-style-type: none"> <li>2. Envía la solicitud y espera a su aceptación por correo electrónico.</li> </ol>	<ol style="list-style-type: none"> <li>3. Recibe la solicitud por email y queda informado de que está pendiente de su validación. Mientras tanto el usuario está registrado no está validado pero puede empezar a enviar incidencias.</li> </ol>

		4. El sistema envía un email indicando que está registrado. Indicando su usuario y contraseña.
	5. Lee el email donde informa de su usuario y contraseña.	
Flujo alternativo:	<ul style="list-style-type: none"> <li>• Ninguno</li> </ul>	
Excepciones:	El solicitante puede utilizar un email que ya esté registrado en el sistema. Por lo tanto el sistema antes de pasar al punto 3, tiene que verificar que esto no ocurra. Si es así, tendrá que avisar al solicitante, indicando que ya existe.	
Casos de uso incluidos:	- Registrar alta usuario	
Prioridad:	Alta	
Frecuencia:	Media	
Reglas de negocio:	- El supervisor indicará por una regla al sistema, un número máximo de registros de usuario no verificados por el administrador. Por seguridad, se evita un número ilimitado de altas.	
Requisitos especiales:	<ul style="list-style-type: none"> <li>- Tiene que ser cliente de Kernel Informática S.L, en caso contrario el administrador no verifica el usuario y lo puede eliminar.</li> <li>- Cuando llegue al sistema una solicitud web, el sistema informará al supervisor inmediatamente indicando que tiene que verificar la solicitud.</li> </ul>	
Supuestos:	- En el caso de que un usuario se de alta y envíe una incidencia, el sistema notificará visualmente al empleado que contacte con el supervisor.	
Notas y cuestiones:	-	

ID Caso uso:	<b>CU-15</b>		
Nombre Caso uso:	<b>Atender cliente telefónico</b>		
Creado por:	Francisco Javier Montero Vega	Última actualización por:	Francisco Javier Montero Vega
Fecha Creación:	10-11-16	Fecha última actualización:	15-03-17

Actores:	Usuario de atención, técnico, jefe técnico y administrador.	
Descripción:	El cliente llama a Kernel informática S.L y el recepcionista captura la llamada, escucha el problema del cliente y registra el caso en el sistema mediante una incidencia o nota.	
Se inicia cuando:	<ul style="list-style-type: none"> <li>El cliente se pone en contacto vía telefónica para notificar un problema.</li> </ul>	
Precondición:	<ol style="list-style-type: none"> <li>El usuario de atención al cliente tiene que estar logueado.</li> <li>Quien llama tiene que ser cliente con contrato de mantenimiento en vigor.</li> </ol>	
Postcondición:	La llamada queda registrada en el sistema a su vez que la incidencia o nota capturada.	
Flujo normal:	Cliente	Usuario de atención
	1. Llama a Kernel Informática S.L	
		2. Visualiza la llamada que identifica textualmente al cliente además con información adjunta.
		3. Descuelga y atiende
	4. Expone su problema	
		5. Registra en el sistema el problema y la actuación.
Flujo alternativo:	<ol style="list-style-type: none"> <li>El cliente llama para realizar una consulta o dejar un mensaje a un empleado.</li> <li>El usuario de atención lo registra con una nota.</li> </ol>	
Excepciones:	El cliente no ha pagado la cuota de mantenimiento mensual, por lo que únicamente se informa del caso al cliente y se toma nota de la incidencia. Este caso ha de ser informado con urgencia al supervisor.	

Casos de uso incluidos:	<ul style="list-style-type: none"> <li>- “Crear incidencia”, Que será del tipo “nota comunicativa”, o bien, pudiera llegar a ser de otro tipo.</li> <li>- Visualizar llamada: Capturada la llamada y mostrar la información del cliente, sus incidencias no cerradas y descripción, antes de descolgar.</li> <li>- “Consultar base de conocimiento”</li> </ul>
Prioridad:	Alta
Frecuencia:	Alta
Reglas de negocio:	<ul style="list-style-type: none"> <li>- Hay que atender a los clientes que pagan lo acordado en su contrato de mantenimiento. El sistema tiene que alertar visualmente. La incidencia se captura y hay que avisar al supervisor.</li> </ul>
Requisitos especiales:	Mientras el usuario de atención al cliente esta logueado y accede a la ficha del cliente se requiere que el sistema compruebe si el cliente ha pagado en el periodo de mes actual, su mantenimiento. Esto deriva a requisito no funcional.
Supuestos:	<ul style="list-style-type: none"> <li>- Si el cliente decide registrar la incidencia vía web, podrá hacerlo siempre que esté registrado en el sistema.</li> </ul>
Notas y cuestiones:	<p>Si la incidencia no queda resuelta durante la llamada o breve unidad de tiempo entonces:</p> <p>El problema no ha quedado resuelto y se transfiere al técnico.</p> <p>El problema ha quedado resuelto y puede crear una incidencia o nota que cerrará el sistema, no interfiere el técnico.</p>

ID Caso uso:	<b>CU-22</b>		
Nombre Caso uso:	<b>Resolver incidencia</b>		
Creado por:	Francisco Javier Montero Vega	Última actualización por:	Francisco Javier Montero Vega
Fecha Creación:	12-11-16	Fecha última actualización:	15-03-17

Actores:	Técnico, encargado técnico, supervisor	
Descripción:	La incidencia ha sido iniciada, está en estado “abierta” y se procede a resolverla tarea a tarea.	
Se inicia cuando:	<ul style="list-style-type: none"> <li>Una vez creada la incidencia, el responsable al cargo de la misma trabajará en su resolución. Todas las tareas e hitos alcanzados se añadirán a la incidencia vía comentario con o sin fichero adjunto. Durante el trabajo de resolución, la incidencia irá transitando por uno o varios estados.</li> </ul>	
Precondición:	<ol style="list-style-type: none"> <li>Haber iniciado sesión</li> <li>Tener información relevante para adjuntar en la incidencia vía comentario. (Procedimiento de tarea resuelta, comentarios técnicos, ...)</li> <li>Clasificar la información adjunta en la captura de la incidencia.</li> </ol>	
Postcondición:	La incidencia a medida que vaya resolviéndose, incrementa su número de comentarios. (Tareas técnicas, eventos, anotaciones, etc.)	
Flujo normal:	TÉCNICO	SISTEMA
	1. Accede a la incidencia.	
	2. Gestiona comentarios y transita por los estados	3. Por cada cambio de estado inserta un comentario automático.
	4. Cambia de estado “cerrado”	5. Inserta un comentario automático con el cambio de estado.
	6. Cumplimenta y envía el parte de trabajo finalizado	7. El sistema envía email con el parte.

		8. Inserta un comentario automático con el parte (informando al jefe técnico, o bien, informando al cliente)
Flujo alternativo:	<ul style="list-style-type: none"> <li>- Paso 6: Si es un técnico el sistema enviará el parte al jefe de servicio técnico para que lo valide y posteriormente él lo envíe al cliente por email.</li> <li>- Paso 6: Es un jefe técnico, cierra el parte de trabajo, lo revisa y directamente lo envía al cliente.</li> </ul>	
Excepciones:	Mientras los técnicos trabajan en resolver la incidencia, se puede dar el caso de cancelación, por orden del supervisor. Basándose en una decisión administrativa. En tal caso, la incidencia pasará a estado "cancelada" y se pasará a la siguiente incidencia en cola.	
Casos de uso incluidos:	<ul style="list-style-type: none"> <li>- Cambio de estado.</li> <li>- Editar incidencia</li> <li>- Añadir comentario</li> <li>- Enviar email</li> <li>- Crear parte de trabajo</li> </ul>	
Prioridad:	Alta	
Frecuencia:	Alta	
Reglas de negocio:	<ul style="list-style-type: none"> <li>- Todo empleado puede comentar en la incidencia desde el sistema de conocimiento, o bien, si tiene acceso, desde la ficha de la incidencia.</li> <li>- Puede cambiar el estado de la incidencia todos los empleados que mínimo sea técnico.</li> <li>- Las incidencias no se eliminan, se cancelan. El supervisor necesita datos de cuantas incidencias se crean, sus estados y todo dato que considere relevante.</li> </ul>	
Requisitos especiales:	<ul style="list-style-type: none"> <li>- Documentar todas las tareas realizadas en la resolución de la incidencia antes del cierre.</li> <li>- Añadir tags si contiene información que ayude al conocimiento.</li> <li>- Revisar que todos los problemas han quedado resueltos y dar constancia de las pruebas realizadas en el informe de la incidencia.</li> </ul>	
Supuestos:	-	
Notas y cuestiones:	-	

## Requisitos no funcionales

1. Para obtener una mejora de la accesibilidad, flexibilidad y movilidad, se decide el desarrollo de un software Web.
2. El entorno de desarrollo a utilizar será de la empresa PCSoft, concretamente la herramienta WebDev versión 21.
3. El software tiene que ser sencillo de mantener.
4. El sistema tiene que trabajar con los navegadores más comunes (Chrome, Internet Explorer y Firefox)
5. La Base de datos actual está en Visual Fox Pro, Por lo que hay que realizar la migración a la base de datos HFSQL de WebDev.
6. El cliente puede consultar el estado y datos relativos a sus incidencias pendientes, pero hay que restringir el acceso a comentarios internos de los empleados, asociados a sus incidencias. La información será filtrada para mostrarle solo lo que solicita.
7. Se establecerá un tiempo máximo de inactividad en el sistema, de modo que si un usuario que se haya logueado en el sistema, sobrepasa dicho tiempo de inactividad, el sistema automáticamente le cerrará la sesión. Razón de seguridad para evitar el hackeo del sistema.
8. La resolución de pantalla mínima será de 1024x768 pixeles.
9. El servidor debe de ser compatible con la plataforma PCSoft WebDev.
10. El proyecto debe simplificar las tareas de mantenimiento del propio sistema.
11. Debe de contener mecanismos de seguridad que eviten el uso malintencionado del software.
12. El proyecto debe de cumplir con la ley oficial de protección de datos, acorde con la ley orgánica 15/1999 del 13 de diciembre (nota: el servidor no puede estar fuera de Europa).
13. El sistema ha de informar al usuario de atención al cliente, si el cliente al que va a atender su problema no ha pagado su cuota mensual. Tanto si el cliente se ha puesto en contacto vía telefónica o por el portal web (Portal web, el sistema software no reconocerá email, es la encargada de recepción, principal actor que atiende al cliente, el que lee el correo con regularidad, por lo que el software de correo de la empresa tiene que notificar al empleado).
14. Sistema tiene que facilitar información al gerente de los usuarios que no hayan pagado la cuota mensual de mantenimiento.
15. Proceso de baja del sistema: El supervisor tendrá presente la ley actual de protección de datos.
16. Todos los informes que se generan siempre serán al momento, y solo se guardarán temporalmente en el servidor hasta que finalice la tarea de visualizarlo o descargarlo. La razón son motivos de seguridad a estar desactualizado y por razones de espacio.

17. Respecto al caso de uso "Descarga uniforme" derivan:
  - a. El navegador debe de estar configurado, de modo que si no tiene predefinido una carpeta de descarga, deberá mostrar un dialogo donde el usuario seleccione la carpeta de descarga.
  - b. En la máquina local tendrá que tener instalado algún software que permita abrir el fichero PDF.
18. Respecto a los informes:
  - a. El navegador ha de ser compatible para visualización PDF. Actualmente hay pocos que no lo sean.
  - b. El PDF tiene que tener un formato listo para ser impreso a papel, teniendo el logo de la empresa, todos los datos identificativos (identificadores, fechas, etc...). Además de todos los detalles.
19. Todo informe del sistema ha de tener dos partes diferenciadas:
  - a. En la parte superior irá la cabecera donde tendrá todos los datos que guarden relación con el documento.
  - b. En la parte inferior y posteriores páginas se indicaran todos los datos referentes a los detalles del informe.
20. Se dan dos requisitos no funcionales derivados del requisito funcional "cerrar incidencia" :
  - a. Generar un informe de parte para el cliente. Para ello se mostrará la información abreviada:
    - i. Cabecera de datos de la incidencia
    - ii. Tareas realizadas y sus tiempos dedicados.
  - b. El sistema ante el cierre de incidencia tiene que enviar un email al encargado técnico para que este quede informado. Tiene que adjuntarse el informe de incidencia y el de parte, ambos en formato PDF y que no superen un tamaño máximo de 1Mb para facilitar el envío de email.
21. El sistema ha de crear una incidencia si al registrar una nota (el usuario de atención al cliente) se especifica :
  - a. Que esté sin resolver.
  - b. Ha llevado un largo tiempo resolverla y tiene que crearse un registro de incidencia.
22. La web seguirá un diseño web minimalista.
23. Se debe mantener una lógica en la navegación de la web cada vez que se pulse la tecla tabulador, principalmente en formularios, con el fin de facilitar el diseño minimalista.
24. En los controles más sensibles (datos obligatorios, información sensible, etc.) se precisará de un control adicional que verifique la correctitud de los datos insertados. Ejemplo: El campo de email tiene que ser correcto, siguiendo las normas estándar del formato email.

## Diagrama de Estado ([UML])

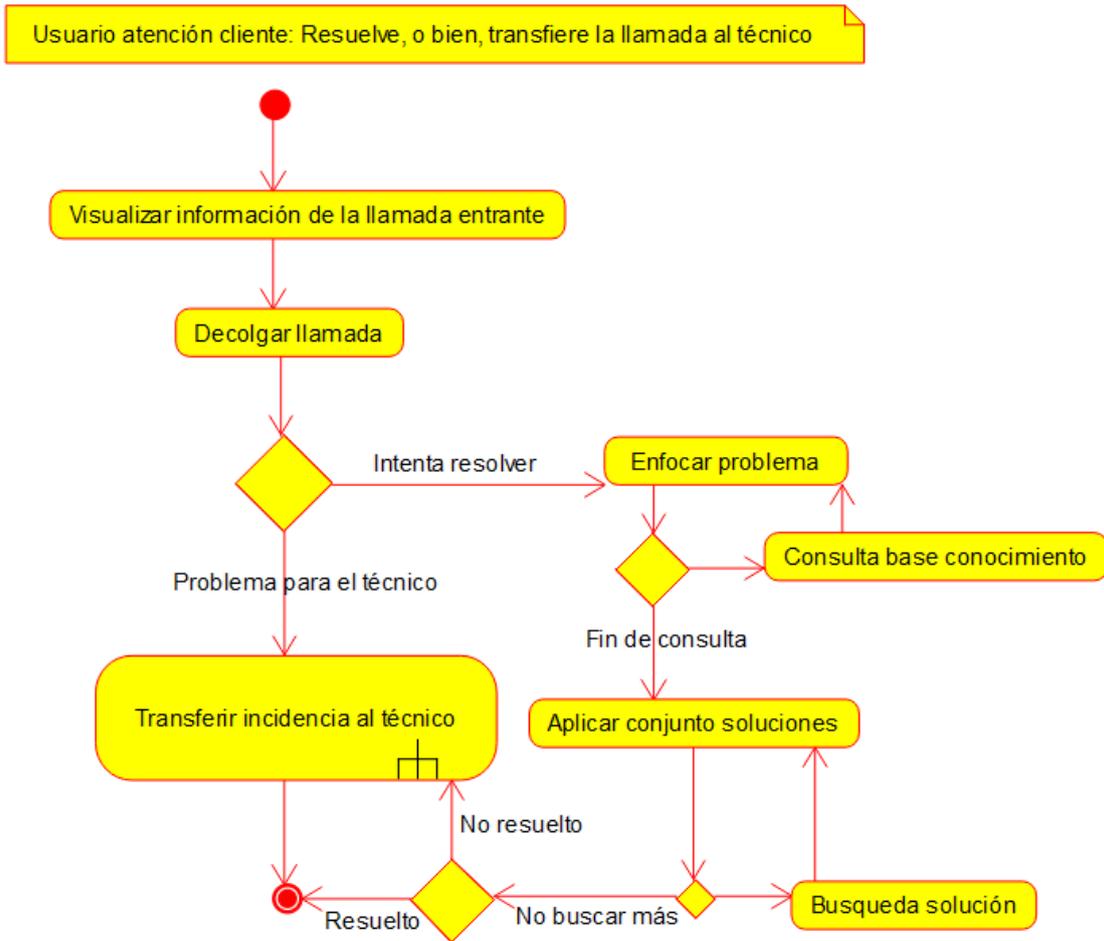
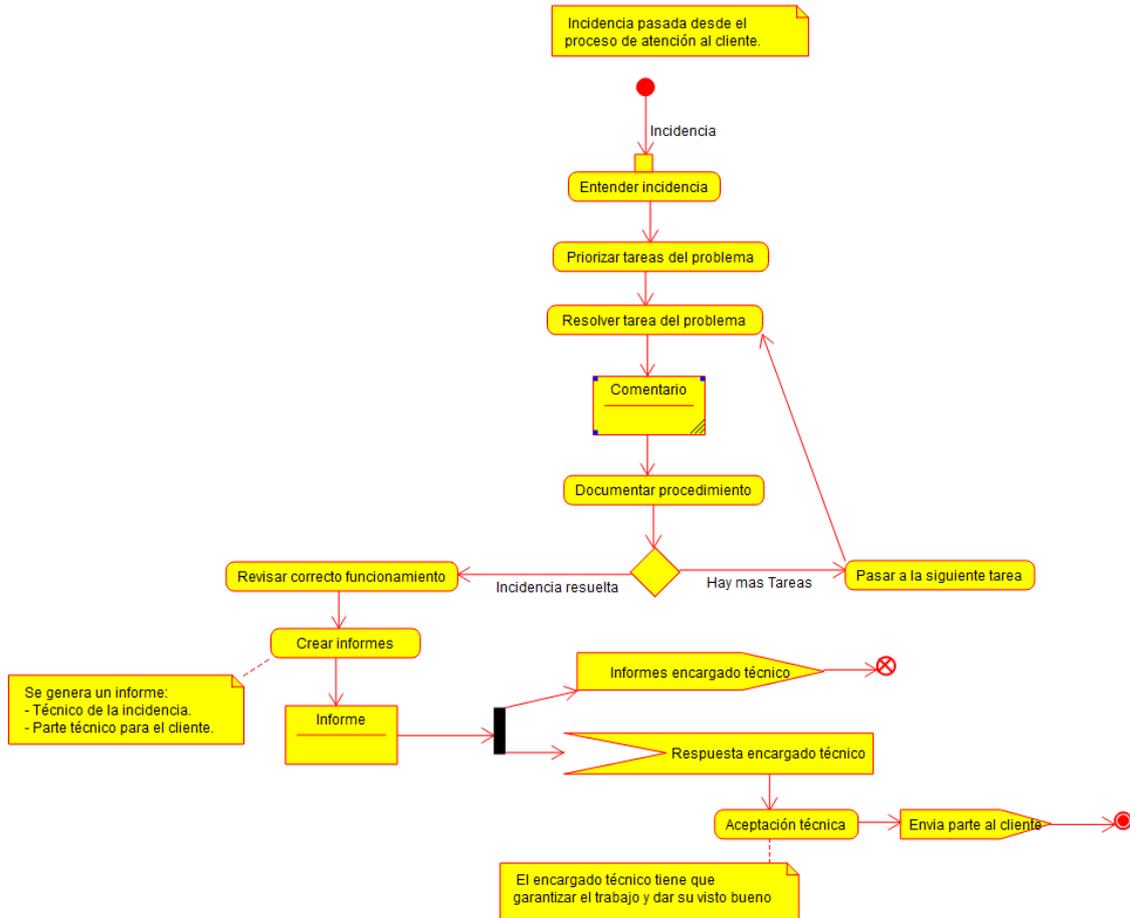
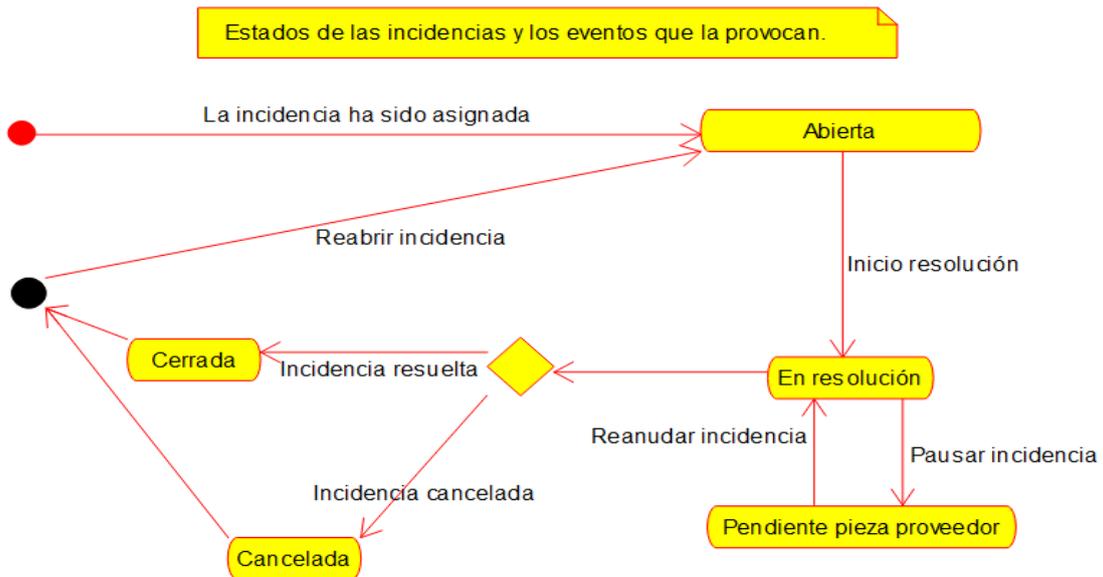


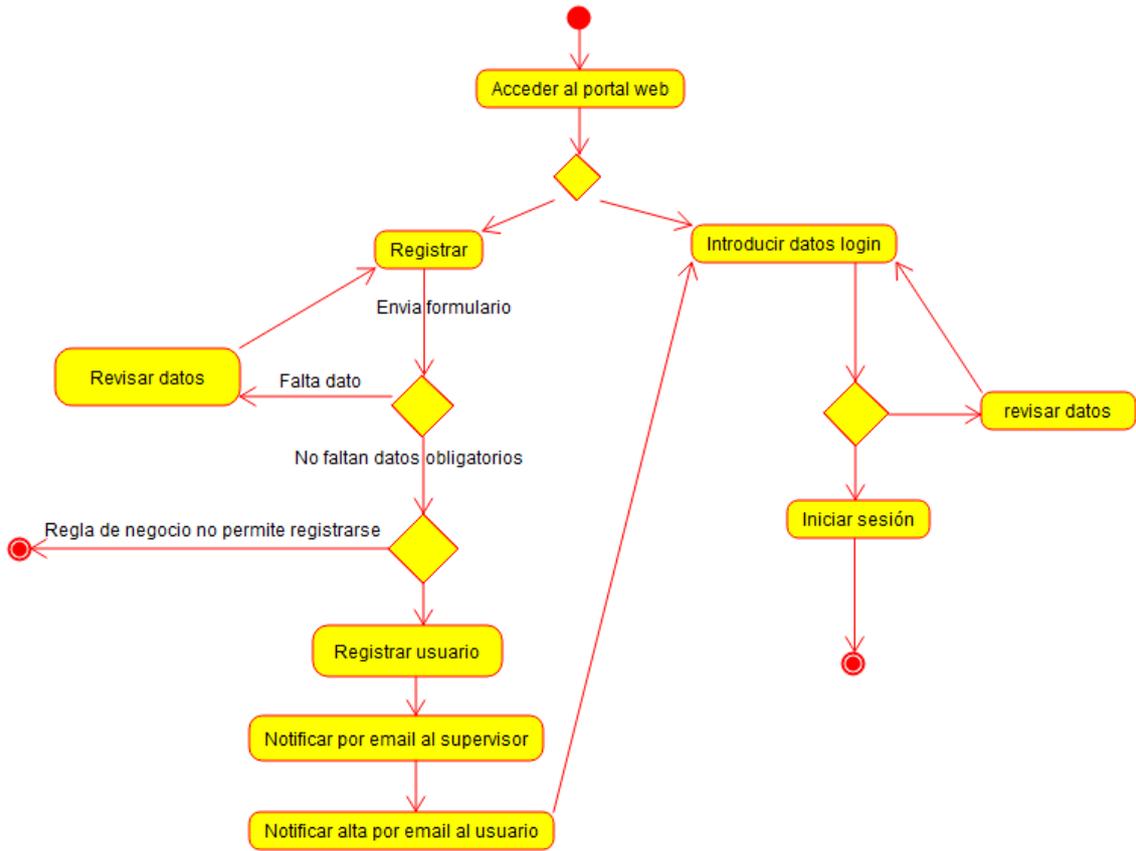
Ilustración 9: Diagrama de estado – Atención al cliente



**Ilustración 10: Diagrama de estado – Procesar incidencia**



**Ilustración 11: Estados de las incidencias**



**Ilustración 12: Registro y login de un cliente vía web**

## Diagrama de clase

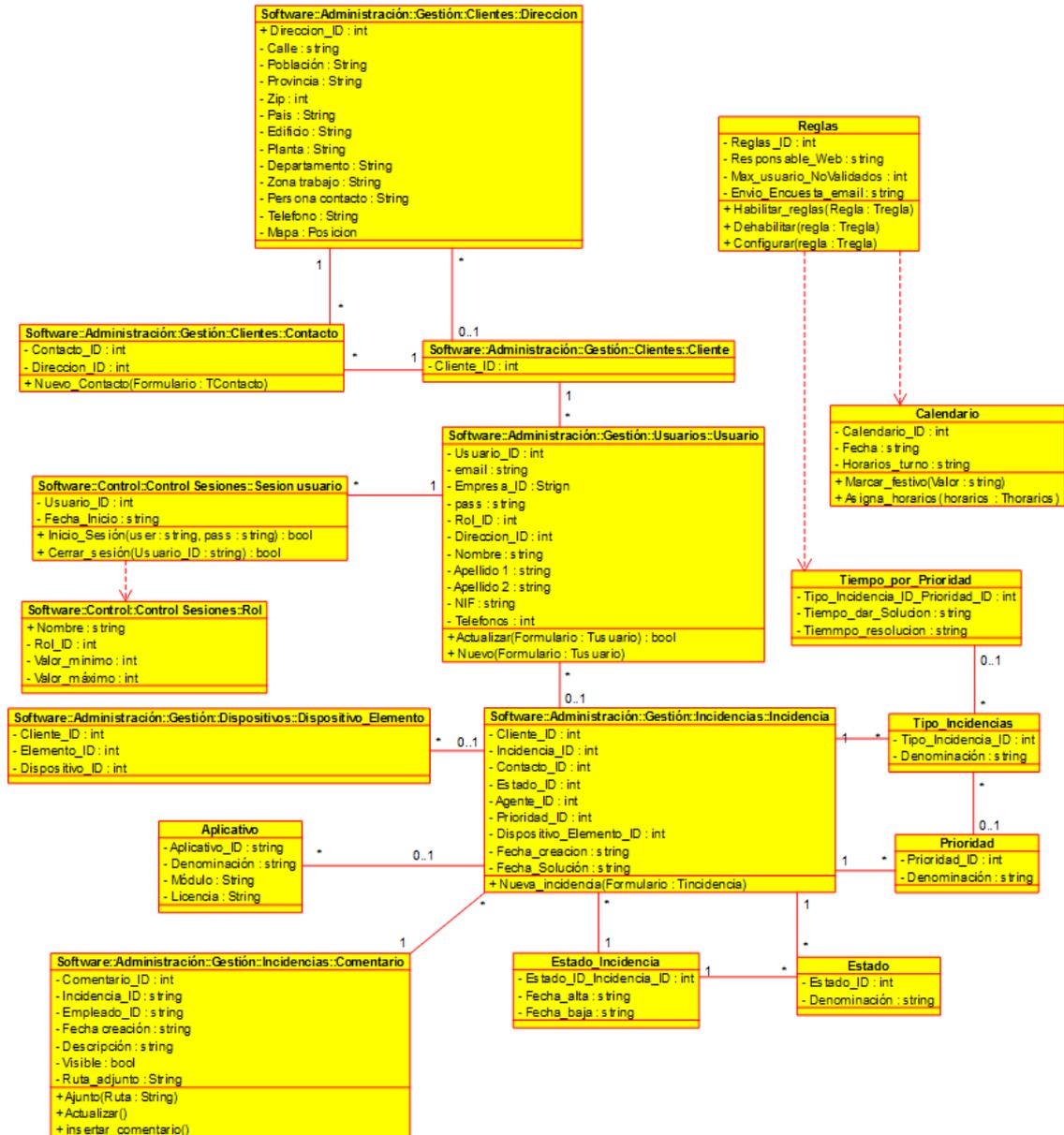


Ilustración 13: diagrama de clases del sistema SRIC

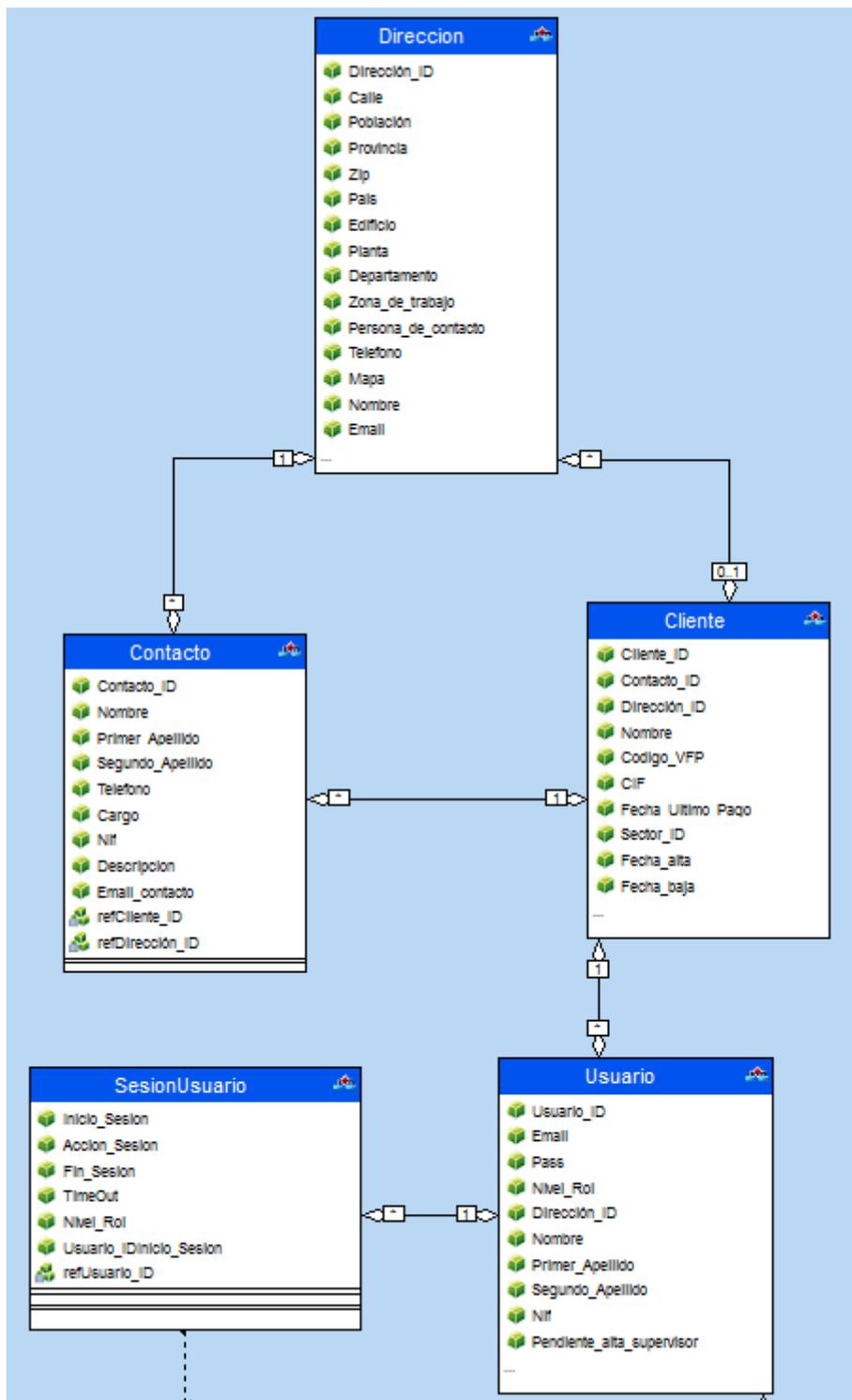


Ilustración 14: Vista detallada del diagrama de clases del sistema SRIC

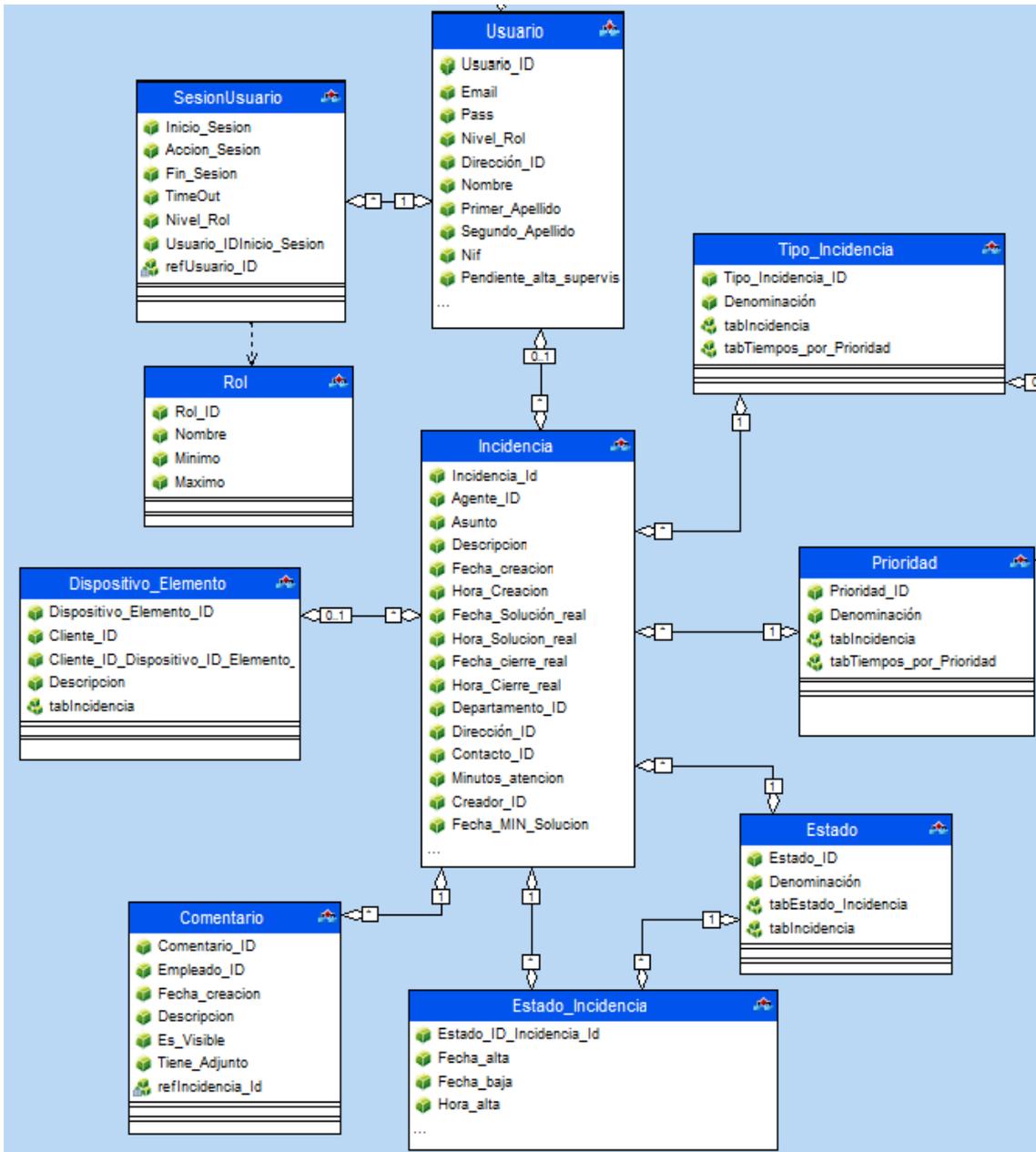


Ilustración 15: Vista detallada del diagrama de clases del sistema SRIC

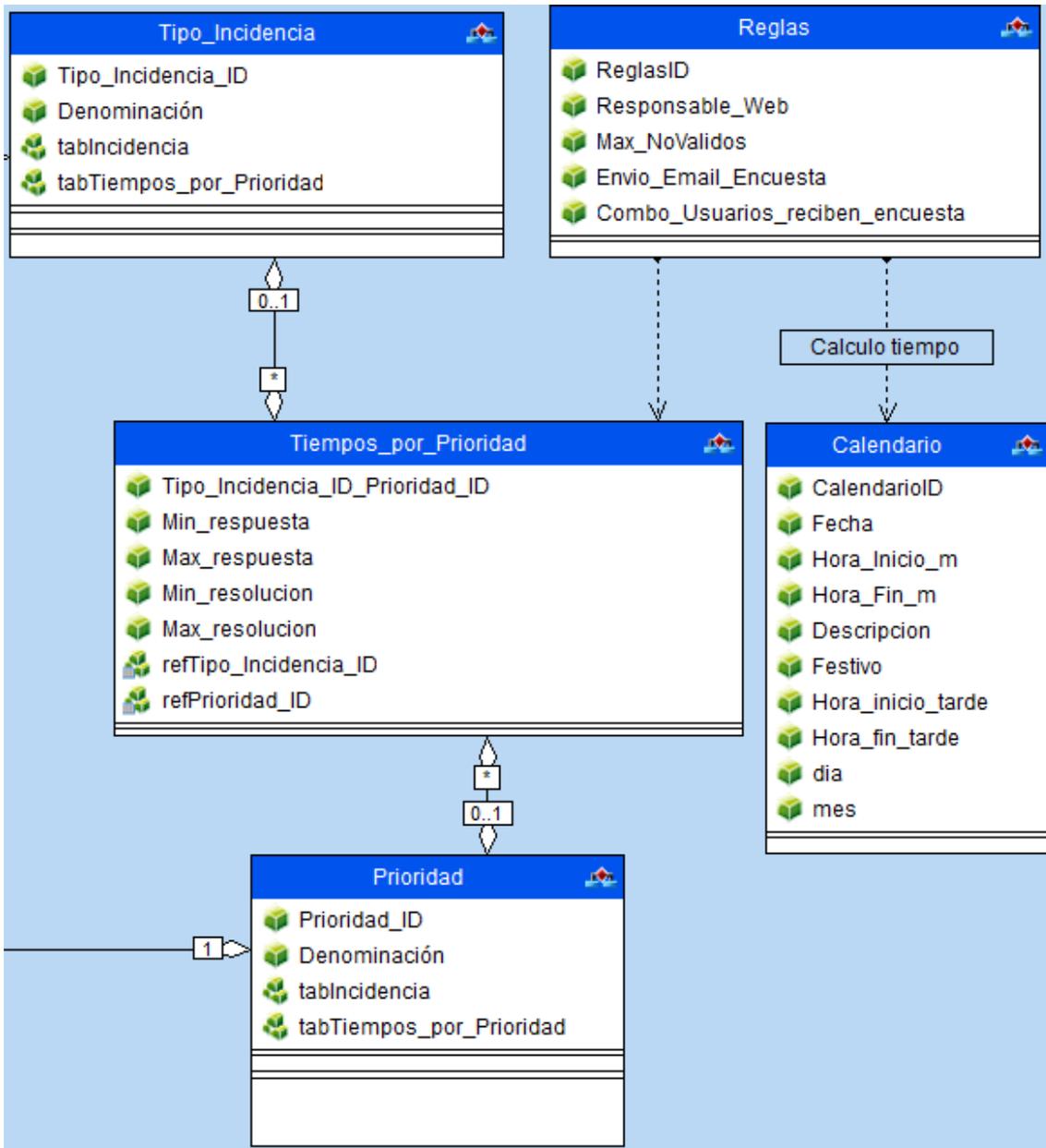
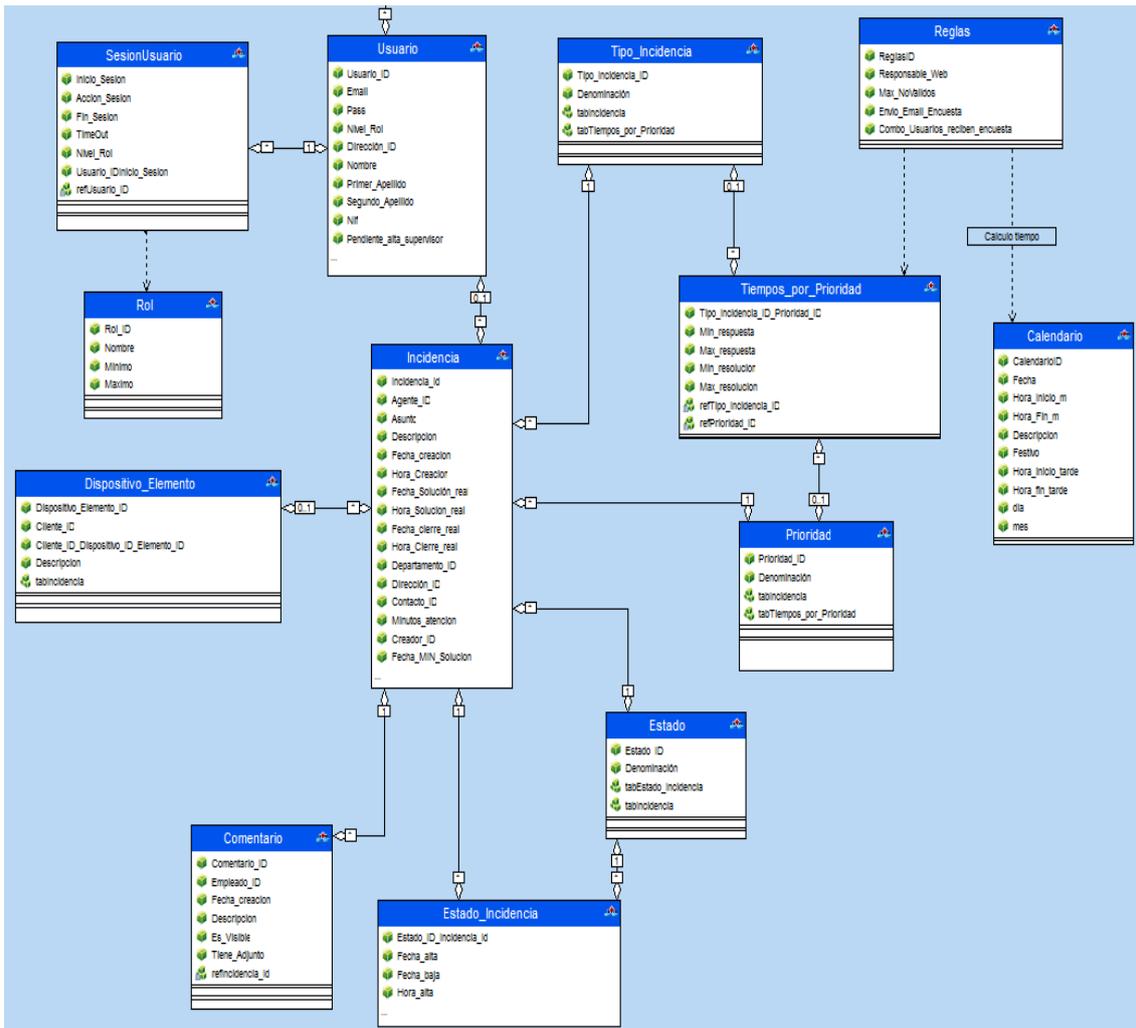


Ilustración 16: Vista detallada del diagrama de clases del sistema SRIC



**Ilustración 17: Vista global del diagrama de clases detallado del sistema SRIC**

## Identificación inicial del Webservice (Solución Centralita Virtual)

Para la solución de uno de los objetivos del proyecto, (página 16, apartado 10.3), se estudia la necesidad de un webservice para el procesamiento de las llamadas entrantes de los clientes de Kernel Informática S.L. Se elige un webservice debido a la interoperabilidad ([WS]) entre dos sistemas diferentes. El sistema Linux con Asterisk enviará la información de las llamadas y el webservice realizará el procesamiento de estas llamadas.

Este webservice se encargará de procesar las llamadas que el sistema Asterisk le pondrá en una carpeta compartida. Al llamar al webservice, comenzará el procesamiento de las llamadas, registro y actualización de su tabla de registro de llamadas y externamente el fichero RSS, que informa de las llamadas a todos los puestos de la intranet de Kernel Informática S.L.

Los requisitos funcionales serán los siguientes:

1. Asterisk facilitará las llamadas al webservice.
2. El webservice procesará todas las llamadas localizadas en una carpeta.
3. Compartirá su base de datos con SRIC.
4. Localmente registrará información de clientes a modo de agenda offline. Con el fin de seguir aportando información en las llamadas entrantes en caso de no tener acceso a la base de datos de SRIC.
5. Al finalizar el procesado de llamada, habrá hecho lo siguiente:
  - a. El registro de cada llamada en una base de datos compartida con SRIC.
  - b. Accederá a la base de datos de SRIC para obtener la información que necesite del cliente.
  - c. Actualizará los datos de las llamadas para que puedan acceder el aplicativo web SRIC.

Los requisitos no funcionales son:

1. El webservice tiene que estar conjunto con SRIC en el mismo servidor.
2. El acceso de SRIC a la información de las llamadas será a través de un RSS actualizado.
3. La base de datos local del webservice es accesible por SRIC con el fin de obtener el registro de llamadas. Tiene que tener permiso de acceso el aplicativo web de SRIC.

## Caso de uso del Webservice



Ilustración 18: Casos de uso del webservice

## Diagrama de flujo del Webservice

A continuación una explicación más global de la función del webservice.

El webservice se encuentra en el Servidor Windows. En el servidor Asterisk (Linux virtualizado) se captura la llamada entrante y acto seguido se intenta transferir al servidor Windows, pudiendo darse el caso de que haya o no conexión con la carpeta de red compartida entre ambos, tal como indica el único rombo de decisión de la Ilustración 19.

Si hay conexión se ha transferido la llamada, acto seguido se invoca al webservice, el cual procesará la llamada que está en la carpeta compartida. Primero localiza al cliente por su número de teléfono en la base de datos de SRIC. Procesa la llamada, actualiza el fichero RSS con toda la información del cliente e información relativa al mismo, tal como su descripción y mensajes de incidencias no cerradas, para que el aplicativo web SRIC muestre la llamada a los empleados de la oficina. Finalmente el webservice actualiza su registro de llamadas.

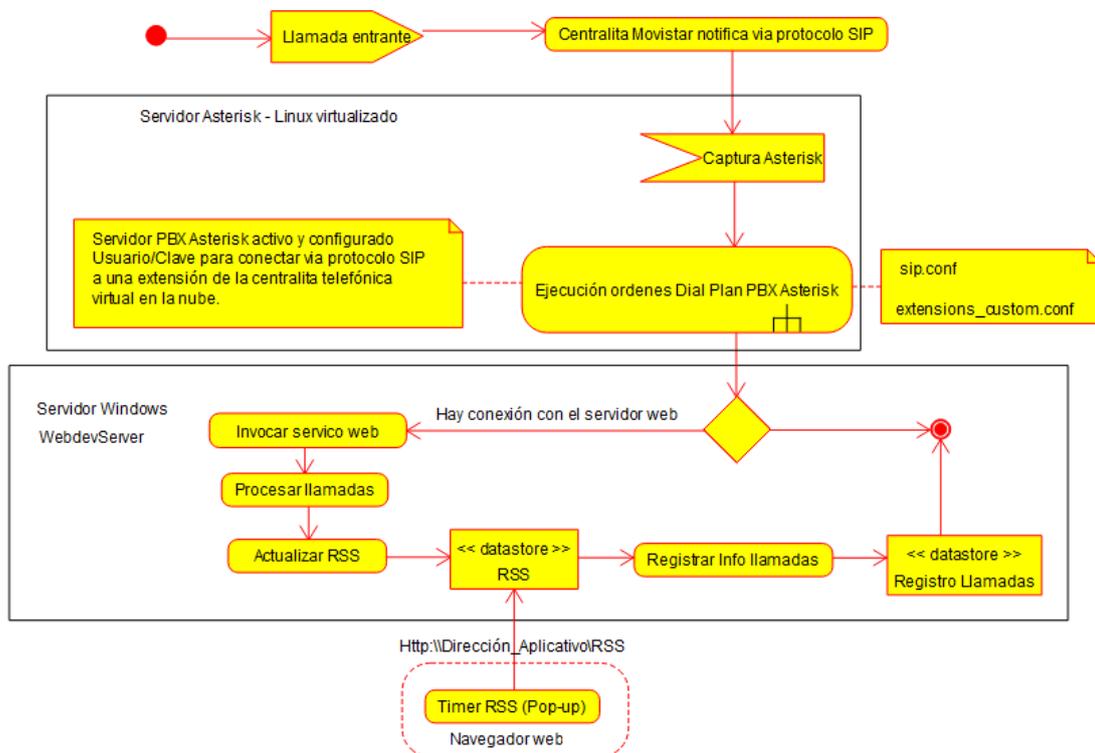


Ilustración 19: Comunicación entre servidor Linux (Asterisk) y Windows (webservice)

## Diagramas de clases del Webservice

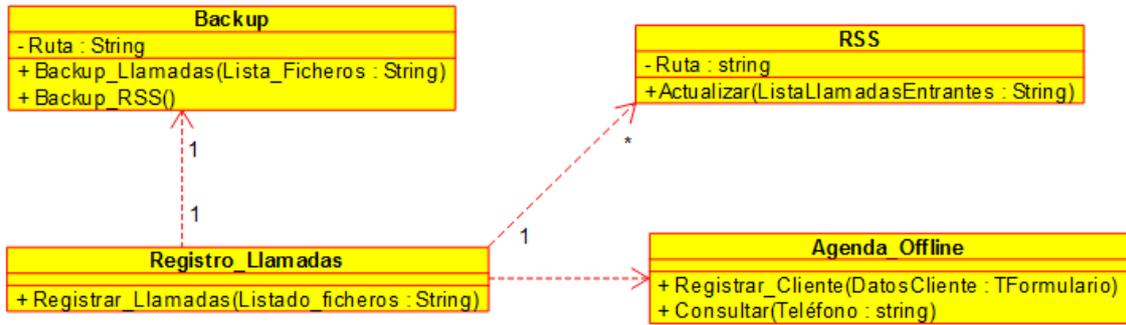
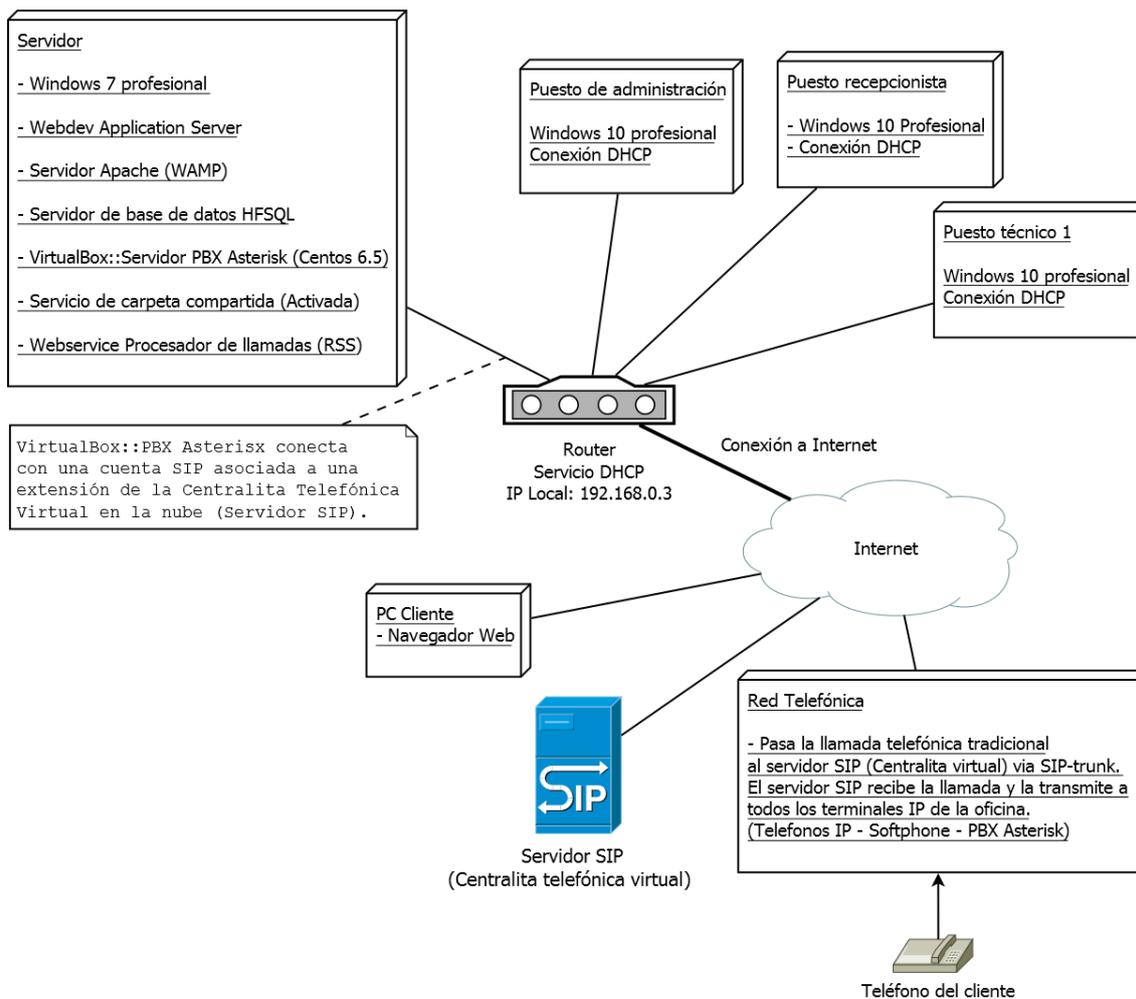


Ilustración 20 – Diagrama de clases del webservice

## Diseño de la arquitectura del sistema

A continuación se muestra un diagrama donde muestre gráficamente la solución hardware para este sistema:



**Ilustración 21: Diseño de la arquitectura del sistema**

Se pretende alojar en un servidor dedicado dentro de la red de Kernel Informática S.L los siguientes servicios:

- Servicio HFSQL de base de datos: Gestiona la base de datos WebDev.
- Webdev Application Server: Ejecuta el aplicativo desarrollado en código WLanguage.
- Wamp: Aplicativo que ejecuta el servicio Apache.
- Servidor PBX Asterisk: Centralita software instalada en un sistema operativo Linux virtualizado en Virtualbox.
- Compartición de carpetas: Servicio propio de Windows.

- **Webservice:** Servicio encargado de procesar las llamadas provenientes de Asterisk y actualizar el fichero RSS con sus datos.

El servidor se puede gestionar dentro de la empresa o bien, mediante conexión remota por terminal server si el administrador del servidor quisiera acceder al mismo.

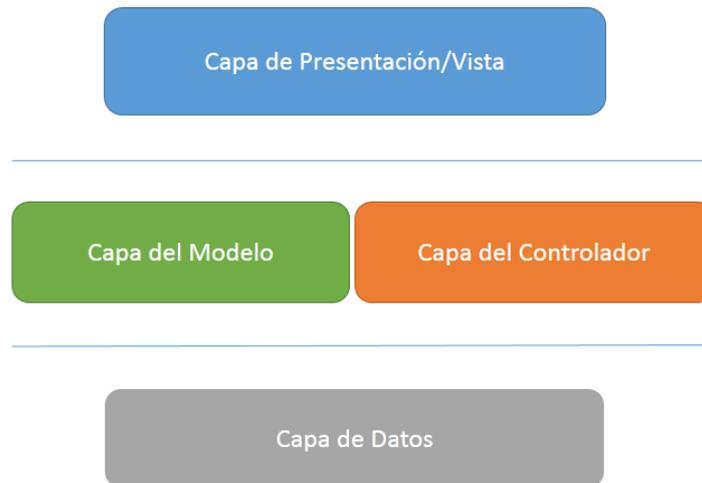
Los empleados accederán al sistema software mediante portal web dentro de la misma red local. Sin embargo los clientes accederán de igual modo pero conectando vía internet.

Los servicios del sistema estarán alojados en el único servidor dedicado, ya que tras simular el rendimiento general se considera inicialmente que la carga de trabajo no es elevada por lo que no se aprecia pérdida de rendimiento.

Se pretende dejar la puerta abierta a la posibilidad de trasladar el servidor local a uno que esté en la nube. Actualmente la empresa gestiona servidores dedicados en la nube. Esta posibilidad es factible ya que no se precisa de una configuración diferente.

## Diseño de capas

El diseño web tendrá las siguientes capas:



**Ilustración 22. Diseño de capas**

Las razones por la cual se ha decidido este diseño es:

1. Proporciona escalabilidad, capacidad de administración y utilización de recursos mejorados.
2. En cada capa hay un grupo de componentes que realizan una función específica.
3. Posibilidad de poder actualizar una capa con independencia de las otras.

**Capa de presentación/vista:**

Muestra los diferentes objetos gráficos, de cara al cliente: Botones, combos, etc. Será la encargada de mostrar la información y lógica de negocio proveniente de la capa de modelo.

**Capa de modelo:**

Representará la información y las reglas de negocio. Se encargará de devolver la información solicitada a la capa de presentación. Las peticiones de acceso o manipulación de la información llegan a esta capa proveniente de la capa del controlador.

**Capa del controlador:**

Se encargará de atender los eventos e invocar peticiones a la capa de modelo cuando se hace alguna solicitud sobre la información.

**Capa de Datos:**

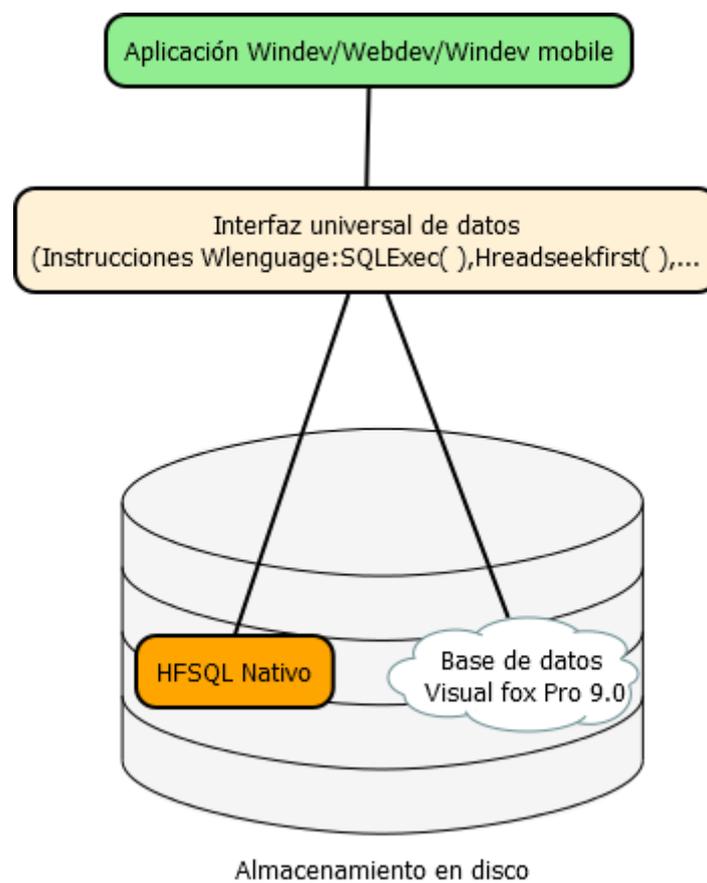
Encargada de albergar el sistema de gestión de base de datos.

## Diseño detallado

### Definición de sistemas, subsistemas y módulos

Sistema de base de datos:

Representado por el servidor HFSQL de PCSsoft.



**Ilustración 23: Diseño del servidor HFSQL**

## Sistema software SRIC

Representación del sistema SRIC:

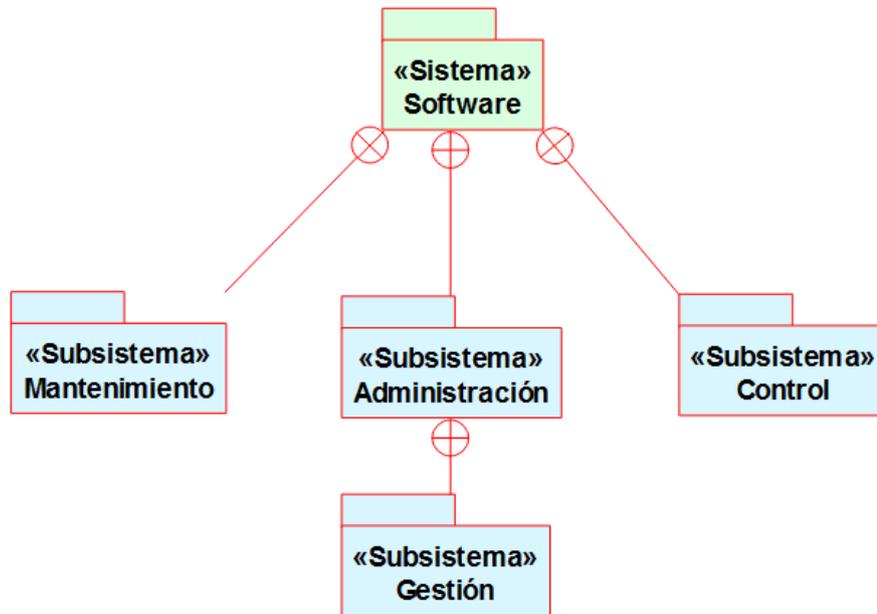


Ilustración 24: Sistema software

## Subsistema Software de SRIC

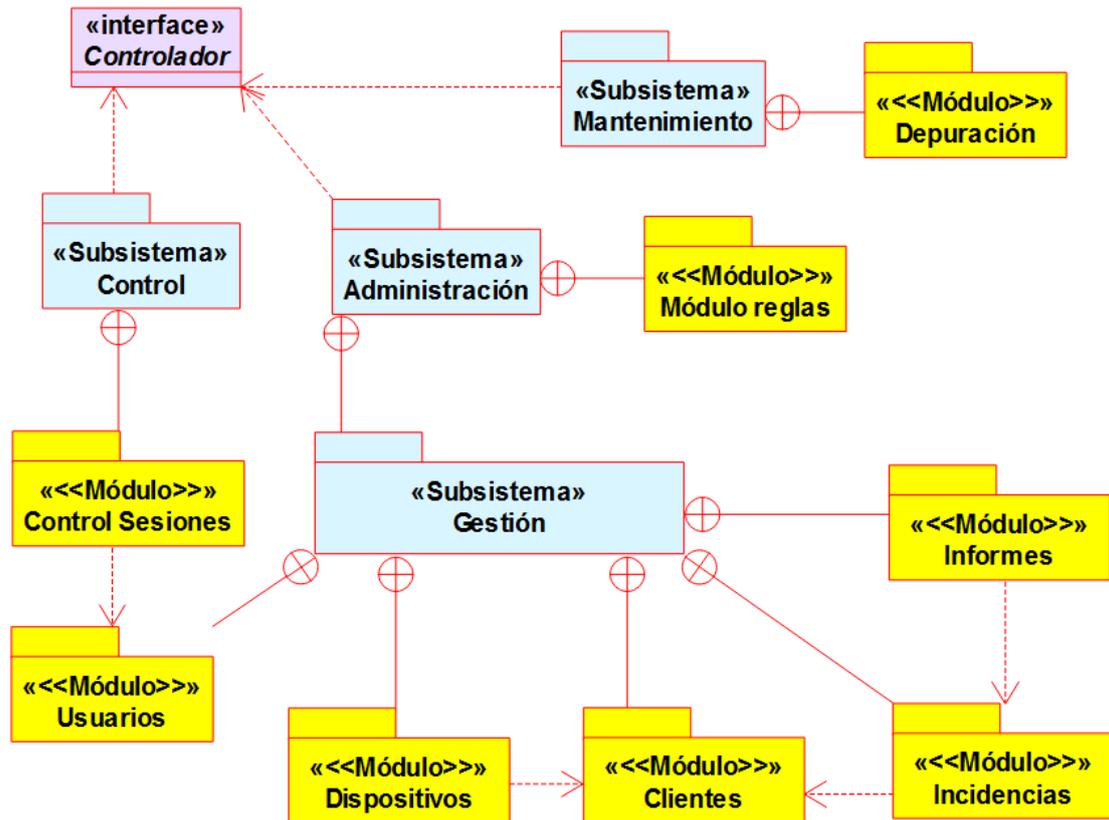
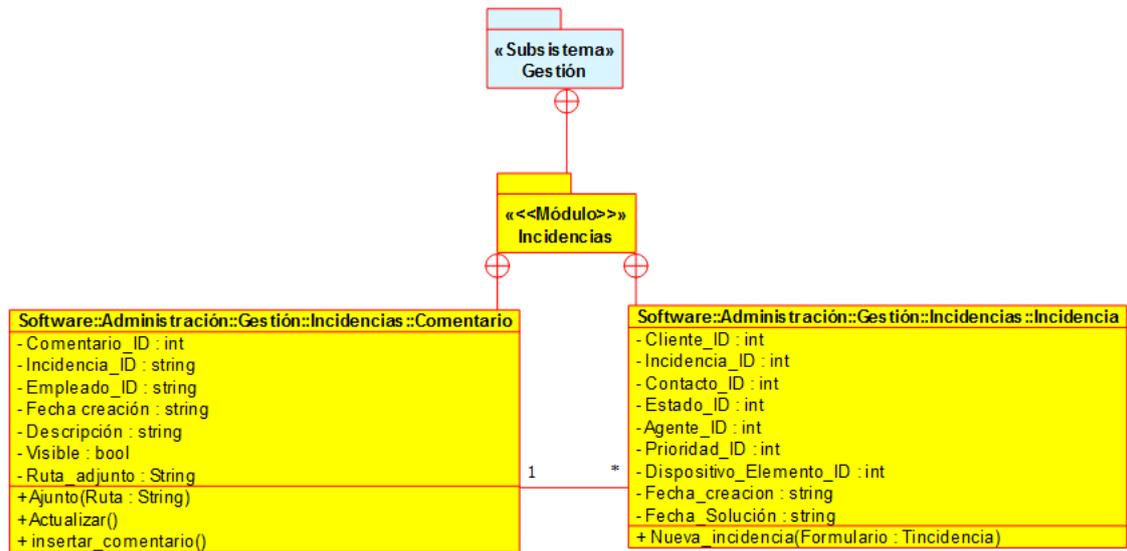


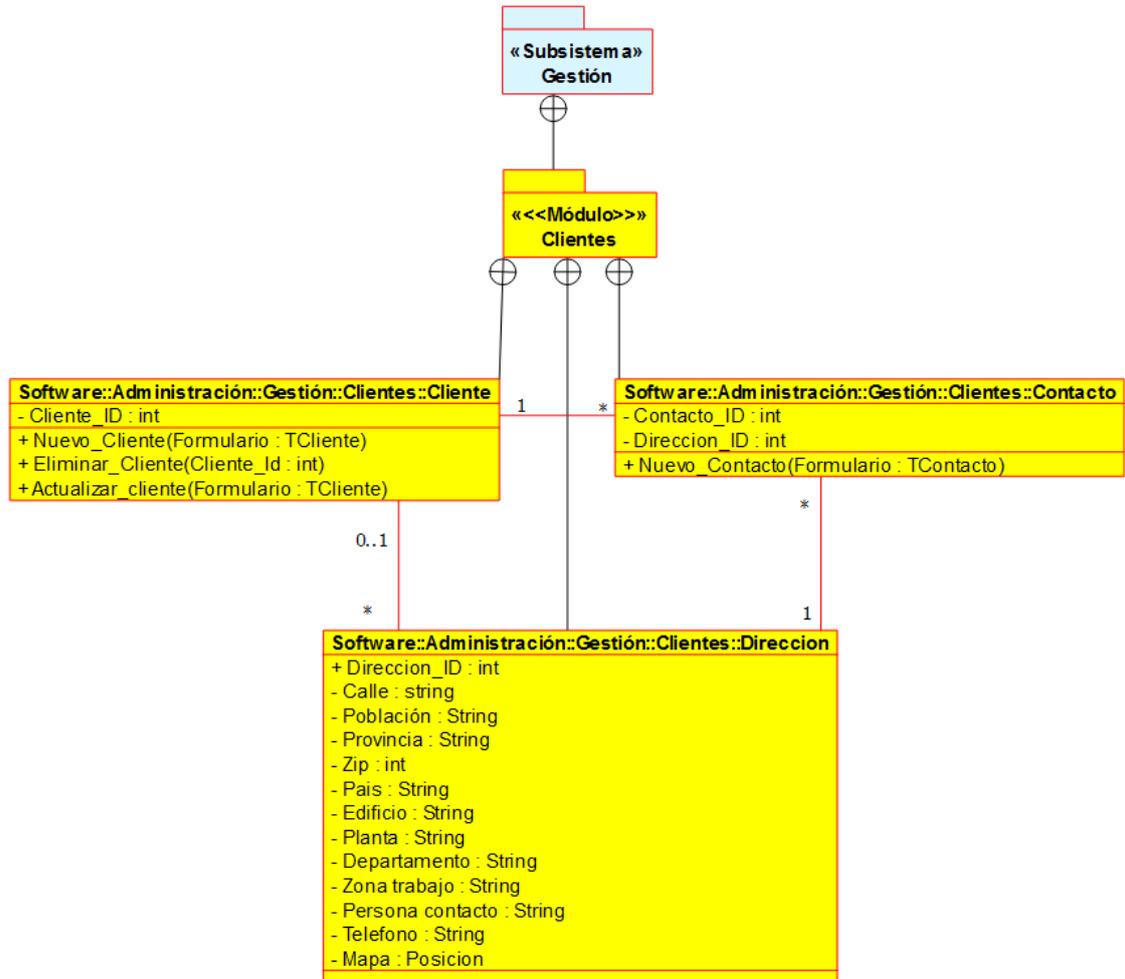
Ilustración 25: Subsistema software

## Subsistema de gestión

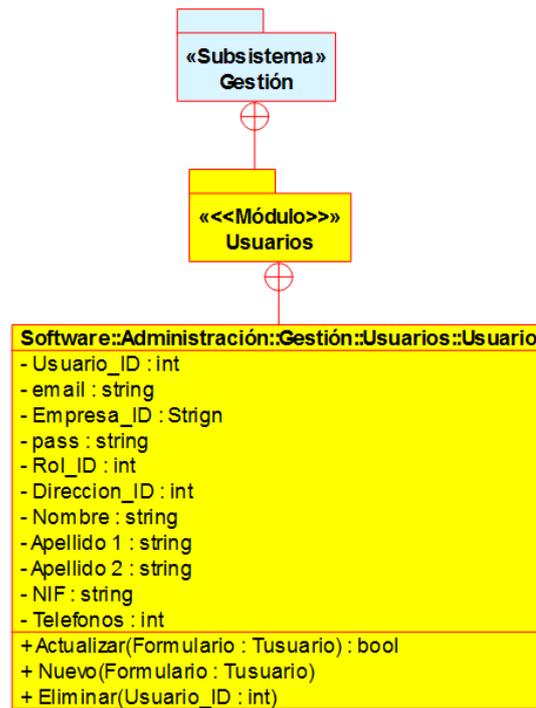
### Módulo de incidencias



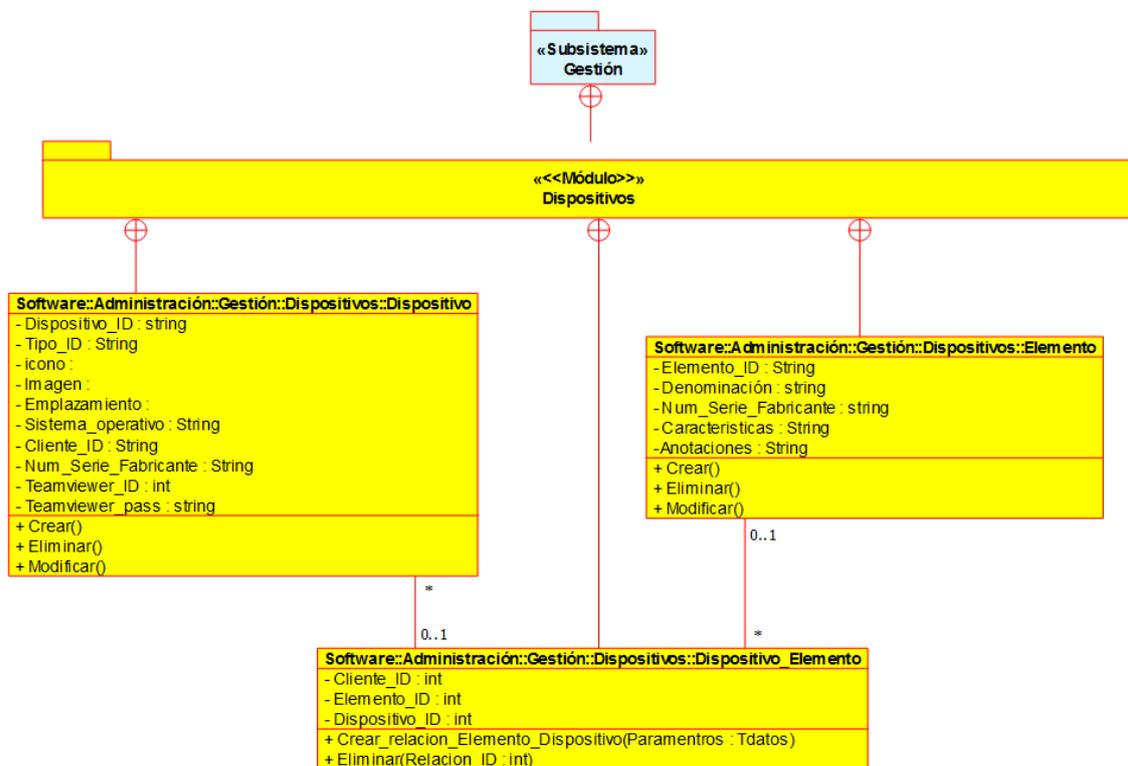
## Módulo de clientes



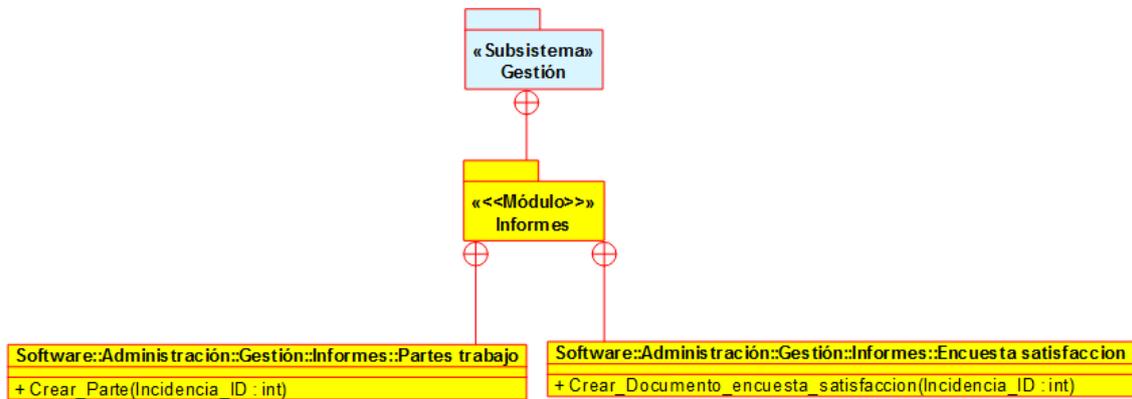
## Módulo de usuarios



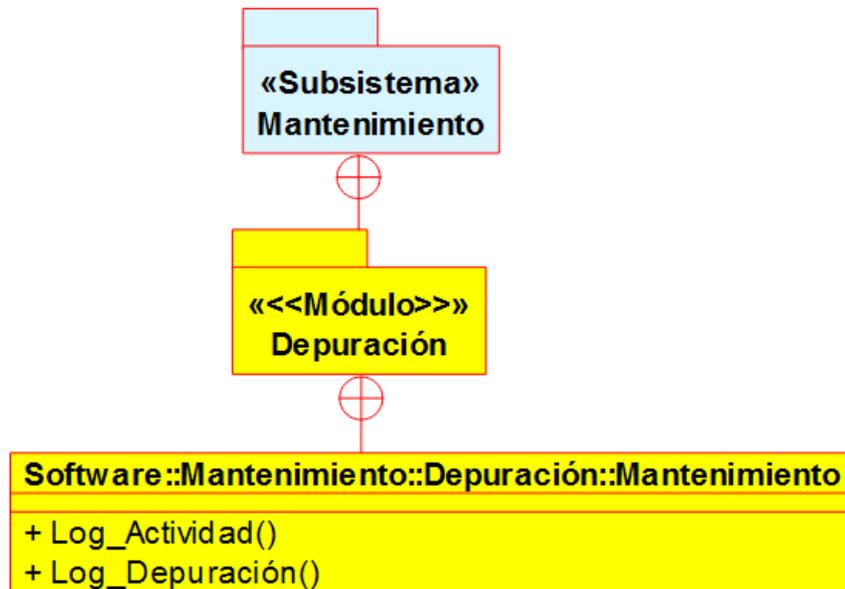
## Módulo de dispositivos



## Módulo de informes

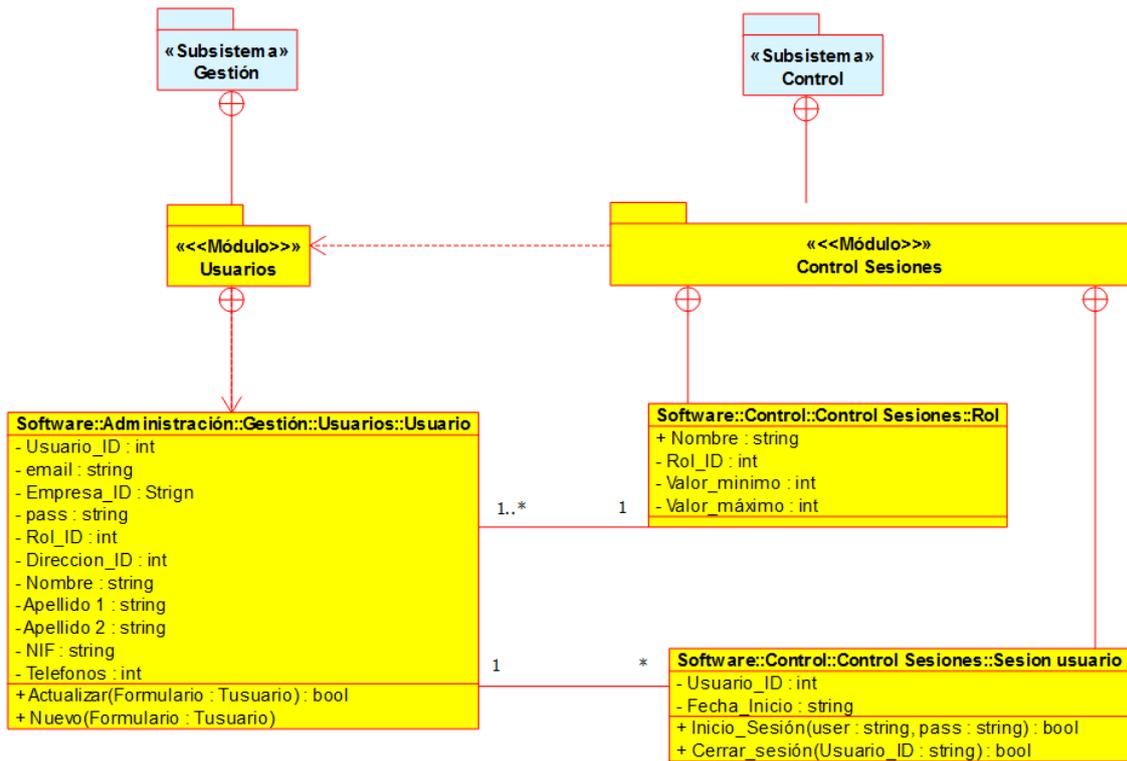


## Subsistema de mantenimiento

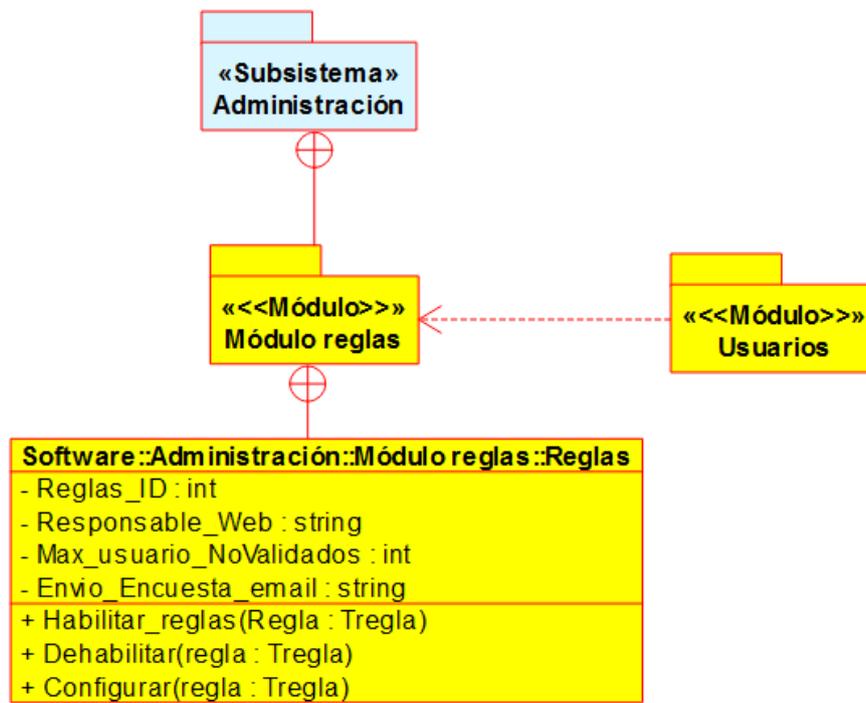


# Subsistema de Control

## Sesión de usuario



## Subsistema de administración



## Identificación de módulos críticos

Tras revisión de cada módulo hay que identificar cuáles son críticos teniendo en cuenta los siguientes criterios:

1. Están dirigidos a varios requisitos
2. Tiene un mayor nivel de control
3. Son complejos o propensos a errores
4. Tienen requisitos no funcionales muy definidos.

Por lo tanto, los módulos que cumplen en gran medida con estos puntos son:

1. Subsistema de gestión
  - Módulo de incidencias
  - Módulo de usuarios
2. Subsistema de control
  - Módulo de control de sesiones

## Patrones de diseño

### Patrón controlador vista modelo

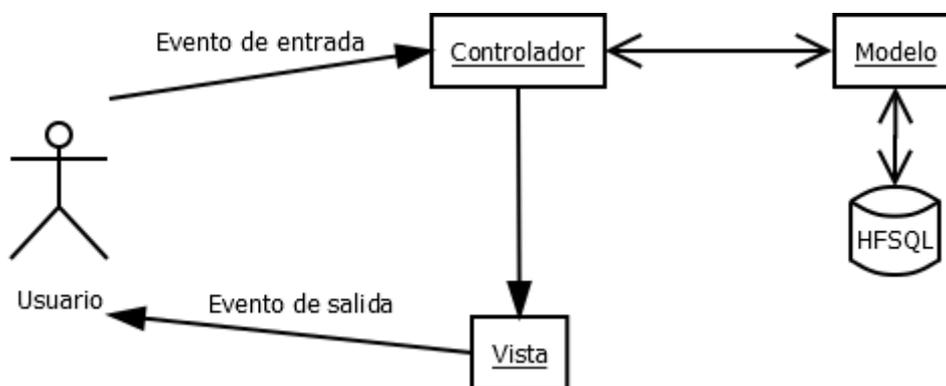


Ilustración 26: Patrón MVC

Se aplicará el patrón MVC por las siguientes razones:

- 1) Interactúan con el usuario a través de la interface de usuario o vista.
- 2) El controlador está presente para cualquier evento proveniente de la vista. Es el encargado de informar al modelo el cual aplicará las reglas y acciones a realizar para mostrarlo en la vista.
- 3) Separadamente, el modelo tiene comunicación bidireccional con el sistema de base de datos.

## Patrón Singleton

El patrón Singleton tiene como objetivo garantizar que una clase solo tenga una instancia y esta proporcione un punto de acceso global a ella.

Este patrón lo veremos en el módulo de mantenimiento. El propósito de este módulo es generar un Log de eventos del sistema. Por lo tanto interesa que sólo haya uno, será el mismo para todos los usuarios conectados simultáneamente.

## Roles y navegabilidad ([OOWS])

Para esta sección se hará uso de la herramienta OOWS (método Orientado a Objetos para Soluciones Web).

Esta metodología proviene de la extensión UML para el modelado de aplicaciones web. Se ha elegido por ser útil a la hora de indicar la navegabilidad.

Observando los casos de uso, los actores y los roles que desempeñan, se obtiene a partir de las funcionalidades a las cuales tiene acceso, el árbol de navegación:

### Sin Rol, usuario aún no logueado

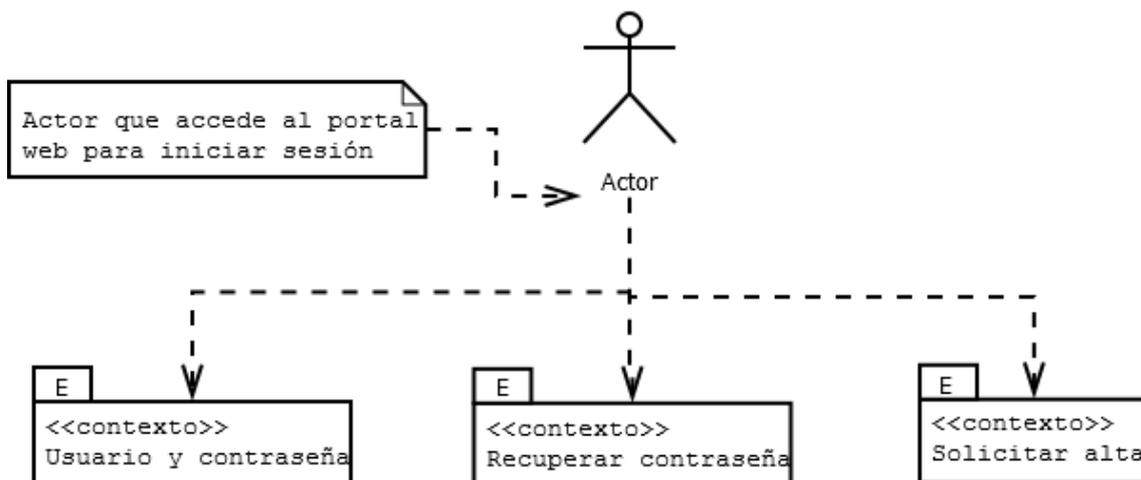


Ilustración 27: Navegabilidad – Acceso portal web

## Rol Cliente

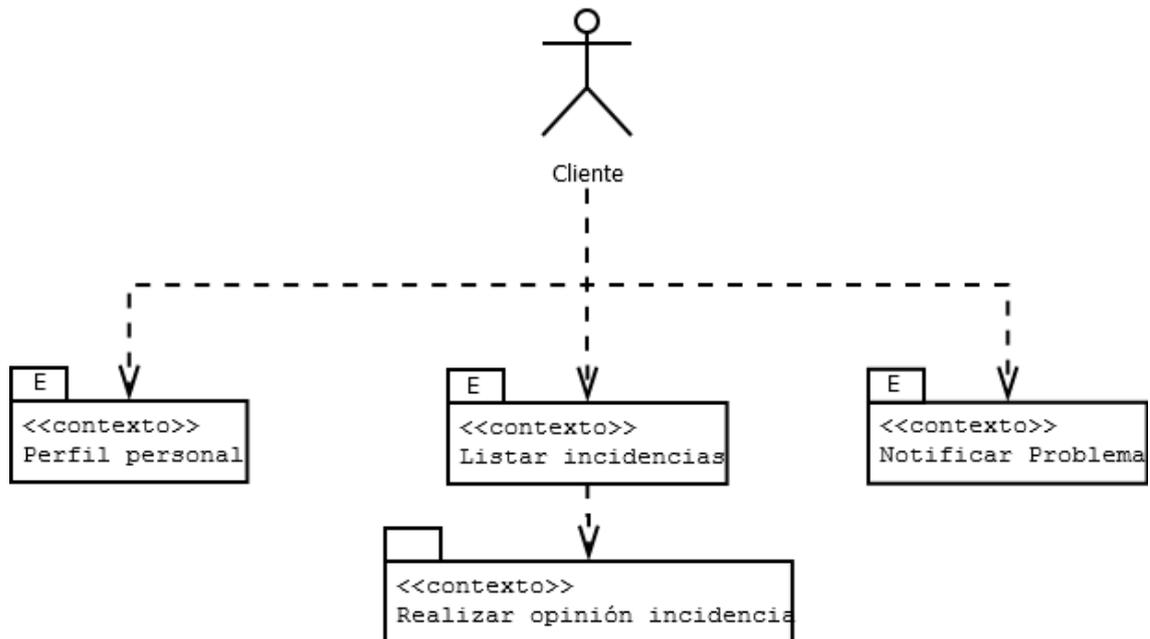


Ilustración 28: Navegabilidad – Portal web del cliente

## Rol usuario atención al cliente

El usuario que atiende al cliente accederá inicialmente a la captura de la llamada y la creación de la nota que identifique al cliente que llama y la razón de su llamada. Que puede derivar en un mensaje o una incidencia. En ambos casos hay que abrir una nota.

También tendrá acceso a la lectura de las incidencias de un cliente, solo consulta ya que no está al cargo de crear incidencias.

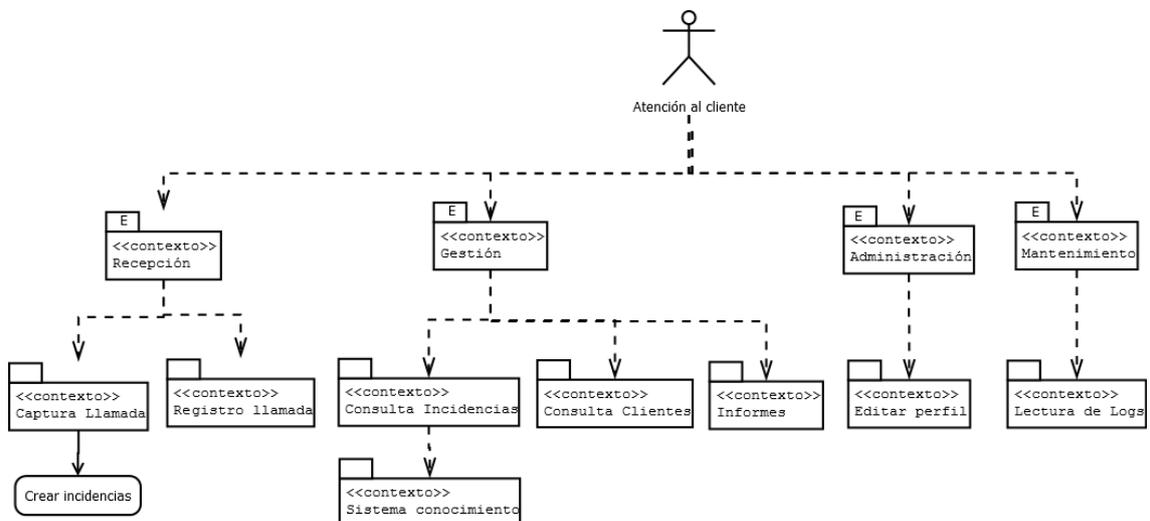
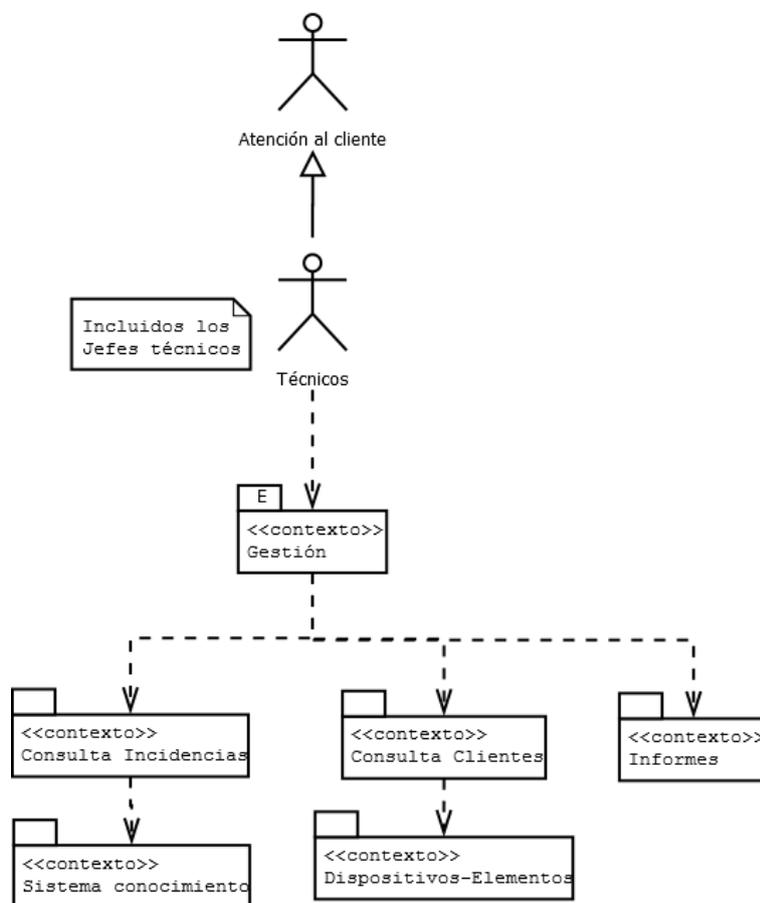


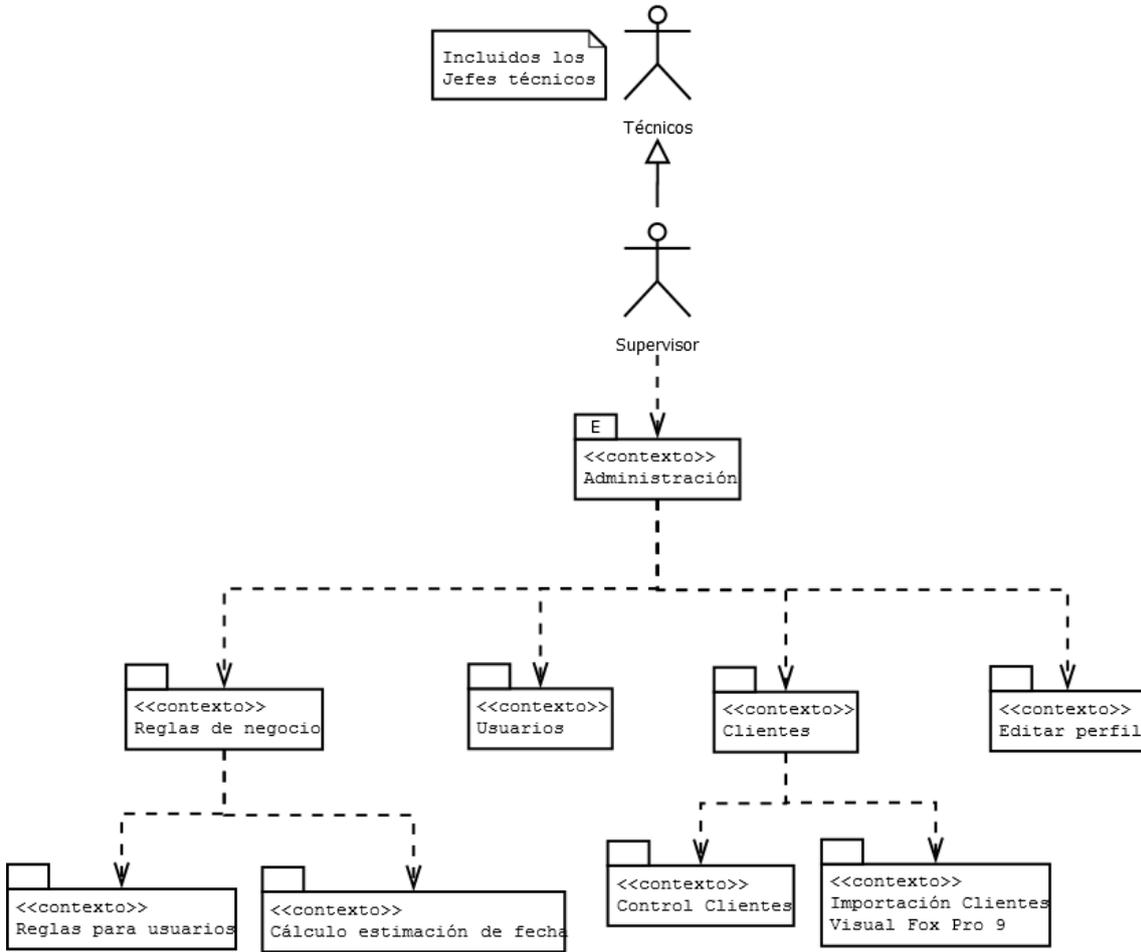
Ilustración 29: Navegabilidad – usuario atención al cliente

## Rol Técnico y encargado técnico.



**Ilustración 30: Navegabilidad – usuarios técnicos**

## Rol Supervisor



**Ilustración 31: Navegabilidad – usuario supervisor**

## Diseño del módulo de acceso y logueado

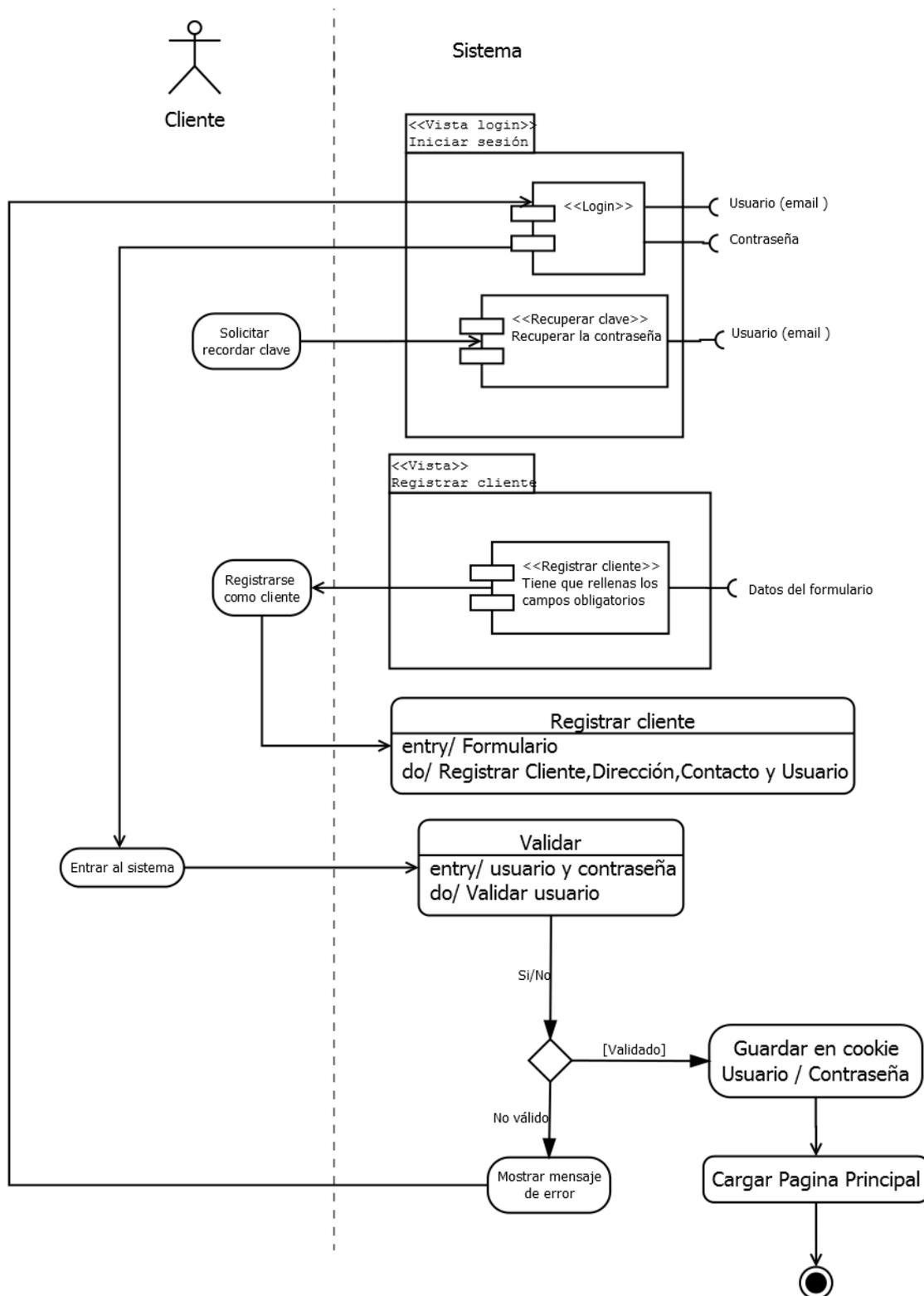
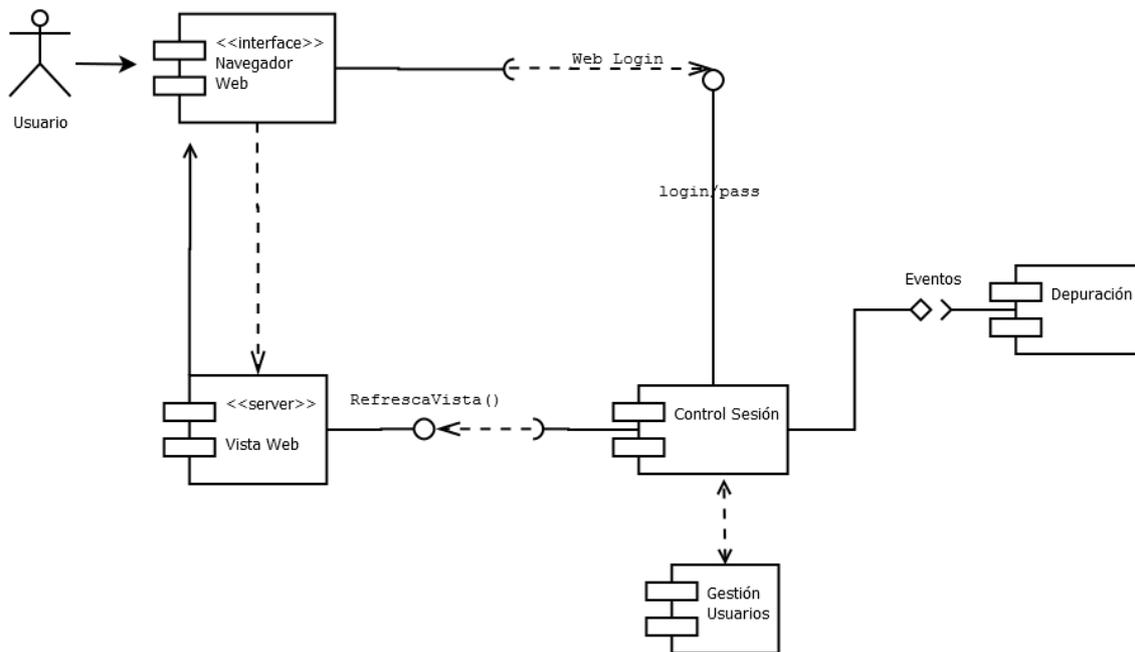


Ilustración 32 – Módulo de acceso y logueado



**Ilustración 33: Vista de componentes con patrón MVC**

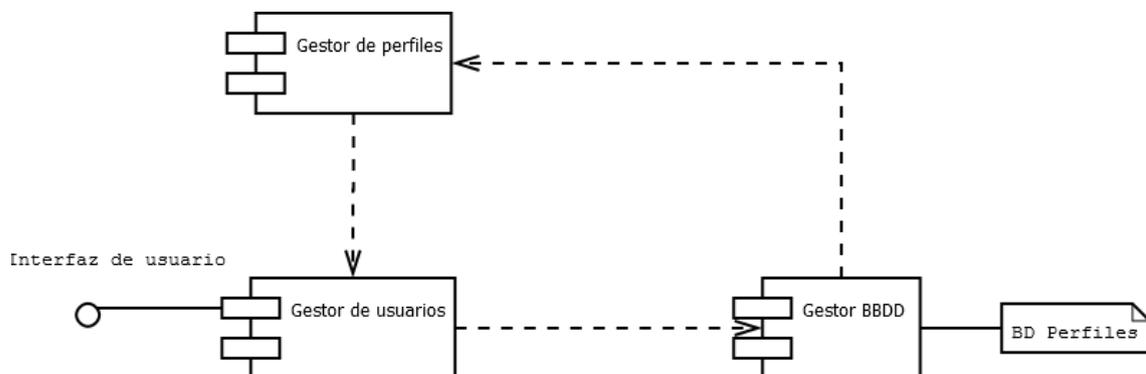
En este diagrama se muestra el uso del patrón controlador vista modelo.

Vista – controlador (control de sesión) – Modelo (Consulta datos de los usuarios)

Tal como define uno de los requisitos funcional, el módulo de depuración tiene la función de registrar los eventos de usuario y las incoherencias bajo código para los desarrolladores.

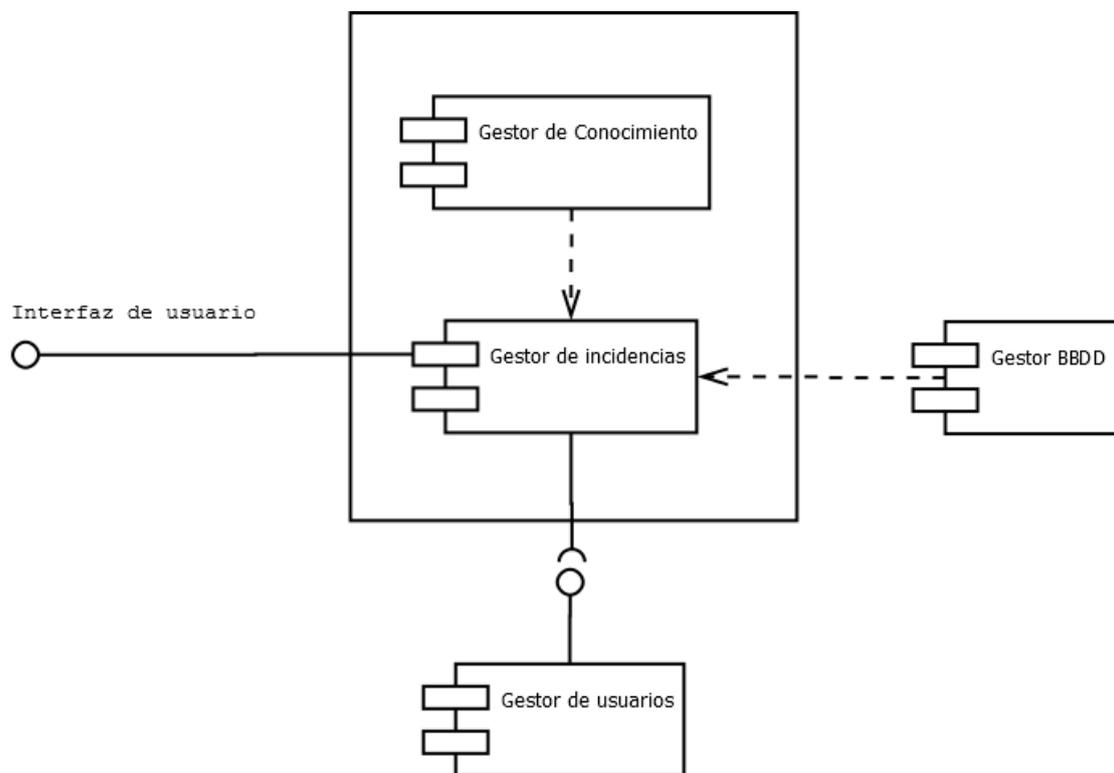
## Diseño de módulos de aplicación

### Diseño de módulo de gestión de usuarios



**Ilustración 34: Gestión de usuarios**

## Diseño de módulo de gestión de incidencias



**Ilustración 35: Gestión de incidencias**

## Diagramas de interacción

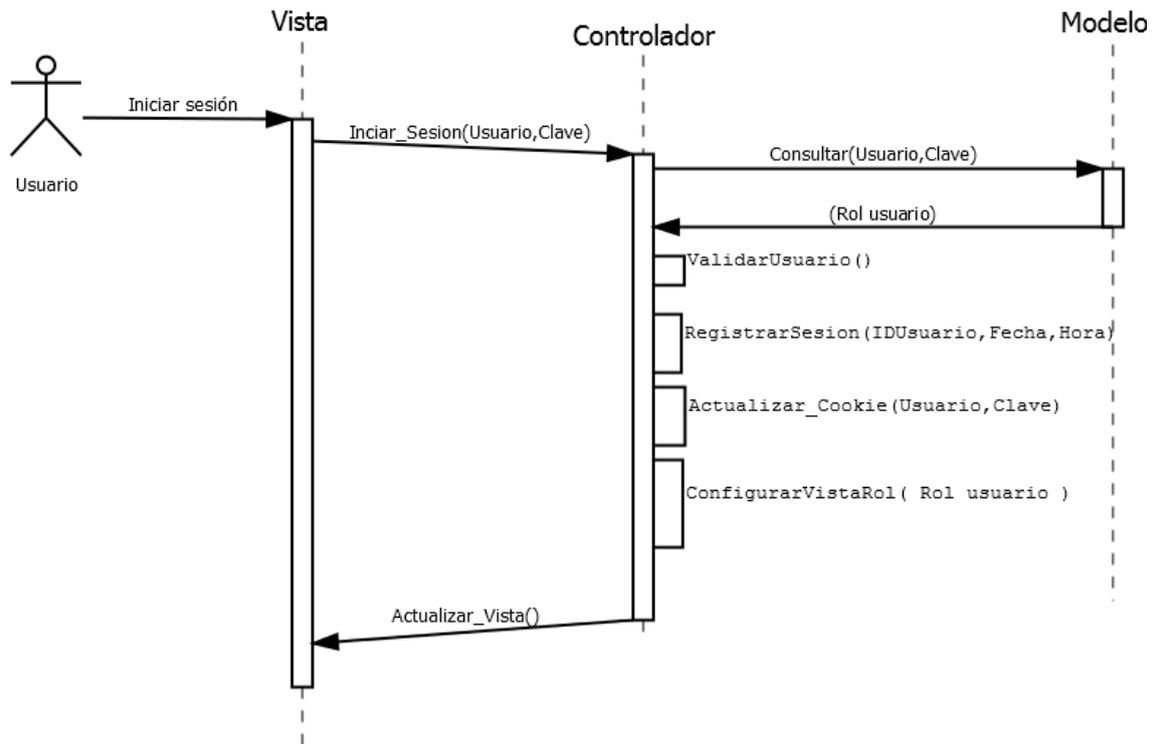


Ilustración 36: Diagrama de interacción – Iniciar sesión

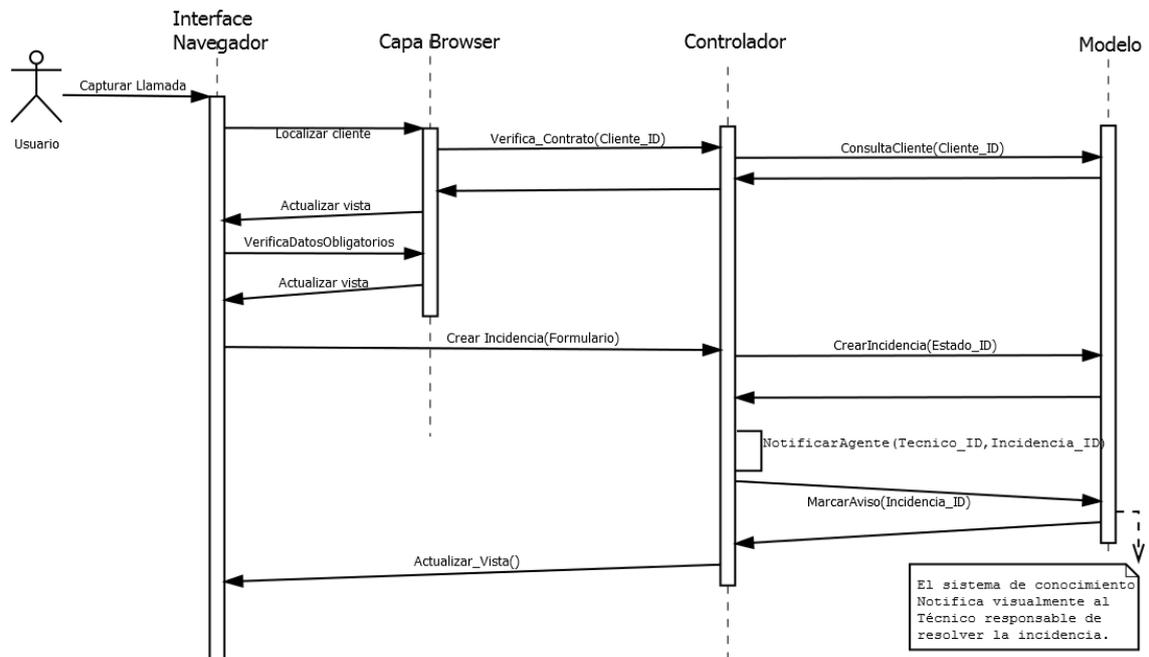
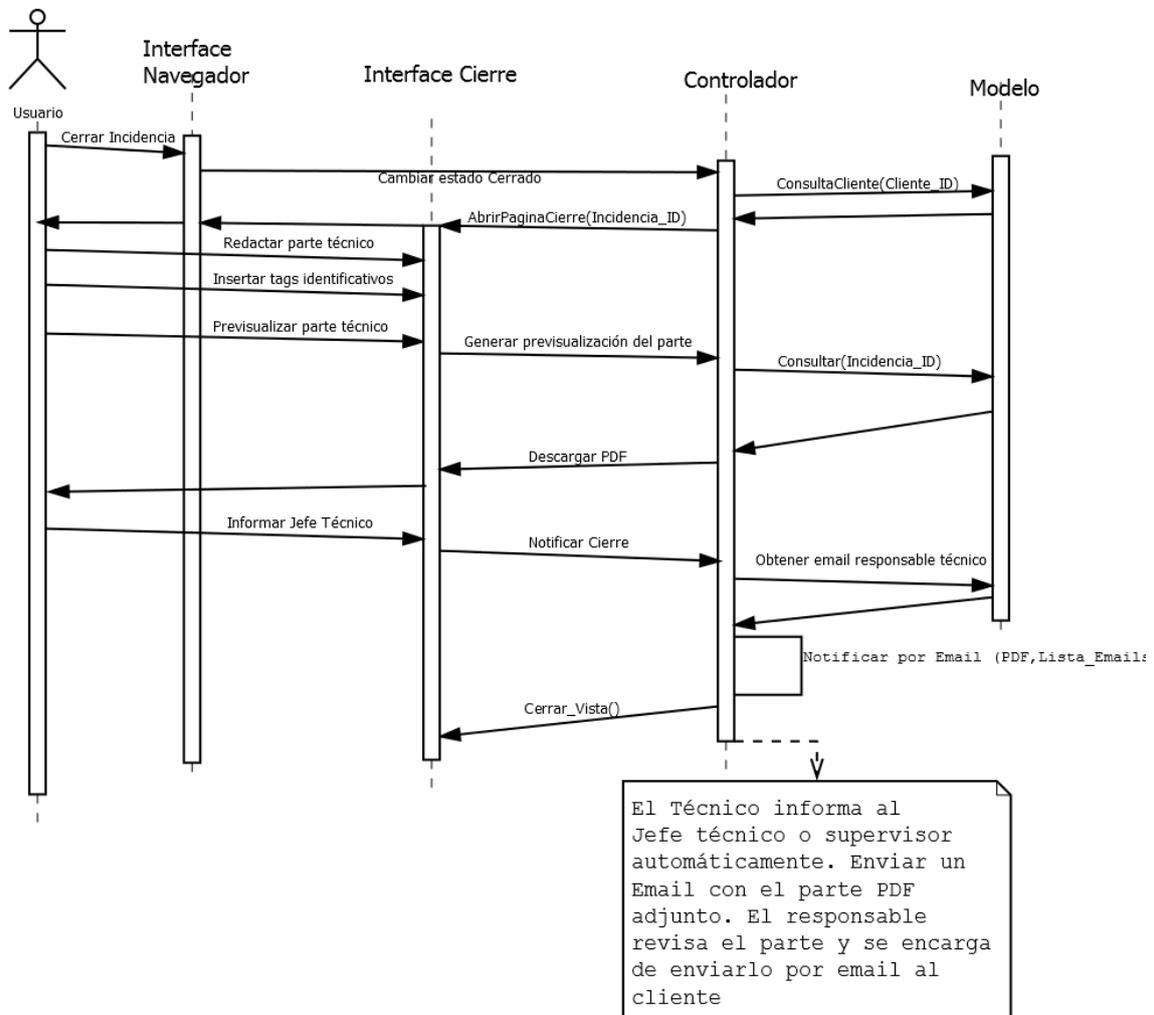


Ilustración 37: Diagrama de interacción – Capturar incidencia telefónica



**Ilustración 38: Diagrama de interacción – Cerrar incidencia**

## Diseño de Base de datos: Modelado Entidad – Relación.

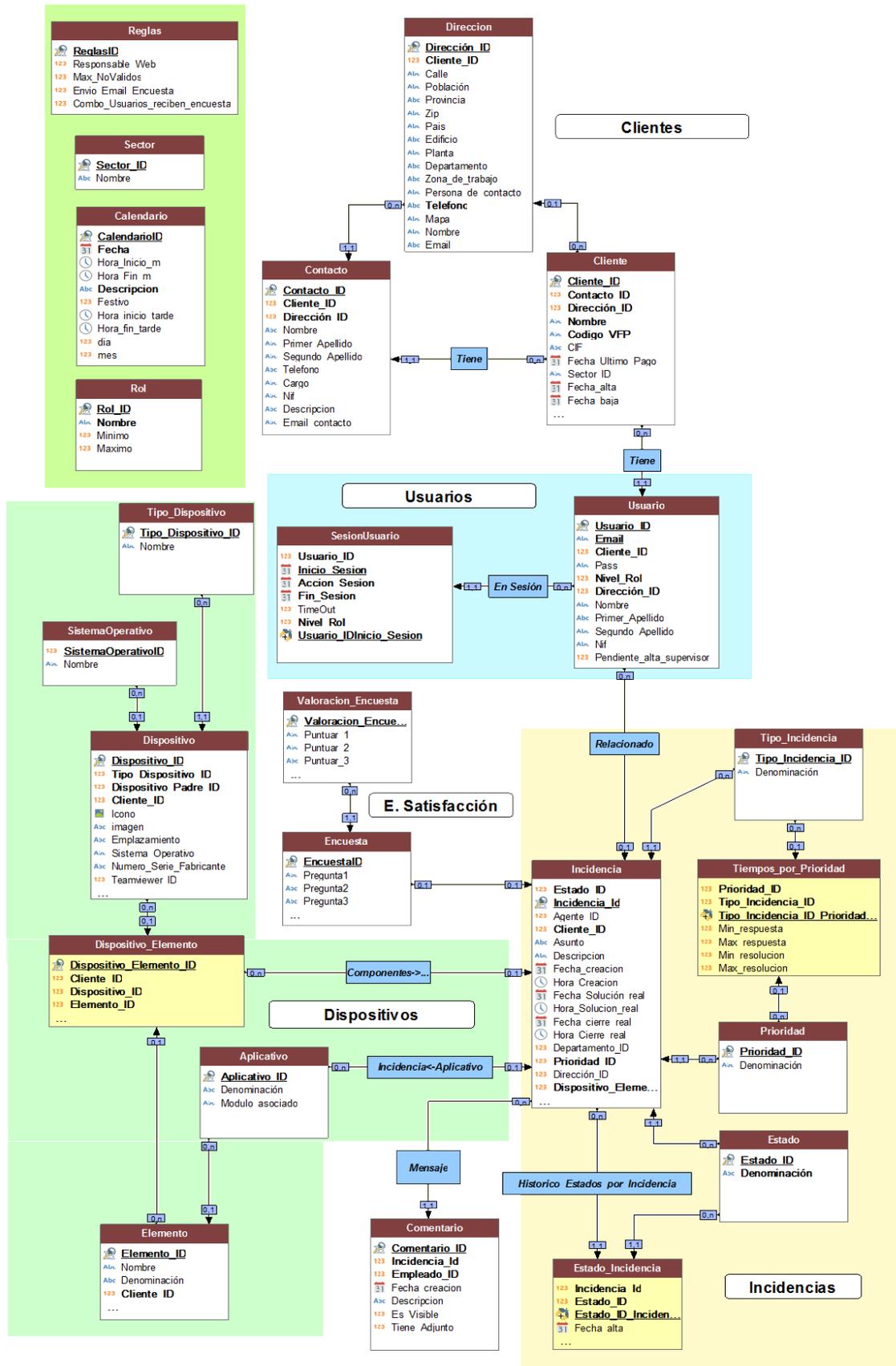


Ilustración 39: Modelo de base de datos del sistema SRIC

Origen	Clave	Cardinal	Destino	Clave	Cardinal
Aplicativo	Aplicativo_ID	0,n	Elemento	Aplicativo_ID	0,1
Aplicativo	Aplicativo_ID	0,n	Incidencia	Aplicativo_ID	0,1
Ciente	Ciente_ID	0,n	Usuario	Ciente_ID	1,1
Ciente	Ciente_ID	0,n	Contacto	Ciente_ID	1,1
Ciente	Ciente_ID	0,n	Direccion	Ciente_ID	0,1
Direccion	Dirección_ID	0,n	Contacto	Dirección_ID	1,1
Dispositivo	Dispositivo_ID	0,n	Dispositivo_Elemento	Dispositivo_ID	0,1
Dispositivo_Elemento	Dispositivo_Elemento_ID	0,n	Incidencia	Dispositivo_Elemento_ID	0,1
Elemento	Elemento_ID	0,n	Dispositivo_Elemento	Elemento_ID	0,1
Encuesta	EncuestaID	0,1	Incidencia	EncuestaID	0,1
Estado	Estado_ID	0,n	Estado_Incidencia	Estado_ID	1,1
Estado	Estado_ID	0,n	Incidencia	Estado_ID	1,1
Incidencia	Incidencia_Id	0,n	Estado_Incidencia	Incidencia_Id	1,1
Incidencia	Incidencia_Id	0,n	Comentario	Incidencia_Id	1,1
Prioridad	Prioridad_ID	0,n	Incidencia	Prioridad_ID	1,1
Prioridad	Prioridad_ID	0,n	Tiempos_por_Prioridad	Prioridad_ID	0,1
SistemaOperativo	SistemaOperativoID	0,n	Dispositivo	SistemaOperativoID	0,1
Tipo_Dispositivo	Tipo_Dispositivo_ID	0,n	Dispositivo	Tipo_Dispositivo_ID	1,1
Tipo_Incidencia	Tipo_Incidencia_ID	0,n	Incidencia	Tipo_Incidencia_ID	1,1
Tipo_Incidencia	Tipo_Incidencia_ID	0,n	Tiempos_por_Prioridad	Tipo_Incidencia_ID	0,1
Usuario	Usuario_ID	0,n	Incidencia	Ciente_ID	0,1
Usuario	Usuario_ID	0,n	SesionUsuario	Usuario_ID	1,1
Valoracion_Encuesta	Valoracion_EncuestaID	0,n	Encuesta	Tipo_Valoracion_EncuestaID	1,1

**Ilustración 40: Cardinalidad de las relaciones entre tablas**

En cada una de las relaciones por defecto se aplican unas reglas automáticas para las operaciones de borrado y modificación de la clave compartida, tanto para la tabla origen como destino.

Por defecto, las reglas son:

1. Prohibido borrar un registro si la clave relaciona las tablas origen y destino.
2. Prohibido modificar la clave si ésta relaciona las tablas origen y destino.

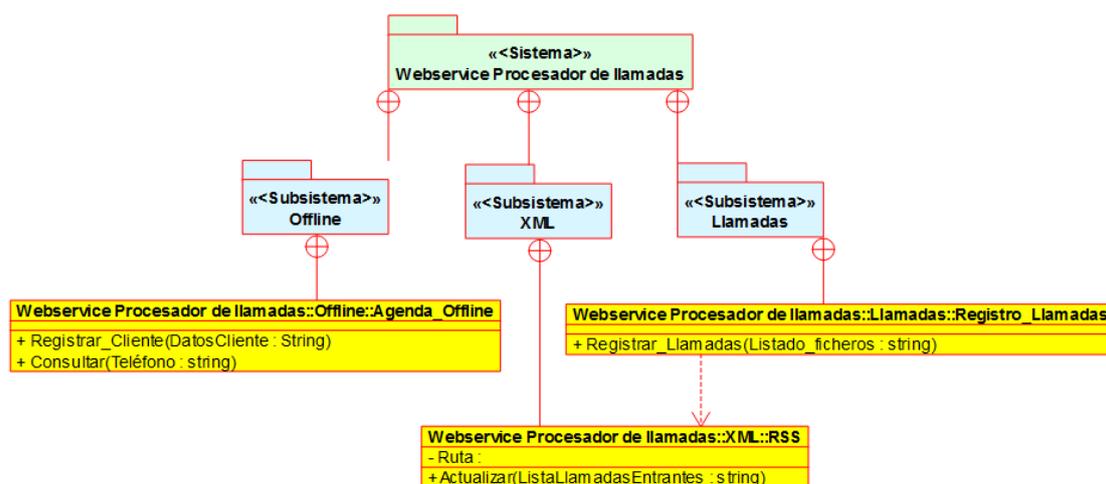
El modelo de base de datos requiere la personalización de algunas relaciones entre tablas para obtener mayor estabilidad.

- **Tabla Clientes y Dirección:**
  - Al borrar un cliente borra todas sus direcciones (Tabla Dirección)
  - Prohibido modificar el identificador del cliente si tiene al menos una dirección.
- **Tabla Dirección y Contacto:**
  - Prohibido borrar una dirección que tenga al menos un contacto
  - Prohibido modificar el identificador de Dirección si al menos tiene un contacto.
- **Tabla Cliente y Contacto:**
  - Al borrar un cliente borra todos sus contactos
  - Prohibido modificar Cliente\_ID si al menos tiene un contacto.

- **Tabla Usuario y Cliente:**
  - Al borrar un usuario, mantener todas sus incidencias
  - Prohibido modificar su Usuario\_ID si al menos tiene una incidencia.
  
- **Tabla Usuario y SesionUsuario:**
  - Prohibido borrar un usuario si al menos tiene una entrada en SesionUsuario
  - Modificar Usuario\_ID en la tabla Usuario implicará que cambie igualmente en Sesión de Usuario.
  
- **Tabla incidencia y Comentario:**
  - Al borrar una incidencia se eliminará todos los comentarios relacionados
  - Prohibido modificar la clave compartida.
  
- **Tabla Encuesta e Incidencia:**
  - Al borrar una encuesta, no borrará el registro de incidencia.
  - Prohibido modificar la clave compartida.

## Diseño del Webservice

### Sistemas y subsistemas

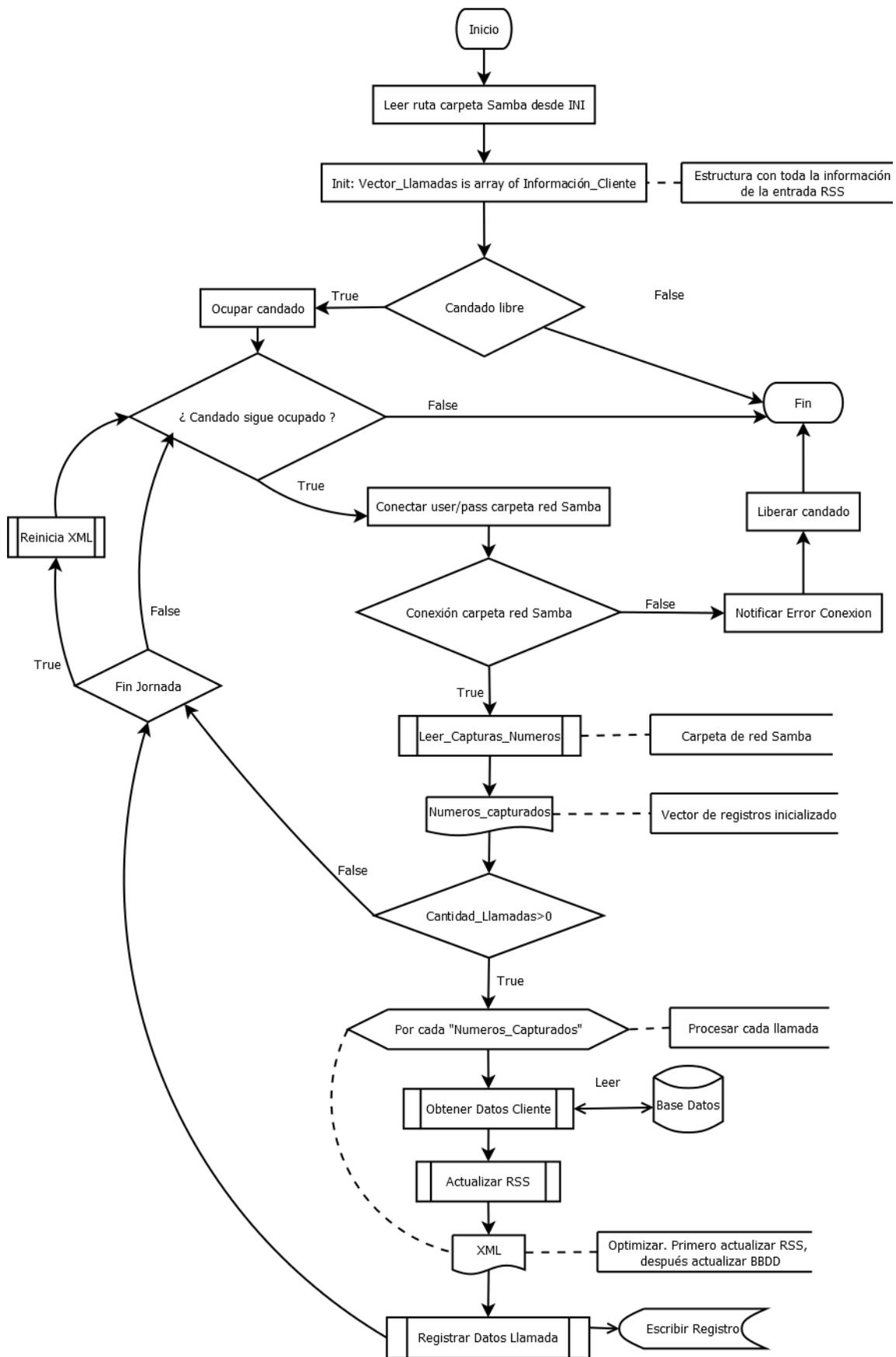


**Ilustración 41: Subsistemas del webservice**

La llamada al webservice se realiza desde Linux mediante un script situado dentro del Dialplan de Asterisk durante el proceso de una llamada entrante.

La llamada al webservice ejecuta el método que procesa la llamada. El método necesita los siguientes datos:

- Cada llamada es un fichero de texto formato TXT, cuyo nombre contiene el número de teléfono origen, teléfono destino, fecha y hora exacta de la llamada.
- Como salida depositará en formato XML v2.0 dentro del RSS estas llamadas y esto permitirá que quien se conecte a dicho RSS (por razón de seguridad y protección de datos) será en la intranet de Kernel Informática S.L, podrá leer en tiempo real las llamadas entrantes.



**Ilustración 42: Diagrama de flujo del método que procesa las llamadas en el webservice.**

## Arquitectura de Base de Datos del webservice

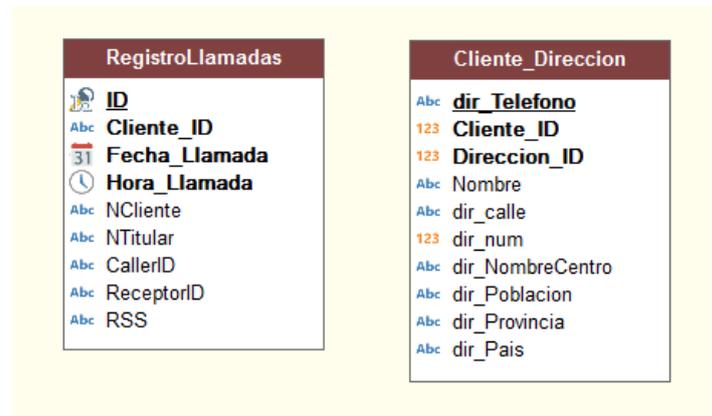


Ilustración 43: Base de datos del webservice

## Diseño arquitectónico para transferir las llamadas desde Asterisk

A continuación se muestra un diagrama completo que indica el proceso de captura de la llamada desde el dialplan de Asterisk. Posteriormente, el nombre del fichero se pasa por parámetro al script Procesar\_Registro.sh, quien se encarga de asegurar de conexión con el servidor WebDev. Si consigue conectar, transfiere el fichero de la llamada a la carpeta compartida y se llama al webservice para que obtenga los datos que necesita. Actualiza el RSS y finaliza.

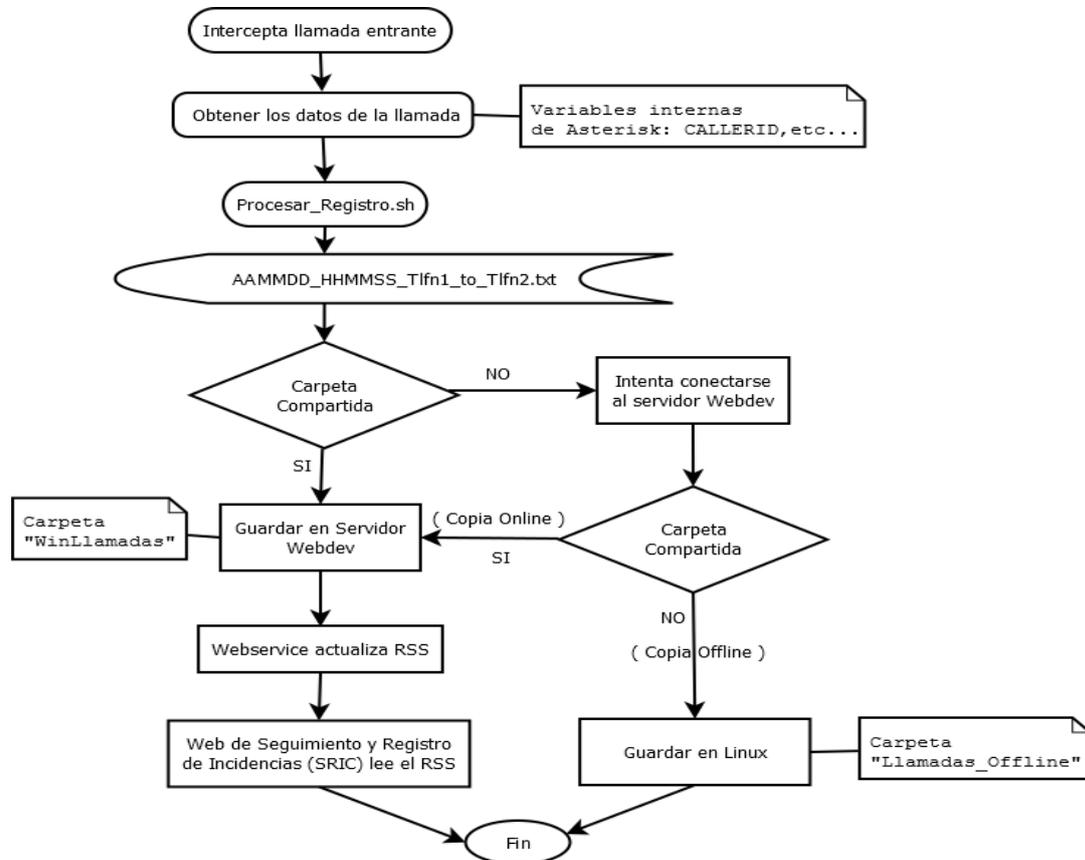


Ilustración 44: Captura de llamada mediante Asterisk y procesamiento del webservice.

## Estudio de herramientas para el sistema SRIC

### Entorno de programación (Webdev 21)

La herramienta web a utilizar es WebDev en su versión 21 de la plataforma PCSoft

Es una herramienta orientada al desarrollo de aplicaciones web dinámicas. Adaptado para aplicaciones web con gestión de datos en tiempo real.

W-Language es el lenguaje de programación que utiliza WebDev. Permite la programación por procedimientos y la programación orientada a objetos.

WebDev se basa en HTML, HTML5, XML, CSS, JavaScript o PHP. Genera el código necesario automáticamente. Además, soporta de manera integrada AJAX 2.0.

### HyperFileSQL server

HFSQL (HyperFileSQL) es un potente SABR (Sistema de Administración de Base de datos Relacional). Existente en cuatro versiones:

1. Versión móvil (integrada)
2. Versión Cliente/Servidor
3. Versión local (independientemente o en red)
4. Versión para grupos (clúster)

Para la realización de este proyecto se requerirá de la versión local ya los ficheros HFSQL se alojaran en el servidor proporcionado por Kernel Informática S.L.

HFSQL está disponible para todos los tipos de aplicaciones: aplicaciones de negocios, aplicaciones críticas en tiempo real 24/7, software, servidores de aplicación, servidores web, PC independiente o dispositivos móviles.

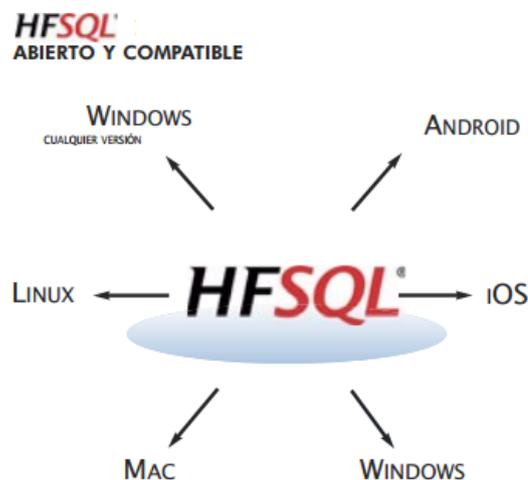


Ilustración 45: Servidor HFSQL

Para que el aplicativo SRIC una vez se instale pueda funcionar correctamente, necesitará tener instalado el “Application server HFSQL”, el servidor de base de datos.

### Editor HFSQL control center

Editor que permita la consulta de los datos contenidos en los ficheros con extensión \*.FIC, los cuales tienen el mismo nombre que las tablas del análisis del proyecto WebDev.

### WAMP

La herramienta dispone de un servidor Apache. Bajo la recomendación de PCSoft, se inhabilita el servicio ISS para que no tenga conflicto con Wamp. Es el encargado de configurar el fichero httpd.conf de forma automática.

Durante la instalación y pruebas realizadas, si tiene en cuenta que la carpeta de base de datos tiene que ser compartida entre el webservice y SRIC, por lo tanto, la carpeta tiene que estar dentro de una de las rutas configuradas dentro de httpd.conf. Si no fuera así, apache no permitiría acceder a otra carpeta local, que no esté declarada.

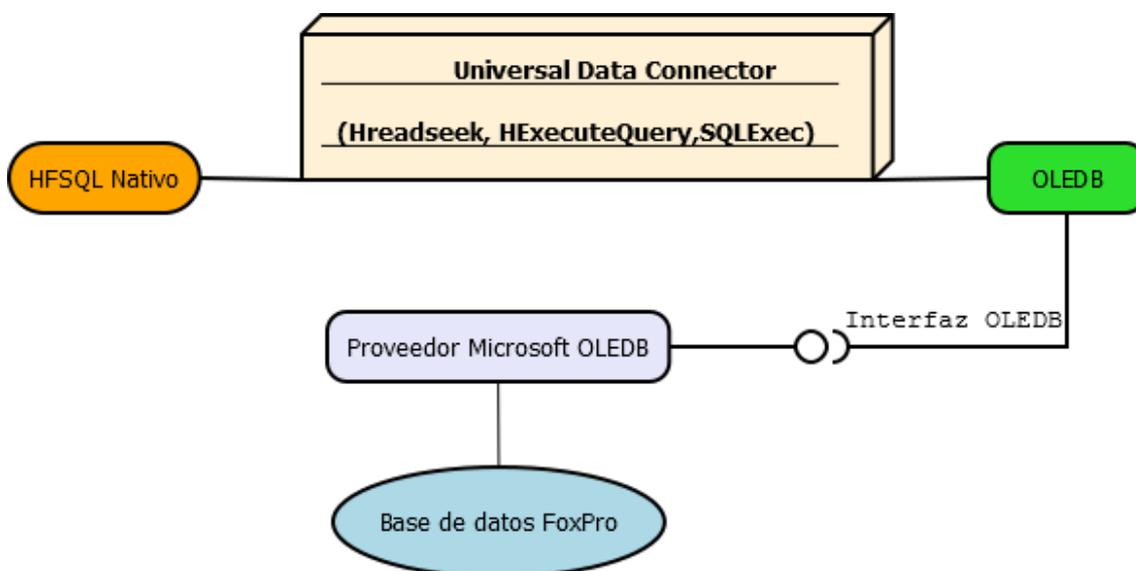
### Driver Proveedor OLEDB de Microsoft para visual Fox Pro 9

Con este driver ([VFPv9]) se pretende hacer uso a las cualidades de WebDev de acceder nativamente a la base de datos en VFP9.

El requisito funcional 1.2.8 indica que hay que realizar una migración de datos desde Microsoft Visual Fox Prov.9, por lo tanto:

1. Hay que analizar la estructura en la que se almacenan los clientes, para realizar el mapeo a SRIC y posterior procesamiento para:
  - a. Crear el usuario, el cliente, la dirección y su contacto en la base de datos relacional HFSQL de WebDev.
  - b. El acceso a la base de datos Visual Fox Pro requerirá de la instalación del interfaz OLEDB. Microsoft facilita el controlador en su web oficial.
  - c. Se verifica en la documentación la compatibilidad de HFSQL para hacer uso de OLEDB.

Tal como se muestra en la siguiente ilustración, quedaría de la siguiente manera:



**Ilustración 46: Conexión a la base de datos de Visual Fox Pro mediante interface OLEDB**

Durante la instalación del controlador OLEDB para Visual Fox Pro, surgió un problema inesperado. El interface no funcionaba correctamente, no era capaz de leer la estructura del fichero .DBF. Tras varios intentos se detectó el problema.

Durante la descarga del controlador, el sistema descarga la versión del controlador OLEDB para 64 bits. Visual Fox Pro es un aplicativo de 32bits por lo que hubo que cambiar el controlador a 32bits. A partir de este punto, funcionó, pudiendo efectuar la implementación de las rutinas para la migración.

### VirtualBox

Herramienta para la ejecución de máquinas virtuales. Al descargar AsteriskNow en una ISO instalable, permite la instalación dentro de virtualbox, creando la máquina virtual.

La máquina contiene un Linux Centos con un Servidor FreePBX Asterisk.

### Estudio de herramientas para el Webservice.

El Webservice para el procesamiento de las llamadas, está en el mismo servidor que el aplicativo web SRIC. Por lo tanto tiene lo que necesita: Webdev Application server, HyperFileSQL server y WAMP.

## Implementación

### Implementación e interface de usuario de SRIC.

#### Declaración en la inicialización del proyecto SRIC

```
// Instancia de las clases
gn_Depuraci on is object c_Depuraci on dynamic
gn_Controlador is object c_Controlador dynamic

// Patrón SINGLETON:
//El objeto Depuración es el mismo para todos los usuarios.
gn_Depuraci on = c_Depuraci on::Tomar_instancia()

// Patrón Controlador-Vista-Modelo:
gn_Controlador= new c_Controlador()

gn_Depuraci on. Log_Acti vi dad("Iniciado el Software")
```

#### Método Constructor de la clase “Controlador” de SRIC

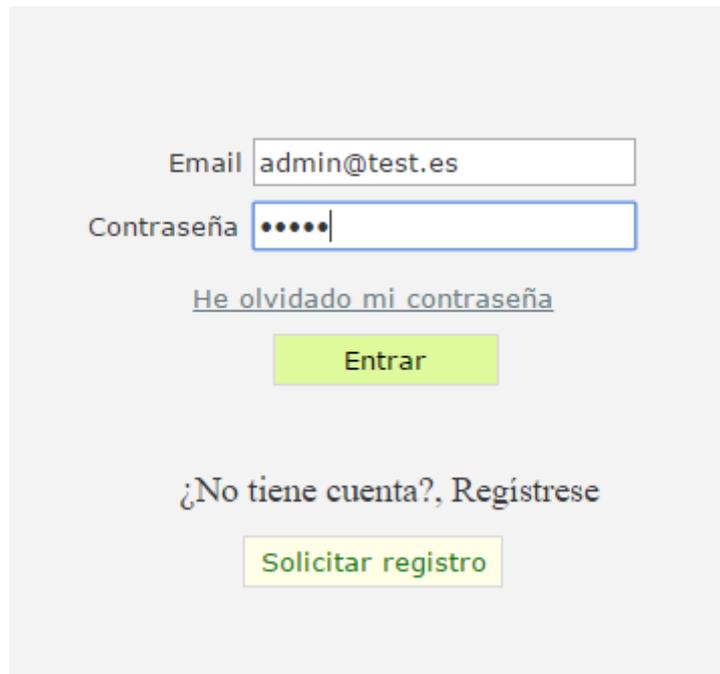
El siguiente código muestra la declaración y la creación de la instancia de las clases Vista y Sesión de usuario

Aclarar que el objeto “modelo” no es necesario, ya que por cada proyecto realizado en Webdev, crea automáticamente su análisis interno y se accede mediante instrucciones Wlenguage nativas. Por lo que el controlador además de sus funciones de control, implementa internamente los métodos de consulta HFSQL.

```
obj_Vista is object c_Vista dynamic
obj_Sesion is object c_Sesion_Usuario dynamic

obj_Sesion = new c_Sesion_Usuario()
obj_Vista = new c_Vista()
```

## Diseño del Login



The image shows a login form with the following elements:

- An "Email" input field containing the text "admin@test.es".
- A "Contraseña" (Password) input field containing five dots, indicating a masked password.
- A link labeled "He olvidado mi contraseña" (I forgot my password).
- A green button labeled "Entrar" (Login).
- A link labeled "¿No tiene cuenta?, Regístrese" (Don't have an account? Register).
- A yellow button labeled "Solicitar registro" (Request registration).

Ilustración 47: diseño de Login

Código del evento Click en el botón "Entrar", sino `gn_Controlador` la variable general que contiene al objeto controlador, declarado en la inicialización del programa.

```
gn_Controlador.SU_Iniciar_Sesion(edt_Email, edt_pass)
```

### Código:

```
PROCEDURE SU_Iniciar_Sesion(user, pass are string)
IF obj_Sesion=NULL THEN
    obj_Sesion = new c_Sesion_Usuario()
END
IF obj_Sesion.Iniciar_sesion(user, pass) THEN
    Selector_Page_Inicio is int = Ctrl_Login_Cliente(user)
    IF Selector_Page_Inicio<0 THEN
        gn_Depuracion.Log_Depuracion("Controlador - SU_Inicio_Sesion: La función
Ctrl_Login_Cliente() devuelve un valor negativo. Abortar inicio de sesión")
        RESULT False
    END
    //Guarda el user y pass en una cookie durante 75 dias
    CookieWrite("Login", user+", "+pass, 75)
```

```

IF Selector_Page_Inicio=1 THEN
    //Es un cliente
    PageDisplay(Portal_de_incidencias)
ELSE
    //Es un empleado
    PageDisplay(Menu_Principal)
END

RESULT True

ELSE
    obj_Sesion.Cerrar_Sesion()

    Info("No ha podido iniciar sesión")
    RESULT False
END

```

## Recordar contraseña

Desde el Login se puede pulsar en "He olvidado mi contraseña". El único requisito es tener el nombre de usuario correcto. La clave la recuerda vía email.

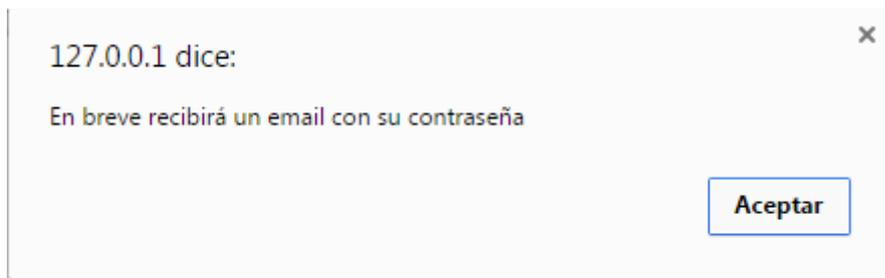


Ilustración 48. Mensaje de confirmación

### Código:

```
gn_Controlador.SU_verificar_email(NoSpace(edt_Email))
```

### Código del método:

```

PROCEDURE SU_verificar_email(email is string)

IF NoSpace(email)="" THEN
    Info("Por favor inserte su email")
    RETURN
END

IF SU_envia_email_password(email) THEN
    Info("En breve recibirá un email con su contraseña")
ELSE
    Info("Por favor, inténtelo más tarde")
END

```

## Registro de nuevo usuario

Desde la ventana principal de login se accede al formulario de registro pulsando el botón “Solicitar Registro”, que contiene el siguiente código servidor en el evento “Click”:

```
IF NOT gn_Controlador.Ctrl_Regla_Permitir_alta() THEN
    Info("Función deshabilitada temporalmente, disculpe las molestias")
    RETURN
END
```

`PageDisplay(Solicitud_de_alta)`

Se ejecuta la regla de negocio que controla, por seguridad, el número de usuarios no validados por el administrador, quien da permiso para acceder al siguiente formulario:

The image shows a web form titled "Formulario de alta" (Registration Form). It is divided into four main sections: "Contacto", "Empresa", "Dirección de la empresa", and "Usuario".

- Contacto:** Fields for Nombre \* (Juan), Segundo Apellido (Sanchez), Primer Apellido (Lopez), Email corporativo \* (juan@farmalios.es), NIF \* (42.234.213), Cargo, and Telefono contacto.
- Empresa:** Fields for Nombre Empresa \* (Farmacia Alisios S.L.) and CIF empresa \* (2134-2134-H).
- Dirección de la empresa:** Fields for Telefono \* (928556677), Email \* (info@farmalios.es), Calle, Población, Provincia, Edificio-Número, País, Zip, Zona, and Planta.
- Usuario:** Fields for Email, Nombre del login completo \* (info@farmalios.es), Contraseña \* (masked with dots), and Repita Contraseña \* (masked with dots).

Buttons for "Cancelar" and "Enviar" are located at the bottom of the form. A "Volver" button is in the top right corner.

**Ilustración 49: Formulario de alta de usuario**

El código servidor en el evento “Click” del botón “Enviar” tiene como propósito verificar que los valores introducidos son correctos y proceder con la llamada al controlador para realizar el registro del usuario

El código es el siguiente:

```
Formulario is ST_Alta_Cliente
```

```
Formulario.Contacto_Alta.Cargo = Cargo_usuario
```

```
IF Length(NoSpace(NIF)) < 8 THEN
```

```
    Info("El nif del contacto tiene que tener 8 números")  
    RETURN
```

```
ELSE
```

```
    Formulario.Contacto_Alta.Nif = NIF
```

```
END
```

```
IF NoSpace(Nombre) = "" THEN
```

```
    Info("Tiene que poner el nombre del contacto")  
    RETURN
```

```
ELSE
```

```
    IF Length(NoSpace(Nombre)) < 3 THEN
```

```
        Info("El nombre del contacto es demasiado corto")  
        RETURN
```

```
    END
```

```
END
```

```
Formulario.Contacto_Alta.Nombre = Nombre
```

```
Formulario.Contacto_Alta.Primer_Apellido = Primer_Apellido
```

```
Formulario.Contacto_Alta.Segundo_Apellido = Segundo_Apellido
```

```
IF Length(NoSpace(Email_contacto)) < 3 THEN
```

```
    Info("El email del contacto tiene que ser correcto")  
    RETURN
```

```
ELSE
```

```
    Formulario.Contacto_Alta.Email_contacto = Email_contacto
```

```
END
```

```
Formulario.Direccion_Alta.Calle = edt_calle
```

```
Formulario.Direccion_Alta.Pais = edt_pais
```

```
Formulario.Direccion_Alta.Poblacion = edt_poblacion
```

```
Formulario.Direccion_Alta.Provincia = edt_provincia
```

```
Formulario.Direccion_Alta.ZIP = edt_zip
```

```
Formulario.Direccion_Alta.Edif_Num = direccion_Edi fNum
```

```
Formulario.Direccion_Alta.Email = direccion_email
```

```
Formulario.Direccion_Alta.Teléfono = Telefono_direccion
```

```
Formulario.Direccion_Alta.Zona = direccion_zona
```

```
Formulario.Direccion_Alta.Planta = direccion_planta
```

```
Formulario.Direccion_Alta.Departamento = ""
```

```
IF Length(NoSpace(CIF_Empresa)) < 3 THEN
```

```
    Info("El CIF de la empresa tiene que ser correcto")  
    RETURN
```

```
END
```

```
Formulario.Empresa_Alta.CIF = CIF_Empresa
```

```
IF Length(NoSpace(Nombre_Empresa)) < 3 THEN
```

```
    Info("El Nombre de la empresa tiene que ser correcto")  
    RETURN
```

```
END
```

```
Formulario.Empresa_Alta.NombreCliente = Nombre_Empresa
```

```

Formulario. Usuario_Alta. Email=Cell_Usuario. Email
IF NoSpace(Cell_Usuario. Pass)=NoSpace(Cell_Usuario. Pass_repetido) THEN
    Formulario. Usuario_Alta. Pass=Cell_Usuario. Pass
ELSE
    Info("La clave de usuario no coincide.")
    RETURN
END

//SEGURIDAD:
//Si conocen la ruta de la página de registro podrán saltar la consulta
//del botón de "registro" en el login. Volver a consultar.

IF NOT gn_Controlador. Ctrl_Regla_Permittir_alta() THEN
    Info("Función deshabilitada temporalmente, disculpe las molestias")
    RETURN
END

IF gn_Controlador. SU_Nuevo_Cliente_Formulario(Formulario) THEN
    Info("En breve recibirá un email, Gracias.")

    Cell_Contacto. Cargo_usuario =""
    Cell_Contacto. Email_contacto=""
    Cell_Contacto. NIF=""
    Cell_Contacto. Nombre=""
    Cell_Contacto. Primer_Apellido=""
    Cell_Contacto. Segundo_Apellido=""
    Cell_Contacto. Telefono_contacto=""

    Cell_Dirección. dirección_Edificio=""
    Cell_Dirección. dirección_email=""
    Cell_Dirección. dirección_planta=""
    Cell_Dirección. dirección_zona=""
    Cell_Dirección. edt_calle=""
    Cell_Dirección. edt_pais=""
    Cell_Dirección. edt_población=""
    Cell_Dirección. edt_provincia=""
    Cell_Dirección. edt_zip=""
    Cell_Dirección. Telefono_dirección=""

    Cell_empresa. CIF_Empresa=""
    Cell_empresa. Nombre_Empresa=""

    Cell_Usuario. Email=""
    Cell_Usuario. Pass=""
    Cell_Usuario. Pass_repetido=""
ELSE
    Info("No ha sido posible crear su cuenta, Por favor contacte con Kernel
    Informática."+CR+"Disculpe la molestias ocasionadas.")
END

```

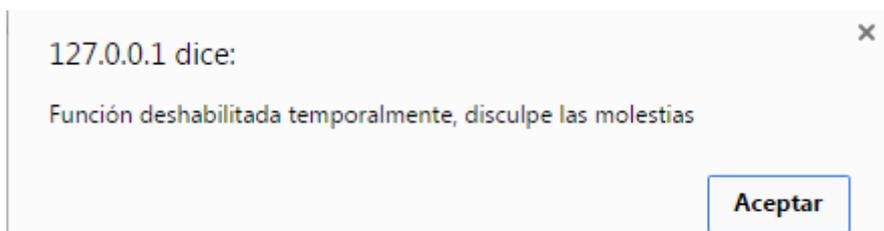


Ilustración 50: Aviso

## Captura de incidencia telefónica.

**Ilustración 51: Formulario de captura de incidencias del sistema SRIC**

Se observa el formulario de captura de incidencias. Al seleccionar un cliente SRIC notifica visualmente en rojo que el cliente no tiene contrato de mantenimiento en vigor. El usuario de atención, puede acceder a la ficha del cliente y ver la descripción donde el administrador ha escrito la razón, para más información. De todos modos tiene que informar al gerente.

En la parte superior central, SRIC informa visualmente con una campana, al usuario "Administrador", en este ejemplo, que tiene un mensaje pendiente. Si hace "Click" sobre la campana accede directamente al sistema de conocimiento (Tab: Gestión->Conocimiento) donde muestra en color Cyan la incidencia nueva o ya existente pero con comentario o algún cambio nuevo.

En la parte superior derecha, el pop-up fijo que muestra las llamadas en orden cronológico. Adjunta el nombre del cliente, Número de teléfono emisor y receptor y además información adicional, siendo "I\_" el termino "Incidencias" la estadística calculada en el instante y debajo el texto que contiene en ese momento el campo descripción en la ficha del cliente. Todo esto se muestra en el primer o segundo tono, mientras la llamada suena :

**I\_Abiertas:3 ; I\_En\_Resolucion:0 ; I\_Pendiente\_Proveedor:0**

**===== Descripción Cliente =====**

**El cliente se ha registrado via web**

Código servidor del evento "Click" del botón "Crear":

```
IF YesNo(No, "¿ Desea crear la incidencia de tipo  
"+COMBO_Ti po_Inci denci a. . Di spl ayedVal ue+" ?") THEN  
  
    Tab_Captura_Guardar_Inci denci a()  
  
END
```

Código del procedimiento servidor "Tab\_Captura\_Guardar\_Incidencia":

```
PROCEDURE Tab_Captura_Guardar_Inci denci a(captura_resuel ta is boolean = False)
```

```
Estructura_Inci denci a is ST_Inci denci a
```

```
Estructura_Inci denci a. Fecha_creaci on =  
tab_Menu_Pri nci pal . tab_Captura_Atenci ón. edt_fecha_captura  
Estructura_Inci denci a. Hora_Creaci on =  
tab_Menu_Pri nci pal . tab_Captura_Atenci ón. edt_hora_captura
```

```
Estructura_Inci denci a. Fecha_ci erre_Real = ""  
Estructura_Inci denci a. Hora_Ci erre_real = ""
```

```
//Calculada la estimación al seleccionar el tipo y prioridad de la incidencia.  
Estructura_Inci denci a. Fecha_MAX_Sol uci on =  
tab_Menu_Pri nci pal . tab_Captura_Atenci ón. edt_fecha_estimada_max  
Estructura_Inci denci a. Hora_MAX_Sol uci on =  
tab_Menu_Pri nci pal . tab_Captura_Atenci ón. edt_hora_estimada_max  
Estructura_Inci denci a. Fecha_MI N_Sol uci on =  
tab_Menu_Pri nci pal . tab_Captura_Atenci ón. edt_fecha_estimada_mi n  
Estructura_Inci denci a. Hora_MI N_Sol uci on =  
tab_Menu_Pri nci pal . tab_Captura_Atenci ón. edt_hora_estimada_mi n
```

```
//CALCULO AUTOMÁTICO DE FECHAS DE ESTIMACIÓN (Basado en reglas de negocio)
```

```
//Fecha y horas estimadas para el técnico  
debug_str is string =  
gn_Controlador. Ctrl_Cal cul o_Fecha_estimada(tab_Menu_Pri nci pal . tab_Captura_Aten  
ción. edt_fecha_captura, tab_Menu_Pri nci pal . tab_Captura_Atenci ón. edt_hora_captur  
a, tab_Menu_Pri nci pal . tab_Captura_Atenci ón. COMBO_Ti po_Inci denci a, tab_Menu_Pri nci  
pal . tab_Captura_Atenci ón. COMBO_Inci denci a_pri ori dad, "MIN_RES")  
Estructura_Inci denci a. Fecha_MI N_Resuel to=Left(debug_str, 8)  
Estructura_Inci denci a. Hora_MI N_Resuel to=Right(debug_str, 4)
```

```
debug_str =  
gn_Controlador. Ctrl_Cal cul o_Fecha_estimada(tab_Menu_Pri nci pal . tab_Captura_Aten  
ción. edt_fecha_captura, tab_Menu_Pri nci pal . tab_Captura_Atenci ón. edt_hora_captur  
a, tab_Menu_Pri nci pal . tab_Captura_Atenci ón. COMBO_Ti po_Inci denci a, tab_Menu_Pri nci  
pal . tab_Captura_Atenci ón. COMBO_Inci denci a_pri ori dad, "MAX_RES")  
Estructura_Inci denci a. Fecha_MAX_Resuel to=Left(debug_str, 8)  
Estructura_Inci denci a. Hora_MAX_Resuel to=Right(debug_str, 4)
```

```

//El técnico tiene que indicar estos datos

IF captura_resuelta THEN

    temp_datetime is DateTime= DateSys()+TimeSys()
    temp_date is Date
    temp_time is Time

    FOR i=1 TO
Val (tab_Menu_Principal.tab_Captura_Atención.edt_tiempo_atención)
        temp_datetime..Minute++
    END

    temp_date..Year=temp_datetime..Year
    temp_date..Month=temp_datetime..Month
    temp_date..Day=temp_datetime..Day

    temp_time..Hour=temp_datetime..Hour
    temp_time..Minute=temp_datetime..Minute

    Estructura_Incidencia.Fecha_Resolución_Real =DateToString(temp_date, "YYY
YMMDD")
    Estructura_Incidencia.Hora_Resolución_Real =TimeToString(temp_time, "HHM
M")

    Estructura_Incidencia.Fecha_Solución_Real = DateSys()
    Estructura_Incidencia.Hora_Solución_Real =TimeSys()

ELSE

    Estructura_Incidencia.Fecha_Resolución_Real = ""
    Estructura_Incidencia.Hora_Resolución_Real = ""

    Estructura_Incidencia.Fecha_Solución_Real = ""
    Estructura_Incidencia.Hora_Solución_Real = ""

END

Estructura_Incidencia.Agente_ID =
tab_Menu_Principal.tab_Captura_Atención.ComboBox_tecnicos
Estructura_Incidencia.Aplicativo_ID =
tab_Menu_Principal.tab_Captura_Atención.ComboBox_Captura_Aplicativo
Estructura_Incidencia.Asunto =
tab_Menu_Principal.tab_Captura_Atención.edt_titulo

Estructura_Incidencia.Cliente_ID =
tab_Menu_Principal.tab_Captura_Atención.ComboBox_Clientes
Estructura_Incidencia.Contacto_ID =
tab_Menu_Principal.tab_Captura_Atención.ComboBox_Contactos
Estructura_Incidencia.Creador_ID = ComboBox_Creador
Estructura_Incidencia.Descripción =
tab_Menu_Principal.tab_Captura_Atención.edt_TextoIncidencia
Estructura_Incidencia.Dirección_ID =
tab_Menu_Principal.tab_Captura_Atención.ComboBox_Dirección
Estructura_Incidencia.Dispositivo_Elemento_ID =
tab_Menu_Principal.tab_Captura_Atención.ComboBox_Dispositivo

Estructura_Incidencia.Incidencia_ID = - 1
Estructura_Incidencia.Minutos_atención =
tab_Menu_Principal.tab_Captura_Atención.edt_tiempo_atención
Estructura_Incidencia.Prioridad_ID =
tab_Menu_Principal.tab_Captura_Atención.COMBO_Incidencia_prioridad
Estructura_Incidencia.Tipo_Incidencia_ID =
tab_Menu_Principal.tab_Captura_Atención.COMBO_Tipo_Incidencia

IF captura_resuelta THEN

```

```

    IF HReadSeekFirst(Estado, Denominación, "Cerrada") THEN
        Incidencia.Estado_ID = Estado.Estado_ID
    ELSE
        gn_Depuracion.Log_Depuracion("Incoherencia - Error al asignar el estado a
la incidencia")
        Incidencia.Estado_ID = - 1
    END

ELSE

    IF HReadSeekFirst(Estado, Denominación, "Abierta") THEN
        Estructura_Incidencia.Estado_ID = Estado.Estado_ID
    ELSE
        gn_Depuracion.Log_Depuracion("Incoherencia -Error al asignar el estado a
la incidencia")
        Estructura_Incidencia.Estado_ID = - 1
    END

END

//NOTAS:
//1) Fecha y hora del último movimiento, se realiza posteriormente llamando al
controlador con el id incidencia.
//2) El campo aviso de incidencia se marcará true/false en
Ctrl_Nueva_Incidencia

IF gn_Controlador.Ctrl_Nueva_Incidencia(Estructura_Incidencia) THEN
    //Ha creado la incidencia con éxito, hay que marcar el último
movimiento
    IF
gn_Controlador.Ctrl_actualizar_fecha_hora_modificaciones(Estructura_Incidencia.In
cidencia_ID) THEN
        //Correcto, continúa.
    ELSE
        gn_Depuracion.Log_Depuracion("Error después de crear una incidencia no
permite actualizar su último movimiento, Incidencia
ID:"+Estructura_Incidencia.Incidencia_ID)
    END
ELSE
    gn_Depuracion.Log_Depuracion("Error al crear el registro Incidencia para Incidencia
ID:"+Estructura_Incidencia.Incidencia_ID)
END

IF Estructura_Incidencia.Incidencia_ID > 0 THEN

    Estructura_estado_Incidencia is ST_estado_Incidencia

    Estructura_estado_Incidencia.Incidencia_ID =
Estructura_Incidencia.Incidencia_ID
    Estructura_estado_Incidencia.EstadoID = Estructura_Incidencia.Estado_ID

    Estructura_estado_Incidencia.Descripción = "Creación de la incidencia ID:
"+Estructura_Incidencia.Incidencia_ID

```

```

IF
gn_Controlador.Ctrl_Nuevo_Estado_Incidencia(Estructura_Estado_Incidencia) THEN
ELSE
    gn_Depuracion.Log_Depuracion("Error al crear el registro Estado_Incidencia
para Estado id:"+Estructura_Incidencia.Estado_ID+", e Incidencia
ID:"+Estructura_Estado_Incidencia.IncidenciaID)
END

```

```

END

```

```

//Todo ha ido bien, reiniciar el formulario.

```

```

tab_Menu_Principal.tab_Captura_Atención.edt_Busqueda=""
ExecuteProcess(tab_Menu_Principal.tab_Captura_Atención.btn_busqueda, trtClick)

```

```

tab_Menu_Principal.tab_Captura_Atención.ComboBox_Dispositivo=0
tab_Menu_Principal.tab_Captura_Atención.ComboBox_Captura_Aplicativo=0

```

```

tab_Menu_Principal.tab_Captura_Atención.COMBO_Tipo_Incidencia=1
tab_Menu_Principal.tab_Captura_Atención.COMBO_Incidencia_prioridad=1

```

```

ExecuteProcess(tab_Menu_Principal.tab_Captura_Atención.edt_fecha_captura, trtInit)

```

```

ExecuteProcess(tab_Menu_Principal.tab_Captura_Atención.edt_hora_captura, trtInit)

```

```

ExecuteProcess(tab_Menu_Principal.tab_Captura_Atención.COMBO_Tipo_Incidencia, trtInit) //Inicializa la fecha y horas estimadas

```

```

ExecuteProcess(tab_Menu_Principal.tab_Captura_Atención.COMBO_Incidencia_prioridad, trtInit)

```

```

tab_Menu_Principal.tab_Captura_Atención.edt_titulo=""
tab_Menu_Principal.tab_Captura_Atención.edt_TextoIncidencia=""

```

```

Slider1.Value=1
edt_tiempo_atencion=1

```

```

tab_Menu_Principal.tab_Captura_Atención.edt_Busqueda=""
ExecuteProcess(tab_Menu_Principal.tab_Captura_Atención.btn_busqueda, trtClick)

```

2017/04/11:02:49:12 Farmacia Lopez Info 1  
2222=>3008

I\_Abiertas:3 ; I\_En\_Resolucion:0 ; I\_Pendiente\_Proveedor:0  
==== Descripción Cliente =====  
El cliente se ha registrado via web

Desconectar Administrador Tiene 1 mensajes pendientes.

KERNEL INFORMATICA

Recepción Gestión Administración Mantenimiento

Incidencias Clientes Informes

Conocimiento Ficha

Filtrar Cliente Agente Todos Prioridad Todas Ordenar por Últimos movimientos Desde: 05/06/2017

NOMBRE	DESCRIPCION	ESTADO	Creación	Estimación Sol.	Estimación Resol.	Ultimo Mov.
Farmacia Collado DBF Visual Fox Pro	AsAsas	Cerrada	12-06-2017 10:06	12-06-2017 17:07	14-06-2017 18:04	12-06-2017 10:19
Farmacia Collado DBF Visual Fox Pro	Buena impresion de la camara	Abierta	12-06-2017 10:16	12-06-2017 11:16	12-06-2017 12:16	12-06-2017 10:19
Farmacia DBF Visual Fox Pro	Incidencia Web ( Farmacia DBF Visual Fox Pro	Abierta	11-06-2017 19:44	12-06-2017 00:00	17-06-2017 11:20	11-06-2017 19:45

Comentario Nuevo comentario

Notificación - 2017/05/13 - 12:24  
Mensaje automático: el usuario Juan ha notificado, el parte de trabajo, por email al supervisor: Superjefe  
Emails notificados:fj.proteus@gmail.com

Notificación - 2017/05/13 - 11:37  
Mensaje automático: Se registra un cambio de estado en la incidencia, pasa a estar en estado->En resolución

Notificación - 2017/05/12 - 22:43

El servidor tiene mucha latencia en el arranque es imposible trabajar, se trata de una urgencia.

Ilustración 52 – Sistema de conocimiento y vista de los comentarios de una incidencia.

2017/04/11:02:49:12 Farmacia Lopez Info 1  
2222=>3008

I\_Abiertas:3 ; I\_En\_Resolucion:0 ; I\_Pendiente\_Proveedor:0  
==== Descripción Cliente =====  
El cliente se ha registrado via web

Desconectar Administrador Tiene 1 mensajes pendientes.

KERNEL INFORMATICA

Recepción Gestión Administración Mantenimiento

Incidencias Clientes Informes

Conocimiento Ficha

Busqueda Titulo Incidencia Web ( Farmacia DBF Visual Fox Pro

Clientes Farmacia DBF Visual Fox Pro Supervisor : SuperMonty

Direccion Central de Ingenio Dispositivo ACER server

Contactos perico Aplicativo Novophar

Tipo Comunicaciones Estado Abierta

Prioridad Normal Resolución

Creación 31/05/2017 13:48

Min. Solución 12/03/2017 00:00

Max. Solución 12/03/2017 00:00

Solución real 00:00 Ahora Descripción

Min. Resol. 31/05/2017 17:06

Max. Resol. 31/05/2017 18:06

Solución real 00:00 Ahora

-- Incidencia creada a través del portal web --  
Cliente:Farmacia DBF Visual Fox Pro  
Contacto:Farmablas@farmacia.com  
Necesito tinta para la impresora que está al lado del robot. Empson DX4359L

Gracias

El cliente ha realizado la encuesta

Ilustración 53: Ficha detallada de una incidencia, vista superior.



**Ilustración 54: Vista del cierre de una incidencia.**

Esta en la interface al indicar en la ficha de la incidencia el cambio de estado a “cerrado”. Se abre automáticamente mostrando los siguiente:

**Parte superior izquierda:** Título de la incidencia

**Recuadro central:** Redacción que acompañará al parte de trabajo. Lo verá el cliente.

**Parte inferior izquierda:** Conocida la resolución se puede aportar información que ayude al sistema de conocimiento a identificar incidencias que ayuden en el futuro a técnicos que estén en incidencias con algún nexo común.

**Campo de Emails alternativos:** Si hay que notificar puntualmente a alguien más, se insertan emails separados por “;”

**Previsualizar parte:** Descarga el parte en PDF para revisarlo

**Notificar al Supervisor:** Seleccionar un supervisor previamente. Envía al supervisor el parte para que lo apruebe y envíe al cliente.

**Enviar parte al cliente:** Adjunta el parte técnico pdf en un email como destinatario el cliente.

El código servidor del evento “click” del botón “Enviar parte al cliente”:

```
Procesar_Parte_Email ("Enviar_Cliente")
```

Dicho procedimiento servidor contiene este fragmento importante:

```
PROCEDURE Procesar_Parte_Email (Opcion is string ="Previsualizar")

.....

.....

CASE "Enviar_Cliente"

    Destinatarios is string =Registro_Incidencia.Email_Cliente
    Aviso is string=""

    IF Length(NoSpace(Text_emails_envio_reporte))>8 THEN
        //Hay correos adicionales
        Destinatarios =
        Destinatarios+";"+Text_emails_envio_reporte
        Aviso = "Incidencia en estado '
        "+gn_Controlador.Ctrl_Nombre_Estado_IDIncidencia(IDIncidencia)+" '
        "+...
        CR+"¿Esta seguro de querer enviar el parte de trabajo? "+CR+"Cliente:
        "+Registro_Incidencia.Nombre_Cliente+CR+"Direccion:
        "+Registro_Incidencia.Direccion_Cliente+CR+"Email:
        "+Registro_Incidencia.Email_Cliente+CR+CR+"Copias ocultas a las
        direcciones:"+Text_emails_envio_reporte

    ELSE
        //Sin destinatarios adicionales
        Aviso = "Incidencia en estado '
        "+gn_Controlador.Ctrl_Nombre_Estado_IDIncidencia(IDIncidencia)+" '
        "+CR+"¿Esta seguro de querer enviar el parte de trabajo? "+CR+"Cliente:
        "+Registro_Incidencia.Nombre_Cliente+CR+"Direccion:
        "+Registro_Incidencia.Direccion_Cliente+CR+"Email:
        "+Registro_Incidencia.Email_Cliente

    END

    IF NOT fFileExist(RutaPDF) THEN
        gn_Depuracion.Log_Depuracion("El parte de trabajo no se ha
        creado, no se puede enviar email al cliente "+Registro_Incidencia.Nombre_Cliente+", su
        ruta PDF es: "+RutaPDF+CR+CR+"Solucion: Revisar caracteres especiales en el nombre del
        cliente")
        Info("Error, el nombre del cliente contiene caracteres
        especiales,"+CR, "No puede crearse el fichero:"+RutaPDF)
        RETURN
    END

    IF YesNo(No, Aviso) THEN
```

```

        Campos_Email is ST_Param_Email

        Campos_Email.Mensaje_Cuerpo= "Estimado cliente,"+...
        CR+CR+CR+"Adjuntamos el parte de trabajo de la incidencia:"+CR+CR+...
        Registro_Incidencia.Titulo_Incidencia+CR+CR+". Atentamente."

        Campos_Email.Titulo_email = "Parte de trabajo -
        "+Registro_Incidencia.Titulo_Incidencia

IF gn_Controlador.Ctrl_Envio_Email (Campos_Email, Destinatarios, RutaPDF) THEN

        Mensaje_Comentario is string =
        gn_string_Cabecera_Comentario_auto+...
        "Mensaje automático: Se ha enviado el parte de trabajo al cliente."+CR+. . .
        "Emails notificados:"+Destinatarios

        IDComentario is int =
        gn_Controlador.CTRL_Nuevo_Comentario_Incidencia...
        (IDIncidencia, Mensaje_Comentario, True)

        Adjuntar_PDF_Parte_Comentario (IDIncidencia, IDComentario, RutaPDF)

        //Adjuntado en email, enviado. Además adjunto a un comentario
        //Se elimina el original
        fDelete(RutaPDF)

END

```

## Implementación Webservice

El webservice tiene una función principal donde procesa todas las llamadas, su nombre es "Estado\_Escucha", la cual tiene un diagrama mostrando su funcionamiento en la Ilustración 42. El código del método más importante del webservice es el siguiente:

```

PROCEDURE Estado_Escucha()

//Inicialización de variables.
//NOTA RutaSamba: Declarado Globalmente contiene la ruta de la carpeta
compartida de red

temp_registroXML is CuerpoXML
temp_registroCliDir is
ST_Cliente_Direccion

NFi chero, RestantesFich are strings
Iterar is
boolean =
True
Array_Ordenado is array of string

//Evitar que el siguiente código se ejecute concurrentemente mediante el uso
de un candado.

IF NOT Candado THEN

    //Ocupar candado Candado= True

    WHILE Iterar

```

```

//Solo esta iteración
Iterar=False

//Captura de la lista de ficheros con la identificación
de las llamadas entrantes.

RestantesFich =
fListFile(RutaSamba+"\*.txt", frNoHiddenDirectory+frNoHiddenFile+
frNotRecursive)

IF ExtractString(RestantesFich, 1, CR) = "" THEN
  //No hay ficheros que procesar

  //Realizar Backup si detecta cambio de día
  RSS_Es_Fin_Jornada()

  Candado = False
  RESULT "No hay ficheros de llamadas en el directorio "+RutaSamba+"\*.txt"+ "---
  descubre:"+ RestantesFich

END

//Limpiar todas las llamadas que puedan haber en el vector.
ArrayDeleteAll(Vector_Llamadas)
ArrayDeleteAll(Array_Ordenado)

//Ordena el listado de llamadas por orden cronológico
Debug_Test1(RestantesFich)
Array_Ordenado = Orden_Cronologico(RestantesFich)

FOR EACH NFi chero OF Array_Ordenado

  //Captura la información del cliente online u offline
  encapsulado en estructura ST_Cliente_Dirección

  temp_registroCliDir =
  Localizar_Cliente(Extraer_IDCALLER(NFi chero))

  //Registrar y actualizar llamada a llamada a medida que
  actualiza uno a uno el RSS - Gestiona la agenda Offline
  (add/modi fy)

  Registrar_Cliente_Dirección(temp_registroCliDir)

  //Inicializar el registro antes de insertarlo en el vector
  //que contendrá en cada posición un registro de llamada.

  temp_registroXML. RutaFi chero=NFi chero
  temp_registroXML. IDCALLER=Extraer_IDCALLER(NFi chero)
  temp_registroXML. IDRECEPTOR=Extraer_IDRECEPTOR(NFi chero)
  temp_registroXML. Fecha_de_Llamada= Extraer_Fecha(NFi chero)
  temp_registroXML. Hora_de_Llamada= Extraer_Hora(NFi chero)
  temp_registroXML. NCliente= temp_registroCliDir. Nombre
  temp_registroXML. Descripción=temp_registroCliDir. Vi sta_RSS

  ArrayAdd(Vector_Llamadas, temp_registroXML)

  //Crear cabecera XML a insertar en el RSS
  // utilizando XML 2.0
  RSS_2_Actualizar(temp_registroXML)

END

//Terminado el proceso de las llamadas, las almacena en base de
datos
// y mueve todos los ficheros

```

```

        procesador al Backup.
        Guardado_Llamadas()

        RSS_Es_Fin_Jornada()

    END

    Candado=False
    RESULT "El RSS ha sido actualizado"

ELSE
    Candado=False
    RESULT "El proceso de actualización del RSS está en curso. Inténtelo nuevamente en breve."

END

```

## Implementación de scripts en el sistema Linux - Asterisk

En la ejecución del dialplan de Asterisk, se ejecuta un script importante que contiene un algoritmo que llama a diferentes scripts. Su código es el siguiente:

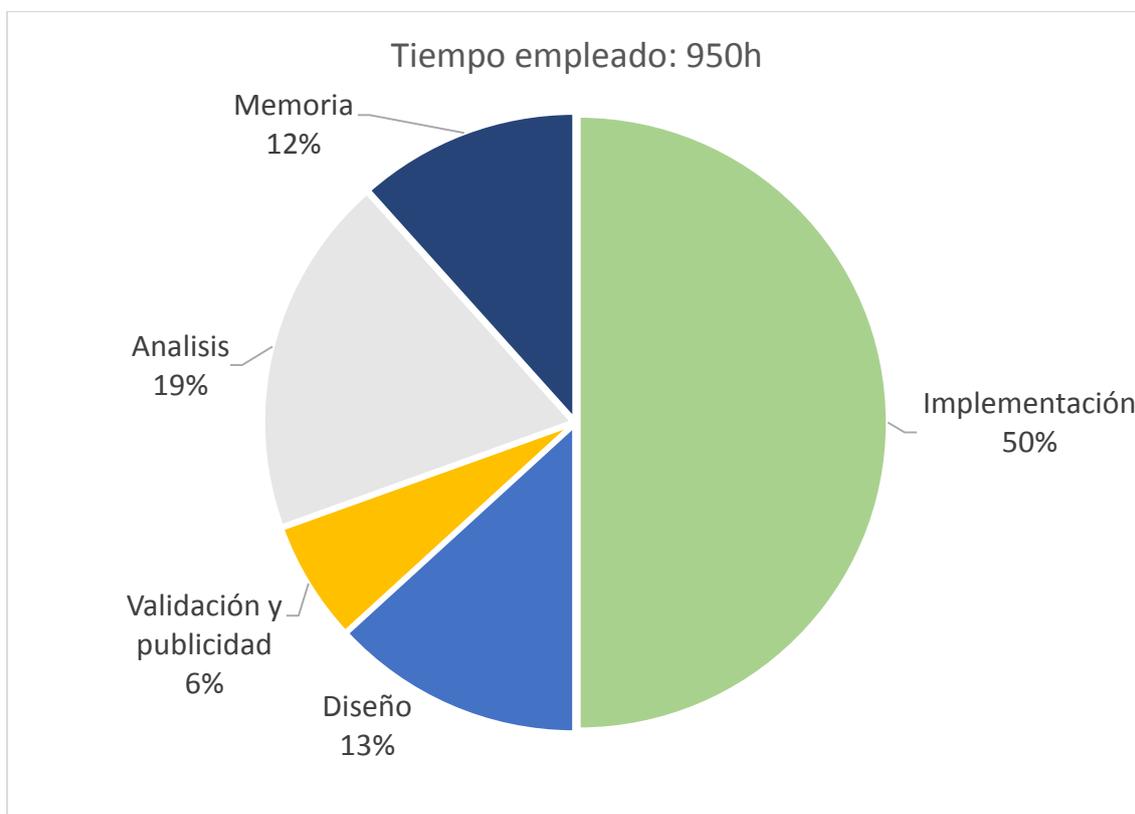
```

1  #!/bin/bash
2  if [ !-f /Samba/WinLlamadas/Baliza_linux.Inx ]
3  then
4
5      echo "No encuentra la carpeta de red. Montándola..."
6      sudo sh /Samba/startwin.sh &
7
8
9      if [ !-f /Samba/WinLlamadas/Baliza_linux.Inx ]
10     then
11         echo "Error de red...Iniciando copia offline."
12         touch /Samba/Llamadas_Offline/$1
13         echo "Copia offline realizada. Fin"
14         exit
15     fi
16
17 fi
18
19 echo "La carpeta de red está montada."
20 echo "Iniciando copia Online. Transferir las llamadas offline al server + Llamada actual."
21
22 touch /Samba/WinLlamadas/$1
23 mv /Samba/Llamadas_Offline/*.txt /Samba/WinLlamadas
24
25 echo "Llamada al webservice..."
26
27 sudo sh /Samba/wget.sh &
28
29
30
31 echo "Finalizado. "
32
33
34 exit
35

```

## Temporalización

El proyecto se ha planificado en las siguientes fases:



**Ilustración 55: Temporalización**

Las horas invertidas en cada una de las fases son las siguientes:

	Horas
Implementación	475
Diseño	125
Validación y publicidad	60
Análisis	180
Memoria	110
<b>Total:</b>	<b>950</b>

## Pruebas y mantenimiento

Pruebas en el aplicativo SRIC:

Para la realización de las pruebas unitarias se hará uso de dos herramientas: Debugger integrado de WebDev y el módulo de depuración propio de SRIC.

### El Debugger integrado en WebDev:

Este depurador facilita mucho las pruebas debido a la facilidad de uso para capturar excepciones, errores y valores incorrectos. Siendo utilizado especialmente para pruebas unitarias.

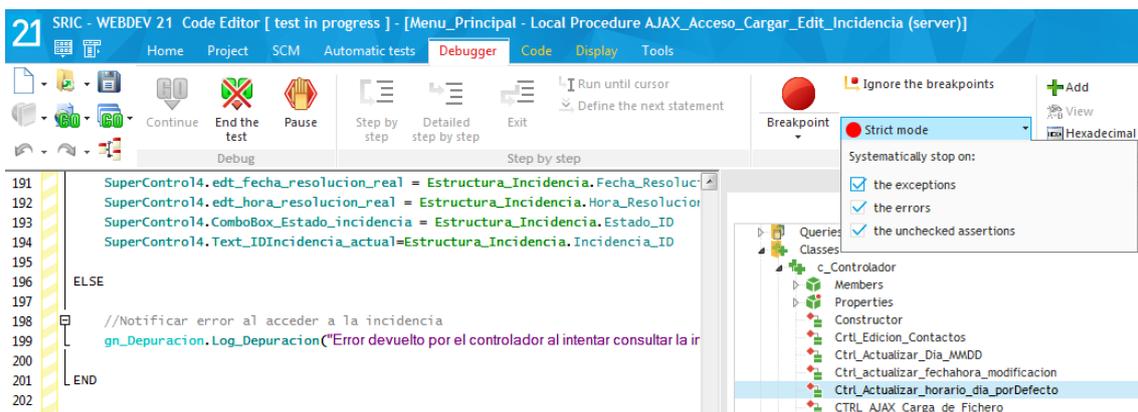


Ilustración 56: Depurador de WebDev

Las pruebas consistieron en ejecutar en código servidor las llamadas unitarias a cada uno de los métodos de cada objeto, comenzando la prueba por los objetos hojas, o sea, los contenidos dentro de la clase "Controlador", posteriormente se repite un nivel por encima que serían los objetos visibles: "Controlador" y "Depuración".

Los resultados devueltos por los métodos son observados en detalle por el visor del debugger de WebDev. En la siguiente captura se puede observar una prueba, se detiene en el breakpoint (punto rojo) mientras se obtienen todos los atributos del objeto Sesión durante el inicio de sesión.

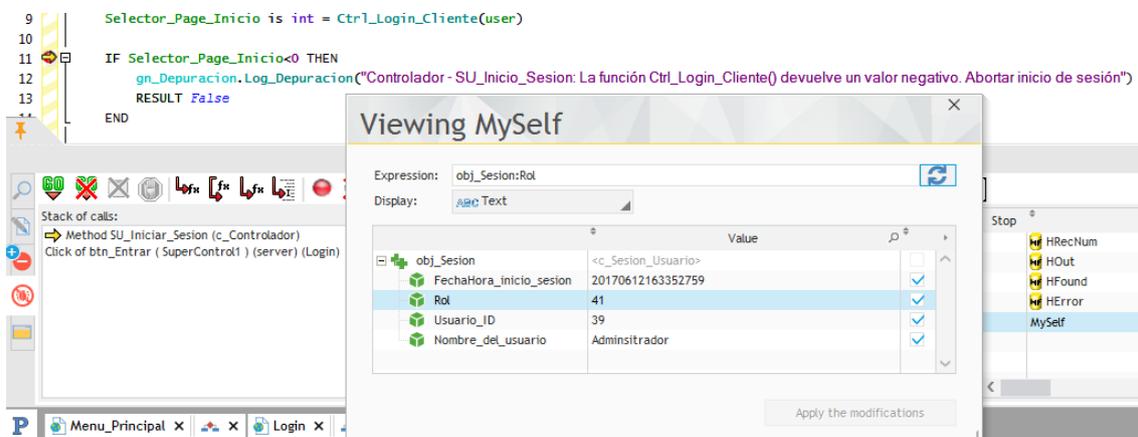


Ilustración 57: Depuración de un objeto y llamada al método Log\_Depuración( )

## Módulo de depuración propio de SRIC:

En el proyecto se ha implementado un depurador que permite crear un log de depuración cumpliendo con los requisitos funcionales.

Se basa en el patrón **Singleton**, ya que el log de depuración es común para todos los usuarios del sistema:

```
PROCEDURE GLOBAL Tomar_instancia()  
IF ::obj_Depuracion=NULL THEN  
  ::obj_Depuracion= new c_Depuracion()  
END  
RESULT ::obj_Depuracion
```

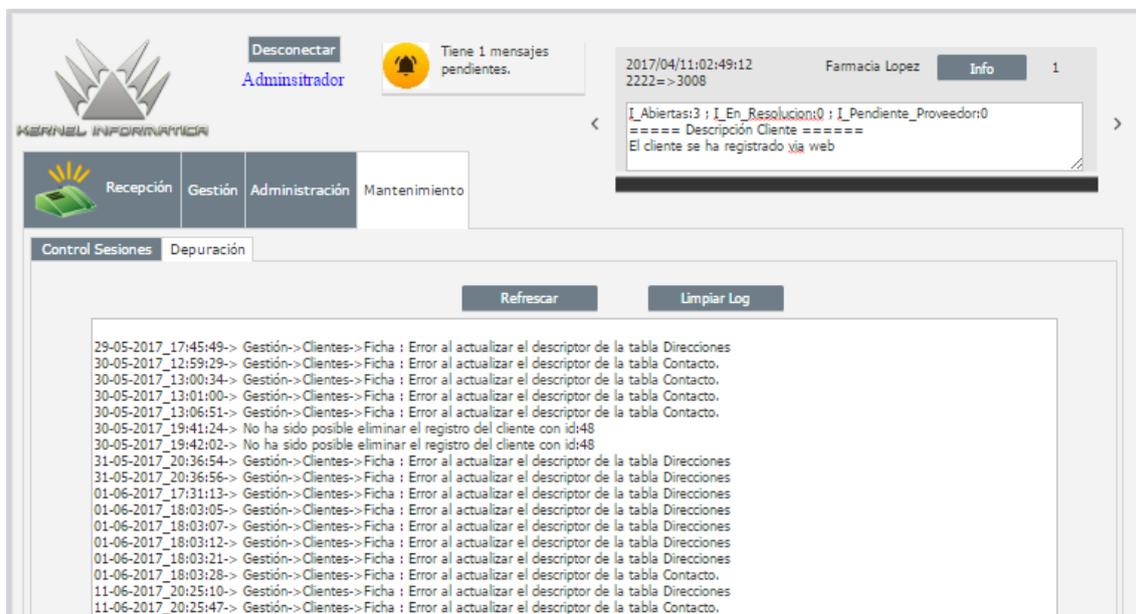
El método para depuración:

```
PROCEDURE Log_Depuracion(Mensaje string)  
IF Fdepuracion_ID<>- 1 THEN  
  
  fWrite(Fdepuracion_ID, Fecha_Hora_actual() + "-> "+Mensaje)  
  
END
```

Esta herramienta ha sido de gran ayuda durante la implementación y pruebas, destacando la de integración. Se ha insertado la llamada a este método, en puntos específicos del código donde suponga un punto crítico de riegos.

Como conclusión el módulo de depuración propio de SRIC ha detectado problemas semánticos, donde el depurador de WebDev no puede detectar, al no ser un error de parada.

A continuación un ejemplo de errores que no impiden trabajar con el aplicativo pero si indican errores de semántica interna y por consiguiente de estabilidad y rendimiento:



**Ilustración 58: Mantenimiento - Vista de la depuración interna del sistema SRIC**

Este es el resultado de cada llamada del método `gn_Depuracion.log_Depuracion("mensaje")`. Observar el código de la Ilustración 57.

## Pruebas unitarias y funcionales

Todos los requisitos funcionales se han tenido presente en estas pruebas ([Ed97], 1997).

Las pruebas empezaron por las clases contenidas dentro de la clase "Controlador".  
Testeo de los objetos haciendo uso de las dos herramientas citadas anteriormente:

- Objeto Vista: Todos los métodos pasan las pruebas correctamente.
- Objeto Sesion\_Usuario: Todos los métodos pasan las pruebas correctamente.

Se repite el proceso, quedando sólo las clases declaradas en el inicio del proyecto:

- Objeto Controlador: Todos los métodos pasan las pruebas correctamente.
- Objeto Depurador: Todos los métodos pasan las pruebas correctamente.

## Pruebas de Integración

Los objetos Vista y Sesion\_Usuario que son parámetros dentro del objeto Controlador han sido correctamente integrados, ya que se ha realizado controles en todas las llamadas de sus métodos desde el controlador. Siguiendo la siguiente metodología:

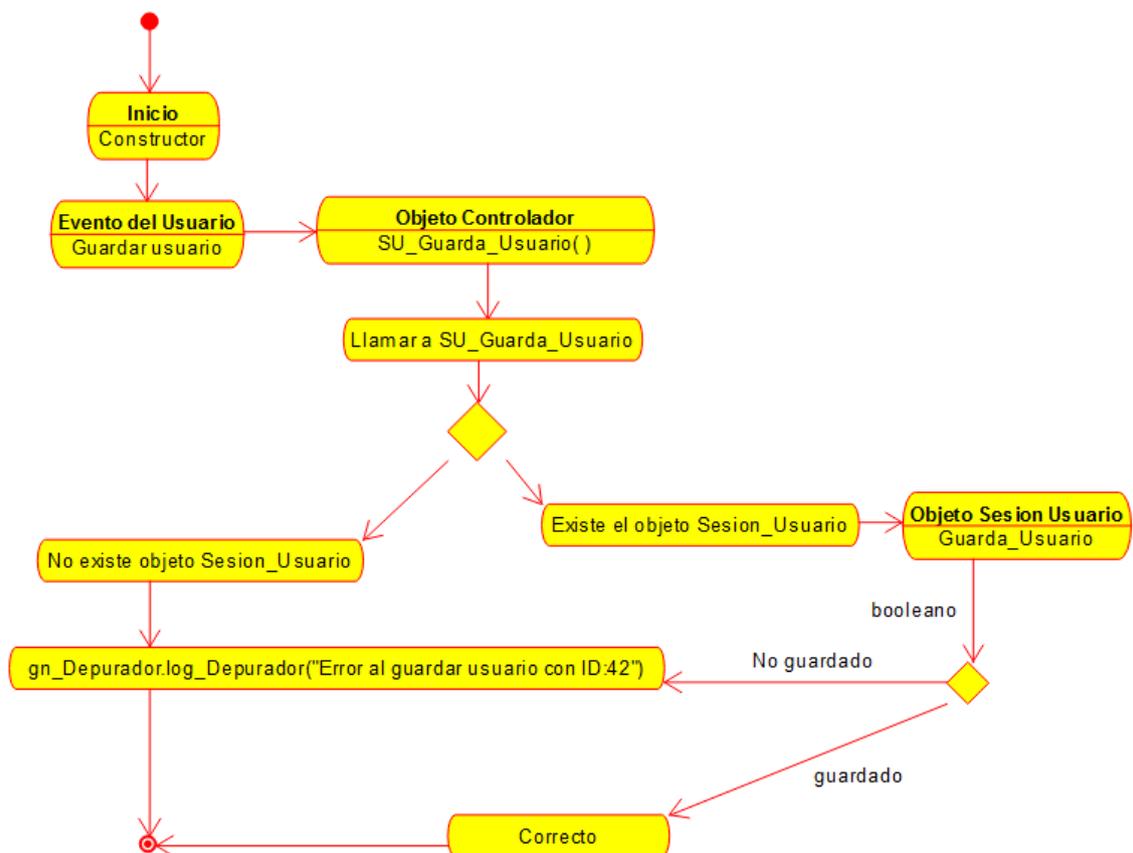


Ilustración 59: Diagrama de estado de una prueba de integración

## Pruebas de sistema

Se ha realizado todo el proceso de detección, captura, gestión de incidencia, cierre de incidencia y posterior generación de informe de parte de trabajo y encuesta. En general no ha habido errores que destacar, la estabilidad es correcta y el rendimiento óptimo.

## Pruebas en Webservice + Virtualbox::Centos::AsteriskNow

La realización del webservice ha sido en WebDev 21 por lo que tiene el mismo fundamento de depuración que SRIC.

La llamada principal del webservice es sólo un método, el encargado de procesar las llamadas. Por lo tanto no hay clases ni métodos, sólo este procedimiento que se secciona en varios estados.

Todos los estados y procesos que realiza han sido expuestos en la **Ilustración 42**.

## Pruebas unitarias y funcionales:

Han sido totalmente satisfactorias al separar en varias funciones externas los subprocesos pesados. Como resultado se testaron cada uno de los métodos externos, con tareas específicas, por ejemplo:

Actualizar\_RSS(...): El único propósito es actualizar el fichero RSS por lo tanto se puede realizar pruebas específicas. Todos los resultados fueron correctos. De igual modo con el resto de métodos externos.

## Pruebas de integración:

Después de la correspondiente integración mediante carpeta compartida con Linux Centos, el cual corre la máquina Asterisk, la cual deposita las llamadas e invoca el método del webservice, encargado del procesamiento de llamadas.

Todo el proceso ha sido exitoso ya que el resultado final es que las llamadas entrantes se muestren lo antes posible en el RSS, tras múltiples pruebas, todas han sido exitosas.

## Prueba de sistema:

Realizando varias pruebas llamadas desde softphone al servidor Asterisk, captura y deposita las llamadas desde Linux al servidor Windows, posteriormente llama al método (procesamiento de llamadas) del webservice. Realiza el procesamiento de varias llamadas a la vez con éxito ya que el fichero de llamadas RSS se actualizó exitosamente y muy rápidamente.

Destaca la prueba de rendimiento, donde el webservice tenía que procesar en una sola llamada a su método de procesamiento 500 ficheros de llamadas. Procesó y actualizó el fichero RSS en 5 segundos. Simulando llamadas que permanecieron offline en Linux por no tener conexión con la carpeta compartida con el servidor Windows.

## Implantación

En esta fase de desarrollo software se pretende comprobar el correcto funcionamiento de los aplicativos desarrollados dentro de un entorno real de operación.

Durante el proceso de implantación se necesita preparar y configurar toda la infraestructura necesaria para configurar el entorno, la instalación de componentes, la activación de los procedimientos manuales y automáticos asociados y cuando proceda la migración de datos o carga inicial.

Se realizan las pruebas de implantación y de aceptación del sistema en su totalidad. Con los siguientes propósitos:

- Las pruebas de implantación cubren un rango muy amplio, que va desde la comprobación de cualquier detalle de diseño hasta aspectos tales como las comunicaciones. Se debe comprobar que el sistema puede gestionar en su conjunto los volúmenes de información requeridos.
- Las pruebas de aceptación se realizan por y para los usuarios. Tienen como objetivo validar formalmente que el sistema se ajusta a sus necesidades.

Las pruebas de implantación y aceptación del sistema deben ejecutarse en el entorno real de operación. El propósito es comprobar que el sistema satisface todos los requisitos especificados por el usuario en las mismas condiciones que cuando se inicie la producción. La implantación del sistema inicial de cara al usuario ayudará en el proceso de refinamiento.

Si fuera necesario, se creará algún caso de prueba adicional que se considere importante y que no se haya tenido en cuenta hasta entonces. Se preparan las condiciones que permitan simular las situaciones límite previstas para las pruebas.

El objetivo de estas pruebas es asegurar que el sistema se comporta de la forma prevista en el entorno de operación, y que responde a todas las especificaciones dadas en cuanto al cumplimiento de los objetivos, tiempos de respuestas, seguridad, comunicaciones, etc.

## Instalación, configuración y carga inicial

Previamente, antes de generar los instalables del Webservice y de SRIC (Seguimiento y Registro de Incidencia de Clientes), hay que ejecutar, en los controles del entorno WebDev, los comandos de sincronización y refresco del proyecto. Con esto se pretende asegurar que se compila la última versión de los dos aplicativos.

Posteriormente hay que indicar durante la instalación en el servidor, las rutas donde ambos se van a alojar:

Intervienen los instalables de SRIC y Webservice (Procesador de llamadas):

1. Rutas de instalación (Indicado durante la instalación)

SRIC:

pathWeb: c:\PFCWeb\SRIC\

pathData: c:\PFCWeb\SRIC\data

Webservice:

pathWeb: c:\PFCWeb\SRIC\

pathData: c:\PFCWeb\SRIC\data

//Parámetro adicional

PathLlamadas: C:\PFCWeb\SRIC\Llamadas\_Asterisk

2. No hay más configuración a destacar, salvo avanzar e instalar primero SRIC y después el Webservice. Ambos aplicativos se instalarán en las siguientes rutas:

C:\PFCWeb\SRIC\SRIC\_WEB

C:\PFCWeb\SRIC\WS

Se ha indicado la siguiente ruta para situar los ficheros de bases de datos. Ambos aplicativos compartirán sus ficheros.

C:\PFCWeb\SRIC\data

Hay que asegurar que en la configuración del fichero **HTTPD.CONF** de WAMP (Servidor Apache) se defina correctamente una ruta raíz en "C:\PFCWeb\SRIC", de tal modo que permita al webservice y a SRIC acceder a la carpeta donde alojan sus ficheros de base de datos. No ha habido que realizar cambios, es correcta.

3. Las rutas de lectura de los INI son las siguientes:
  - a. Ruta para SRIC:
    - i. C:\PFCWeb\SRIC\SRIC\_WEB\SRICConfig.ini
  - b. Ruta para Webservice
    - i. C:\PFCWeb\WS\_PROCESARLLAMADAS\_WEB\WSConfig.ini

### Distribución de los directorios (después de la instalación)

El directorio raíz "C:\PFCWeb\SRIC" contiene:

- La carpeta "**data**":
  - Los ficheros de bases de datos de SRIC y webservice, están compartidos.
- La carpeta "**Llamadas\_Asterisk**":
  - Como indica su nombre, será la carpeta compartida en red con Linux. En ella se depositarán los ficheros de las llamadas. En encargado de transferir dichos ficheros es el sistema Linux mediante unos scripts que se encargarán de establecer la comunicación y de transferir los ficheros con los datos de las llamadas. El webservice leerá las llamadas desde esta carpeta.
- Finalmente, las dos carpetas restantes son los aplicativos:
  - "**SRIC\_WEB**": Carpeta creada durante la instalación. Contiene los ficheros y carpetas del proyecto.
  - "**WS\_PROCESADORLLAMADAS\_WEB**": Carpeta creada durante la instalación. Contiene los ficheros html y ejecutables del webservice.

### Comprobación de Base Datos

Verificación de los datos alojados en los tablas, evitar que tengan información de posibles testeos durante la fase de implementación, o bien, pruebas.

- i. Tablas de Registro\_Llamadas y Cliente\_Dirección correspondientes al webservice
  - ii. Comprobación del modelo de análisis de SRIC. Limpieza de registros de las tablas de datos. Las tablas de configuración deben estar correctas. Se pretende que se genere un modelo base donde poder comenzar utilizar el aplicativo SRIC desde cero.
- Nota importante:
    - Además, se crea manualmente la carpeta "**data\_init**", el cual **contiene los ficheros de base de datos inicializados para su primer uso**, tanto para el webservice como para el SRIC.

## Inicialización de carpetas y ficheros

Con el fin de facilitar el arranque del sistema inicial, se crea manualmente una carpeta llamada “**Restauración**”. Contendrá las siguientes carpetas con todos los ficheros de base de datos y de configuración necesarios para su primer uso:

### **Llamadas\_asterisk:**

Internamente tiene 3 carpetas, que son: carpeta “scripts”, contiene los scripts necesarios en el sistema Linux, la carpeta “XML” para la generación y actualización del fichero RSS (XML) y finalmente la carpeta “Prueba-cliente-Dir\_1\_2\_3” con 3 llamadas reales de ejemplo, para realizar una comprobación inicial de que el webservice trabaja correctamente.

En el interior de la carpeta “scripts” contiene los ficheros con extensión \*.sh, que son ejecutados en Linux, en el momento que Asterisk ejecute su dialplan durante una llamada entrante. Uno de los scripts se encargará de llamar al webservice para procesar las llamadas y actualizar el RSS.

La carpeta “XML” es donde se almacena:

**Fichero “XML\_RSS.xml”:** Cuando el webservice ha terminado de procesar los ficheros de las llamadas. Actualiza el fichero RSS con toda la información. Finalmente mueve los ficheros procesados a la carpeta de Backup de forma organizada.

**Backup de las llamadas:** Internamente se crea otra carpeta, llamada “Backup”. En ella se clasifica por año, mes y día, todas las llamadas del día. Estas se almacenan de forma automática cada jornada de trabajo. Al igual que el fichero RSS, se hace una copia de seguridad y se crea uno nuevo cada día.

### **INI\_Init:**

Siguiendo los pasos anteriores, esta carpeta contiene los dos INI con la configuración por defecto, listo para usarse. Si la ruta por defecto es cambiada, entonces hay que editar estos dos ficheros y colocar las nuevas rutas.

### **RSS\_Init:**

Contiene el RSS limpio.

### **Data\_Init:**

Contiene todos los ficheros de base de datos limpios, tanto del webservice como de SRIC.

A continuación se especifica el procedimiento para la inicialización del sistema:

1. Copiar la carpeta "Llamadas\_asterisk" en la ruta "C:\PFCWeb\SRIC".
2. Dentro de la carpeta INI\_init:
  - a. Copiar SRICConfig.ini y pegar en "C:\PFCWeb\SRIC\SRIC\_WEB".
  - b. Copiar WSConfig.ini y pegar en "C:\PFCWeb\SRIC\WS\_PROCESADOR\_LLAMADAS\_WEB".
3. Copiar el contenido de la carpeta "Data\_Init" y pegar en "C:\PFCWeb\SRIC\data"

Finalmente, el directorio raíz "C:\PFCWeb\SRIC" contiene 5 carpetas que son: "data", "Llamadas\_Asterisk", "Restauración", "SRIC\_WEB" y "WS\_PROCESARLLAMADAS\_WEB".

## Verificar la comunicación entre SRIC y el Webservice

### Comprobación del webservice

El webservice ha sido instalado y hay que comprobar que dentro de la carpeta "C:\PFCWeb\SRIC\Llamadas\_Asterisk\XML" no exista la carpeta "BACKUP", ya que guarda una copia de seguridad todas las llamadas. Si existe, hay que renombrarla, o bien, borrarla. Al igual que el fichero "XML\_RSS.xml", o sea, no tiene quedar nada dentro de la carpeta "XML".

Lo siguiente es copiar las 3 llamadas \*.txt de prueba situadas en "C:\PFCWeb\SRIC\Llamadas\_Asterisk\Prueba-cliente-Dir\_1\_2\_3", y pegarlas en "C:\PFCWeb\SRIC\Llamadas\_Asterisk".

Finalmente, acceder al navegador y colocar la siguiente dirección:

[http://\"IP del servidor\"/WS\\_PROCESARLLAMADAS\\_WEB/awws/index.htm](http://\)

Pulsamos en la opción "Estado\_Escucha" y luego en "test".

Tiene que obtener el mensaje "El RSS ha sido actualizado".

Si es así, el webservice trabaja correctamente. La vista del fichero “XML\_RSS.xml” contendrá las 3 llamadas de prueba, indicando “número desconocido” porque no hay clientes en la base de datos. Simula la llamada de los número ficticios “1111”, “2222” y “3333” a la extensión 3008.

```
<?xml version="1.0" encoding="UTF-8"?>
- <rss version="2.0">
  - <channel>
    <title>Número desconocido</title>
    <description/>
    <lastBuildDate>24 May 2017 18:16:27 Z</lastBuildDate>
    <link/>
    - <item>
      <author>2017/04/11:02:49:16 1111=>3008</author>
      <title>Número desconocido</title>
      <description/>
    </item>
  </channel>
  - <channel>
    <title>Número desconocido</title>
    <description/>
    <lastBuildDate>24 May 2017 18:16:27 Z</lastBuildDate>
    <link/>
    - <item>
      <author>2017/04/11:02:49:15 2222=>3008</author>
      <title>Número desconocido</title>
      <description/>
    </item>
  </channel>
  - <channel>
    <title>Número desconocido</title>
    <description/>
    <lastBuildDate>24 May 2017 18:16:27 Z</lastBuildDate>
    <link/>
    - <item>
      <author>2017/04/11:02:49:14 3333=>3008</author>
      <title>Número desconocido</title>
      <description/>
    </item>
  </channel>
</rss>
```

## Comprobación de SRIC

Después de su instalación, se acceder mediante el navegador a la siguiente dirección:

[http:// "IP del servidor"/SRIC/](http:// )

Aparece el portal web de login. Con lo cual significa que es correcto.

## Instalación y configuración de las máquinas virtuales en Virtualbox

### Configuración virtualbox

Se pretende instalar dos centralitas Asterisk en la máquina Virtualbox. La razón de que haya dos centralitas es para realizar hacer pruebas entre ellas posteriormente.

Cada centralita está instalada en Linux Centos. Desde la web oficial de AsteriskNOW es posible descargar la imagen ISO de Linux con el servidor Asterisk y el framework FreePBX, ya instalados. Por lo tanto, únicamente hay que configurar virtualbox de la siguiente manera:

**Paso 1)** Crear una nueva máquina virtual

**Paso 2)** Tipo: Linux debían 32 bits (ha sido compatible y estable)

**Paso 3)** Configurar opciones en Virtualbox para ambas máquinas.

- Sistema
  - Placa base:
    - RAM: 512MB
    - Orden de Arranque: Disco Duro, CD
    - Chipset PIIX3
    - Características extendidas: Habilitar I/O APIC + Reloj hardware en tiempo UTC.
  - Procesador:
    - Procesadores: 2
    - Limite ejecución: 100%
    - Habilitar PAE/NX
  - Aceleración
    - Habilitar VT-x/AMD-V
    - Habilitar paginación anidada
  - Audio
    - Deshabilitar audio
  - Red
    - Adaptador 1
      - Conectado a: "Adaptador puente":
        - Seleccionar tarjeta de red física del servidor
  - USB
    - Deshabilitar controlador USB

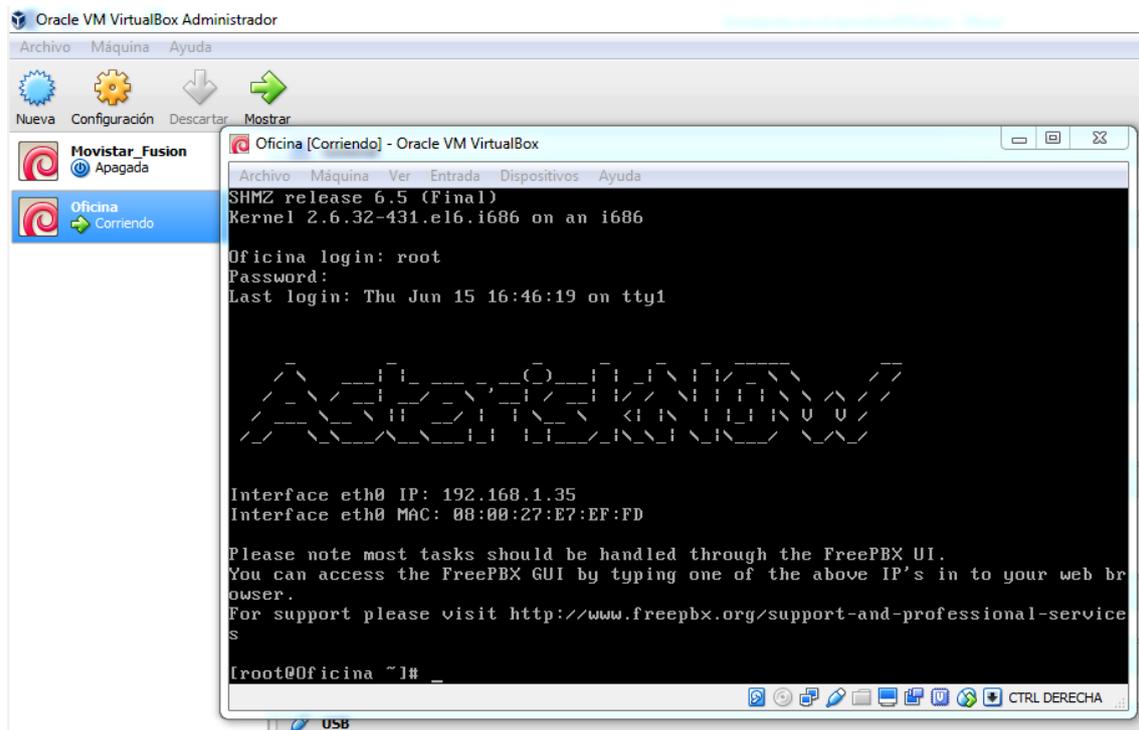
## Instalación de Centos 6.5 (AsteriskNOW) en Virtualbox

Una vez descargada la imagen de la web oficial de AsteriskNOW se procede a la instalación en la máquina virtual.

Hay que acceder a la configuración de Virtualbox, opción de almacenamiento e insertar una nueva unidad IDE. En la cual hay que seleccionar la imagen ISO.

Cuando se inicia la máquina virtual por primera vez, ejecutará el arranque desde CD y mostrando la pantalla de instalación de la distribución Centos.

Durante la instalación se define el usuario root y su contraseña del sistema Linux, necesarias para acceder al terminal una vez se haya reiniciado y en posteriores inicios del operativo.



**Ilustración 60: Máquina Virtualbox ejecutando AsteriskNOW en Linux**

La imagen corresponde a una máquina virtual con AsteriskNOW ya instalado. Una vez inicia el login del sistema Linux mostrará la dirección IP del interface de red "eth0". En el primer arranque, la red no está configurada, se explicará en el siguiente apartado.

En esta captura se observa la máquina virtualbox ejecutando una de las dos máquinas virtuales que aparecen a la izquierda, que son: "Movistar\_Fusion" y "Oficina". En este caso se ejecuta "Oficina". Como se observa es un Linux Centos 6.5 (Release) corriendo FreePBX Asterisk (AsteriskNow es una distribución Linux Centos que contiene Asterisk y FreePBX).

## Configuración de red en Centos

Durante la implantación surgen necesidades de configuración del entorno. En este caso es necesario configurar la red del sistema Linux.

Cuando se ha iniciado por primera vez la máquina virtual, la dirección IP del interface "eth0" no aparecerá. Esto se debe a que la dirección MAC de la máquina virtual es diferente a la que tiene el sistema Linux por defecto, por lo que hay que configurar la red.

La máquina virtual correrá dentro del servidor principal, donde irá el webservice y SRIC. No necesita ser localizado por red, por lo que se configura con conexión DHCP.

Accediendo al fichero:

**/etc/sysconfig/network-scripts/ifcfg-eth0**, indicando lo siguiente:

```
DEVICE="eth0"
HWADDR="00:FC:50:5E:DA:5B"
NM_CONTROLLED="yes"
ONBOOT="yes"
BOOTPROTO="dhcp"
```

En "HWADDR" indicamos la mac asignada por virtualbox al dispositivo de red virtual.

El Dispositivo "eth0" sigue sin estar presente, ya que hay que configurar el fichero:

**/etc/udev/rules.d/70-persistent-net.rules**, revisando que solo haya una línea para "eth0" y la MAC que se ha indicado en el fichero "ifcfg-eth0" es la correcta:

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="00:FC:50:5E:DA:5B",
ATTR{type}=="1", KERNEL=="eth*", NAME="eth0"
```

Finalmente hay que reiniciar la red para que eth0 solicite por DHCP una IP. Para verificarlo, se ejecuta los siguientes comandos que devuelven la dirección IP del interface "eth0" correctamente:

```
Bash> /etc/init.d/network restart
```

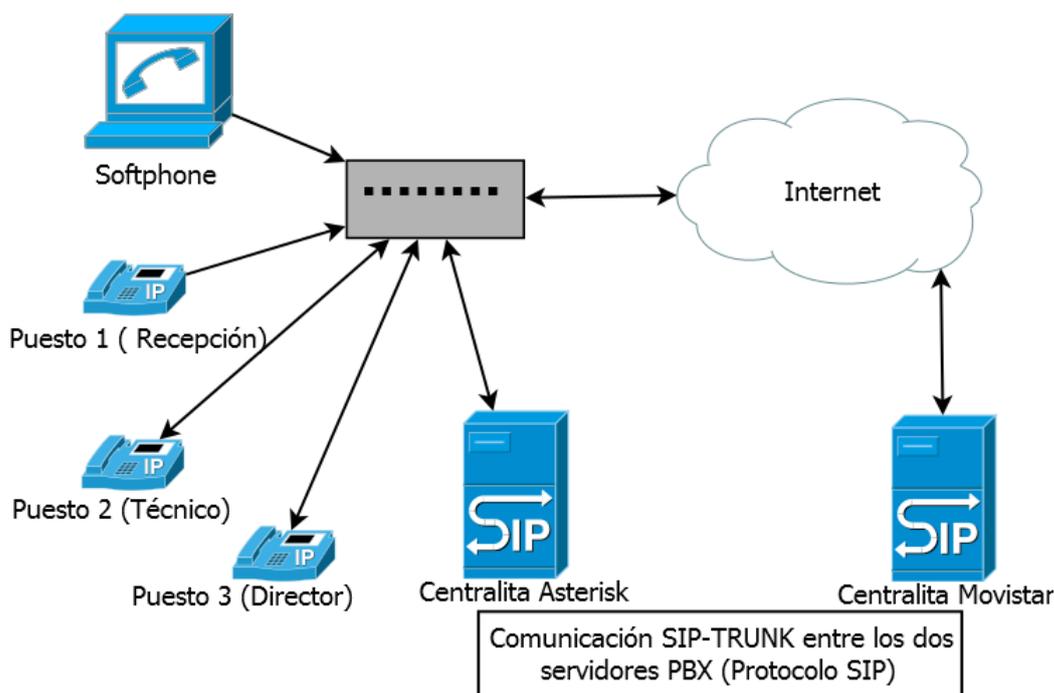
```
Bash> ifconfig
```

## Configuración de AsteriskNOW

Inicialmente Asterisk solicita la primera vez que se ejecuta, una clave para el usuario root del sistema Asterisk.

Posteriormente podemos acceder al portal de FreePBX de Asterisk por medio de la dirección IP del interface de red eth0, que nos aparece al inicio de sesión en el sistema Linux, tal como muestra la Ilustración 60.

A continuación una breve descripción de la conexión entre la centralita virtual Asterisk que se encuentra en el servidor principal donde se está realizando la implantación y la centralita telefónica virtual situada en la nube de Movistar.



**Ilustración 61: Conexión entre la centralita Asterisk (Oficina) y Movistar (Nube)**

Con esta configuración, la centralita Asterisk, a través del router de la empresa donde están conectados todos los teléfonos ip, utilizará su conexión SIP-TRUNK para autenticarse en la centralita telefónica de Movistar. Usando usuario y contraseña propios de una cuenta SIP. Todas las cuentas SIP están asociadas a una extensión telefónica. En esta extensión podrá escuchar todas las llamadas entrantes con la configuración adecuada en su dialplan.

## Simulación

Durante la fase de implantación, hay que realizar las pruebas de verificación que sean necesarias para asegurar el correcto funcionamiento de los sistemas. En este caso se considera importante la interconexión entre dos centralitas telefónicas.

Por lo tanto, se considera que es importante demostrar que es viable la comunicación entre dos centralitas telefónicas a través del protocolo Trunk-Sip ([RFC-SIP]), se realizará una simulación que lo demuestre. Los resultados se obtendrán de primera mano y será una prueba importante ya que se ejecutará en un entorno real de operación.

La primera centralita Asterisk simulará que se encuentra en la oficina de Kernel Informática S.L. La segunda es la centralita Movistar, de donde provienen todas las llamadas del exterior (Clientes, etc...).

Ambos servidores PBX son centralitas telefónicas que pueden conectarse mutuamente a través del protocolo SIP. Para conectar la centralita de la oficina con la de Movistar, debe comportarse como si de un teléfono IP se tratase. Para esta situación, el servidor 1 (**Ilustración 62**), utilizará su conexión SIP-TRUNK, que mediante usuario y contraseña se conectará por protocolo SIP (igual que los teléfonos IP) a la extensión 928998877 del servidor 2. Con esta acción ambos servidores estarán enlazados.

Se pretende completar los siguientes objetivos:

1. Que ambas centralitas se conecten mutuamente mediante protocolo SIP-Trunk.
2. Ejecutar una simulación **Ilustración 62**: El servidor 1 conectará con la extensión "928998877" del servidor 2 a través del protocolo Sip-Trunk. El servidor 2 permitirá que su extensión "927112233" (simulando por ejemplo a un cliente) llame a la oficina de Kernel Informática S.L (928998877). Como resultado el Servidor 1 recibe la llamada y la redirecciona a su extensión 3008 (por ejemplo: Recepción).

Aclarar que la conexión será dentro de la misma red local, sobre el mismo ordenador y ambas máquinas son Asterisk corriendo en el mismo virtualbox.



**Ilustración 62: Demostración de la conexión entre servidores mediante protocolo Trunk-SIP**

Se procede a configurar ambos servidores:

**1. Servidor 1:**

- Simula que es la centralita telefónica software situada en la oficina de Kernel Informática S.L
- Configurar con el usuario y contraseña de la conexión SIP-Trunk para el número 928998877 que es una extensión, o cuenta SIP, del servidor 2.
- Configurar la extensión 3008 del empleado de Kernel Informática S.L

**2. Servidor 2:**

- Simula que es la centralita virtual en la nube de Movistar.
- Crear dos extensiones, o sea, crear dos cuentas SIP normales para los números 927112233 y 928998877.
- NOTA: La cuenta SIP para el número 928998877 es la que utilizará el Servidor 1 mediante su SIP-Trunk para establecer la comunicación entre ambas centralitas telefónicas.
- NOTA: La cuenta SIP para el número 927112233 simulará un cliente que llama porque tiene una incidencia. En la realidad cualquier teléfono externo cuando realiza una llamada va a la red telefónica de Movistar. Posteriormente, se pasa la llamada por internet directamente a la centralita telefónica virtual Movistar en la nube. Justo es lo que se pretende simular en el servidor 2.

## Configuración de terminales mediante FreePBX GUI

FreePBX es el interface web para la configuración de Asterisk de forma gráfica. Se puede acceder a este portal web directamente desde el navegador, indicando la dirección IP que tiene el interface “eth0” en Linux Centos (Ilustración 60):

[http://IP\\_Centos](http://IP_Centos)



**Ilustración 63: Interface gráfica FreePBX del servidor Asterisk en la IP: 192.168.1.52**

Acceder a “FreePBX Administration” y autenticarse con la clave root indicada en la instalación, para acceder al panel de configuración del servidor Asterisk.

Se procede a configurar, las extensiones telefónicas de los servidores 1 y 2, tal como se ha indicado la Ilustración 62.

## Creación de los terminales/extensions en las centralitas

Inicialmente configuramos la centralita Movistar (**Servidor 2**). La cual contiene dos extensiones:

1. Extensión para el cliente
  - a. Numero: “927112233”
  - b. Nombre: “Farmacia Extremadura”
2. Extensión para llamar a la empresa Kernel Informática S.L:
  - a. Número: “92878998877”
  - b. Nombre: “Kernel Informática SL”

Por último la centralita Kernel Informática S.L, (**Servidor 1**):

1. Extensión para el empleado que atiende al cliente
  - a. Número: "3008"
  - b. Nombre: "Recepción"
2. Configurar el SIP-Trunk para conectar con la centralita Movistar
  - a. A través del SIP-Trunk de la centralita Kernel Informática SL, establecerá la conexión SIP con la extensión "928998877" de la centralita Movistar.
  - b. Habilitar el desvío de llamada entrante a la extensión 3008.

## Procedimiento para crear las extensiones en ambas centralitas

En primer lugar hay que acceder al menú de configuración. Acceder a configurar las extensiones. Seleccionar "Applications" → "Extensions". Aparecerá "Add an Extension".



**Ilustración 64: Configuración de los terminales mediante FreePBX.**

Acto seguido se selecciona el tipo de dispositivo. En el caso de la centralita Movistar, los terminales y centralita se basan en protocolo SIP y por lo tanto hay que seleccionar "CHAN SIP". Pulsar el botón Submit para continuar.

A continuación muestra un formulario donde hay que completar los siguientes campos:

1. **User Extension:** Número de teléfono asociado.
2. **Display Name:** Nombre con el que se identifica el dispositivo. Cuando el destino vea su llamada, aparecerá su nombre en la pantalla.
3. **Device options:**
  - a. Secret: Clave de acceso al terminal.
  - b. NAT Mode: Yes (force\_rport)
  - c. Context: El nombre del context es importante a la hora de personalizar, por defecto se indica "from-internal"
  - d. Port: 5060, dicho Puerto **tiene que estar abierto** tanto para udp como tcp en el router.

Por defecto el resto de campos del formulario, son válidos para realizar la prueba de conexión entre dos terminales SIP.

### Como desviar la llamada entrante (vía Trunk-sip) a la extensión de "Recepción" en la centralita "Kernel Informática SL"(3008)

Hay que acceder a la opción del menú "Conectivity" → "Inbound Routes". A la derecha seleccionar Route: "General". Al final del formulario en donde indica "set Destination".

Set Destination

Extensions ▼ <3008> Recepción ▼

Submit Clear Destination & Submit

Seleccionar la opción "Extensions", se habilita un combo donde se puede seleccionar el terminal "3008 - Recepción". Una vez seleccionado, pulsar Submit.

## Conectar dos centralitas Asterisk vía SIP Trunk->Extensión SIP

La centralita de Kernel informática (Servidor 1) tendrá configurada una línea Trunk SIP con los datos de la extensión creada en la centralita Movistar (Servidor 2). Concretamente la extensión SIP correspondiente a la cuenta “928998877” (Ilustración 62).

### Creación de una línea SIP Trunk

En el portal de la centralita de Kernel Informática S.L. Acceder a la opción del menú “Connectivity” → “Trunks”, tal como indica la siguiente captura:



**Ilustración 65: Configuración mediante FreePBX de una línea Sip-Trunk**

Posteriormente, pulsar en: [Add SIP \(chan\\_sip\) Trunk](#)

Indicar los siguientes parámetros:

4. General Settings → Trunk Name = “Conexión a la Centralita Movistar”

Borrar PEER Detail USER Detail

5. Outgoing Settings:
  - a. Trunk Name = 928998877-out
  - b. PEER Details = “sin datos”
6. Incoming Settings:
  - a. User Context = 92878998877
  - b. USER Details = “sin datos”
7. Registratión (**Ristra de conexión a la cuenta sip del servidor 2**)(Ilustración 62)
  - a. Register String: 928998877:prueba928998877@192.168.1.52/928998877
    - i. El formato es el siguiente:
      1. Extension:Clave@IP\_Movistar/Extension

Finalmente pulsar en Submit para aplicar los cambios.

## Configuración de los terminales (softphone)

Como se puede ver en la Ilustración 62 , el cliente utilizará un Smartphone con softphone y el empleado de Kernel Informática un softphone instalado en un PC.

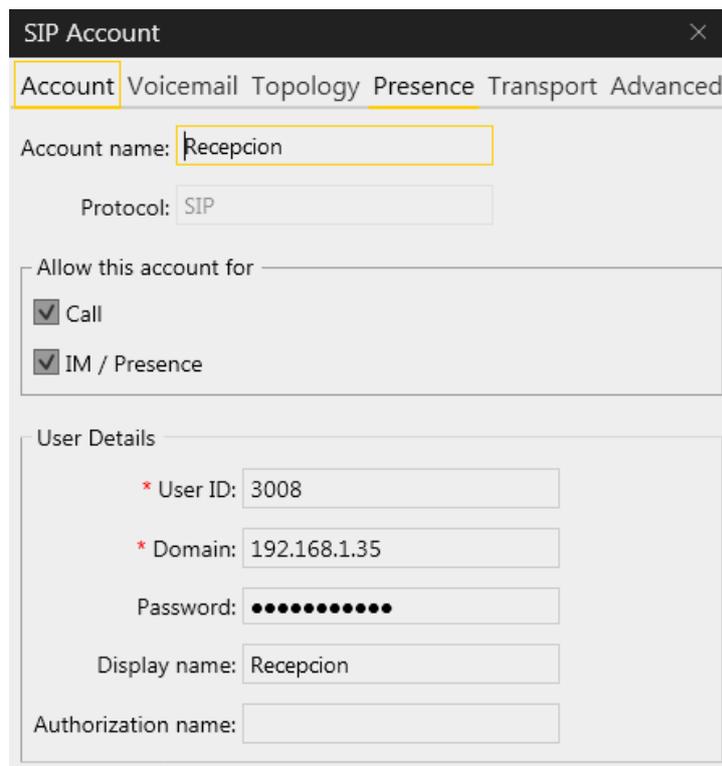
En el caso del softphone de PC, se utilizará el aplicativo X-Lite para Windows. El softphone de Smartphone, será en Android mediante la aplicación softphone Zoiper, obtenido vía google play. Ambas son versiones gratuitas.

Parámetros a configurar en ambos terminales:

1. Nombre Identificación de la cuenta SIP.
2. Dominio: IP de la centralita.
3. Usuario SIP definido en la extensión.
4. Contraseña asociada al usuario SIP.
5. Display name: Identifica su extensión con un nombre.

## Configuración X-Lite (Windows) como softphone de Kernel Informática S.L

- a. Dominio: Se conectará a la IP Centralita Kernel Informática
- b. Usuario: "3008"
- c. Contraseña: "teléfono\_del\_repcionista\_Kernel\_Informatica"



SIP Account

Account Voicemail Topology Presence Transport Advanced

Account name: Recepcion

Protocol: SIP

Allow this account for

Call

IM / Presence

User Details

\* User ID: 3008

\* Domain: 192.168.1.35

Password: ●●●●●●●●

Display name: Recepcion

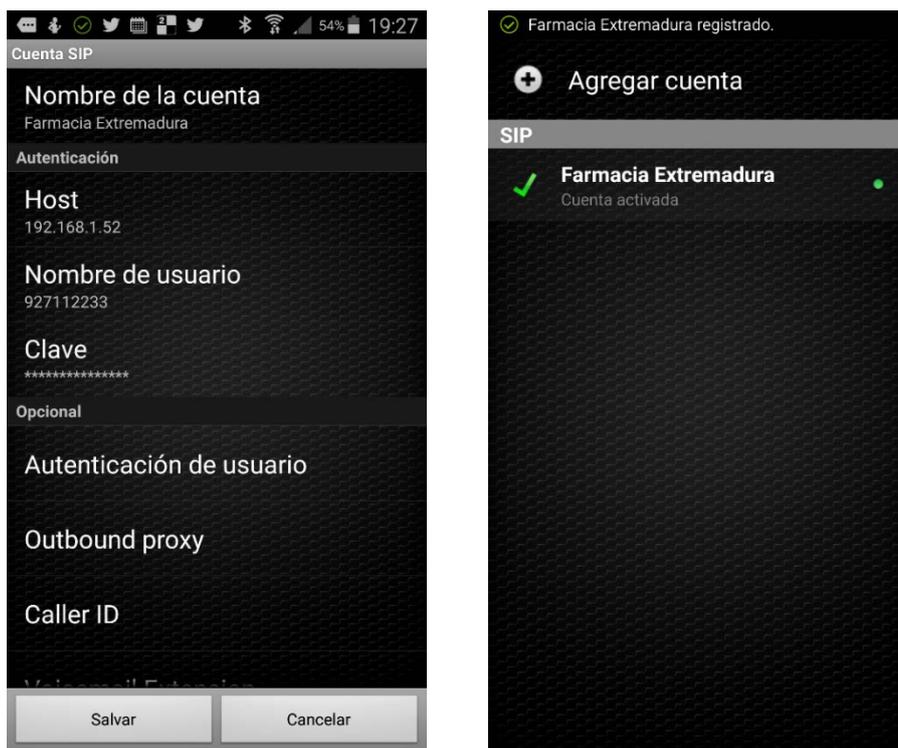
Authorization name:

Ilustración 66 Configuración del Softphone X-Lite en Windows

## Configuración Zoiper (Android), como softphone del cliente “Farmacia Extremadura”

- a. Dominio: se conectará a la IP Centralita Movistar
- b. Usuario: “927 11 22 33”
- c. Contraseña: “Cliente\_Farmacia\_Extremadura”

A continuación, se muestra las capturas de pantalla del aplicativo Zoiper, donde se puede observar que la cuenta SIP ha quedado activada en el Smartphone:



Se observa que después de introducir los datos, la extensión 3008 y el cliente “Farmacia Extremadura” se conectaron satisfactoriamente, cada uno a la IP de su centralita, como indica la Ilustración 62.

## Llamada entre centralitas.

En este punto ya se ha configurado y conectado las dos centralitas. Además de sus teléfonos IP correspondientes.

Con esta prueba se demostrará que los dos servidores PBX Asterisk mediante protocolo SIP pueden conectarse mutuamente. Habiendo utilizado el protocolo SIP-Trunk del servidor 1 (Ilustración 62) configurado con el usuario y contraseña de una extensión telefónica en el servidor 2. Como si se tratase de un teléfono ip más.

Una vez seguido todos los pasos indicados, se ha finalizado de configurar. De modo que se procede a hacer una prueba sencilla. Llamar desde el terminal de un cliente (por ejemplo, “Cliente-Farmacia Extremadura”) a la “recepción” de la oficina de Kernel Informática S.L.

Se efectúa la llamada desde el la Farmacia Extremadura (927112233) a Kernel Informática S.L (928998877). Siendo la prueba satisfactoria. Los dos teléfonos dan tono y al descolgar la comunicación es correcta. Se puede hablar bidireccionalmente y el sonido es de buena calidad. No ha surgido ningún problema. Por lo que se cuelga y finaliza la prueba.

## Personalización de Asterisk

Continuando con la verificación del sistema, se pasa a la siguiente etapa de la implantación. Demostrar en un entorno real de operación que la personalización del dialplan ([Dialplan], 2013) de Asterisk, obtiene los datos de cada llamada entrante correctamente.

El objetivo es capturar quien llama y transferir la información al webservice. Por lo tanto se procede a explicar la personalización del servidor Asterisk:

### Captura de llamada – personalización del Dialplan

Para la captura de las llamadas hay que realizar un desvío en el flujo de ejecución del Dialplan que tiene por defecto el servidor Asterisk.

Para empezar, explicar los dos ficheros relevantes de Asterisk, que son:

**Sip.conf:** Es el encargado de configurar todo lo relacionado con el protocolo SIP y añadir nuevos usuarios o conectar con proveedores SIP (Conexiones Sip-Trunk). En la Ilustración 65 hemos definido los parámetros necesarios para las pruebas, mediante el framework FreePBX de Asterisk.

**Extensions.conf:** Define el comportamiento que va a tener una llamada en nuestra centralita, las reglas que siguen en su enrutamiento. Es por ello, que será necesario alterar el flujo principal de ejecución de reglas o Dialplan para insertar unas subrutinas que permitan la captura de las llamadas entrantes.

La personalización no es posible directamente sobre `extensions.conf`, debido a que todos los ficheros importantes son controlados por el framework FreePBX, y toda personalización que se realice será sobrescrita de forma automática.

La arquitectura de Asterisk contiene los ficheros de personalizaciones preparados y definidos, pero sin configuración alguna (comentados y vacíos). Por lo tanto se procede a explicar cómo realizar la personalización del flujo de entrada de una llamada.

## Análisis y configuración del fichero `Extensions.conf`

La primera cabecera del Dialplan tiene que ver con PSTN y como desvía la llamada a otras cabeceras para su procesamiento.

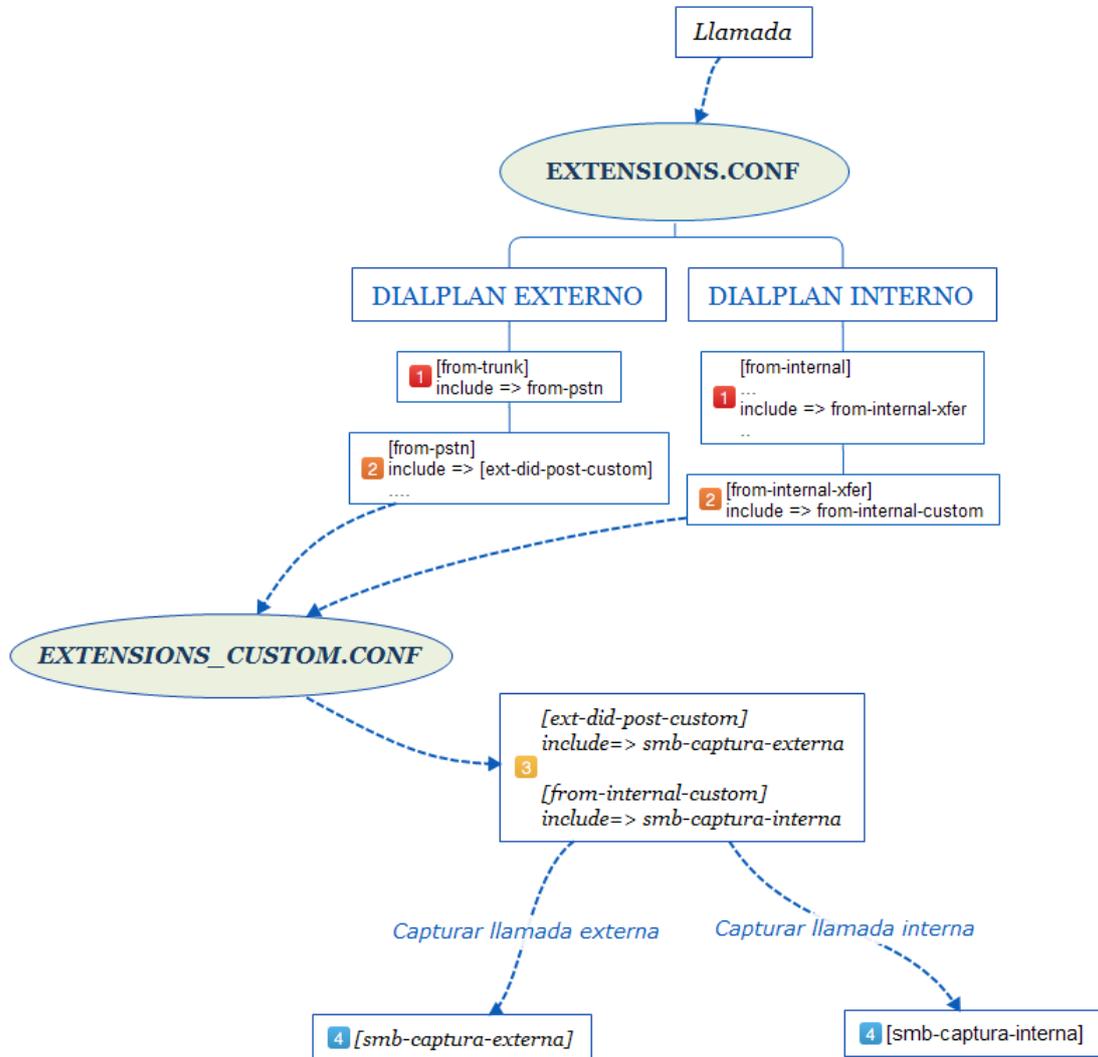
La red telefónica pública conmutada (**PSTN**, Public Switched Telephone Network) es una red con conmutación de circuitos tradicional optimizada para comunicaciones de voz en tiempo real. Cuando llama a alguien, cierra un conmutador al marcar y establece así un circuito con el receptor de la llamada.

Cuando hay una llamada entrante, esta bifurca según su procedencia:

**Dialplan Externo:** Si la llamada proviene del exterior, o sea, que proveniente de una red externa. En este caso la llamada es captada por medio de su conexión Sip-Trunk, que estará configurada y conectada a su proveedor de telefonía.

**Dialplan Interno:** La llamada es interna a la red de la centralita, o sea, es una llamada de una extensión a otra extensión de la misma centralita telefónica.

Se ha creado un esquema que muestre los dos flujos de entrada a la centralita y como se realiza la personalización, de tal modo que en fichero **Extensions\_custom.conf** se ejecuten las órdenes personalizadas a partir del punto 3:



**Ilustración 67 Dialplan personalizado para capturar la llamada**

En el paso 4, se ejecutarán las órdenes para obtener los siguientes valores:

1. Día, mes y año de la llamada.
2. Hora, minutos y segundos de la llamada.
3. Identificador numérico de llamada entrante.
4. Identificador numérico de la extensión que tiene como destino.

## Comandos para capturar la llamada a partir del punto 4.

Para la extracción de información de la llamada, dentro del Dialplan, se ha creado el script "Procesar\_Registro.sh". (**Ilustración 44**)

El propósito de este script es el siguiente:

1. Desde Linux se establece comunicación con el servidor Windows donde se aloja el webservice, conectando con carpeta compartida de red donde lee el webservice.
2. Generar un fichero vacío con toda la información estructurada en el nombre del fichero: "AAMMDD\_HHMM\_Tlfn1\_to\_Tlfn2.txt". "Tlfn1" es quien llama y "Tlfn2" es quien recibe la llamada.
3. Será el responsable de mantener una copia de seguridad de todas las capturas, tanto si tiene conexión con el servidor (Backup Online) como si no tiene (Backup Offline).
  - 1) **Backup Offline:**
    - i. Encargado de mantener guardado todos los ficheros de las llamadas, hasta que se reestablezca la comunicación con la carpeta compartida en el servidor Webdev. En ese momento, volcará todas las llamadas almacenadas, incluida la llamada actual, para que el webservice las registre en base de datos y lo procese ordenadamente.
  - 2) **Backup Online:**
    - i. Hay comunicación con el servidor Webdev. El script deposita en la carpeta compartida de llamadas ("**Llamadas\_asterisk**"), que se ubica en el servidor Webdev, todos los ficheros del Backup Offline, si hubiera. Además de la llamada actual. Finalmente, ejecuta la llamada al webservice para que lo procese y actualice el fichero RSS de llamadas y su base de datos compartida.

“Procesar\_Registro.sh” contiene un algoritmo que hace uso de otros scripts que realizan tareas más sencillas, son los siguientes:

- 1) **startwin.sh**: Encargado de montar la carpeta de red llamada “WinLlamadas”. Necesita previamente tener configurado la carpeta “Llamadas\_asterisk” en el servidor Webdev, con los permisos necesarios para montar la carpeta mediante credencial (usuario y contraseña).



Surge un problema importante, al montar la carpeta de red sobre una carpeta ya existente en Linux. El problema surge una vez se ejecuta el comando “mount”, no hay manera de saber si tuvo o no éxito.

Después de varias ideas, ha surgido una solución viable. Para comprobar si ha montado la unidad correctamente se recurre a un método muy simple. Consiste en colocar un fichero llamado “Baliza\_linux.Inx” (creado manualmente) en la carpeta compartida del servidor Webdev, de tal modo, que al ejecutar “startwin.sh”, consulte si dicho fichero existe, si es así, ha montado la unidad con éxito. Tras muchas pruebas la solución es robusta y no ha fallado.

Surge la necesidad de configurar los permisos de la carpeta de red en el servidor Windows donde se aloja el webservice.

Los permisos son los siguientes, una vez se accede a las propiedades de la carpeta:

- Agregar los nombres: “Administradores” y “Todos”
- Nivel de permiso: Lectura y escritura
- Uso compartido avanzado – Permisos:
  - Usuario → “Todos”: “Control total(Cambiar,Leer)”
- Nota: Con esta configuración, al acceder por red a la carpeta compartida, solicitará el usuario y la contraseña, o sea, la acreditación. El script “startwin.sh” conoce el usuario y la contraseña.

- 2) **wget.sh**: Su propósito es ejecutar el comando “wget” cuyo parámetro es la ruta del servidor Webdev donde invoca el método “Estado\_Escucha” del webservice que procesa la llamada y actualiza el fichero RSS. Es el último comando que se ejecuta en Procesar\_Registro.sh.



Para facilitar la tarea de conocer la dirección del servidor. Si no es una IP fija, existe la posibilidad de configurar un servidor de nombres de red, ya contenido en Linux, llamado WINS (Windows Internet Naming Service). Este servidor ([WINS]) mapea los nombres de las máquinas a direcciones. Se ha decidido configurar esta opción por comodidad. No hay que instalar, ya lo tiene instalado.

Inicialmente al hacer ping “Nombre\_máquina” se obtiene:

```
unknow host “Nombre_máquina”
```

A continuación se edita “/etc/nsswitch.conf” y se añade “wins” en la siguiente línea:

```
hosts:          files wins dns
```

A partir de ahora al hacer ping al nombre de una máquina responderá con su IP y latencia, por ello el script “wget” puede ahora usar la ruta “//Nombre\_servidor/...” como parámetro.

### Prueba de captura de llamada (solo script – sin dialplan)

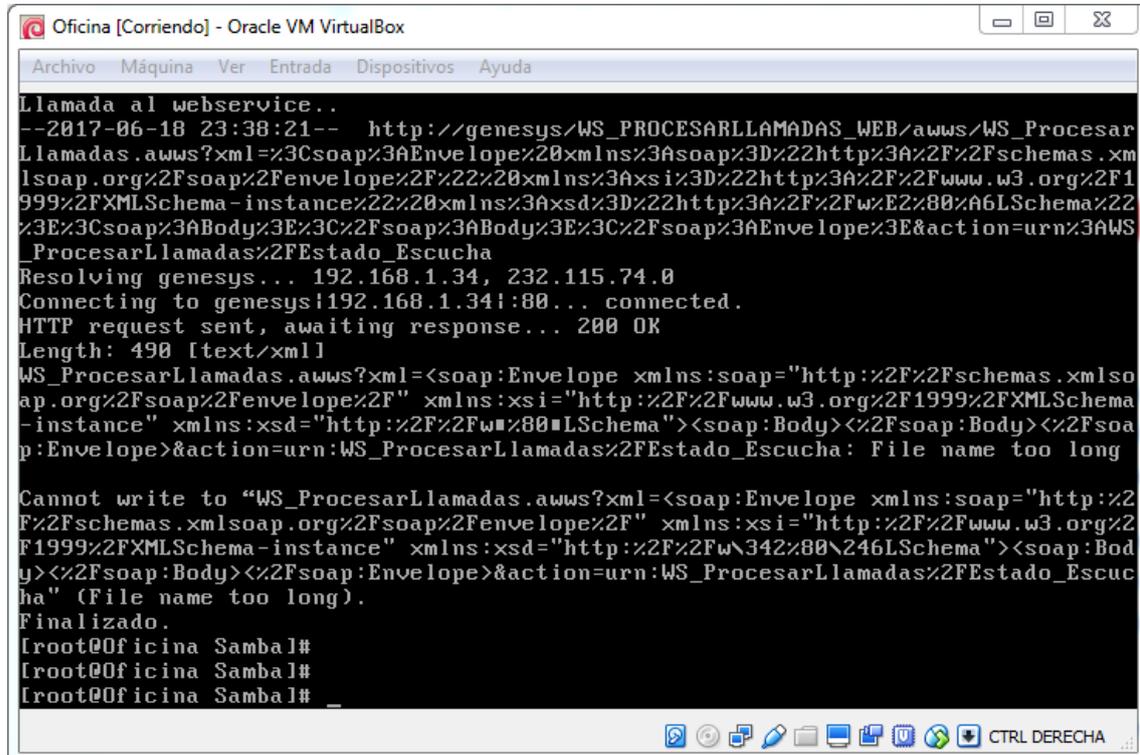
La prueba consiste en ejecutar el script “Procesar\_Registro.sh” que se lanzará desde el Dialplan de Asterisk en el momento que entre una llamada. Para capturar los mensajes que emite el script, a modo de depuración, se ha volcado la salida de la siguiente manera:

```
Sudo sh Procesar_Registros.sh> Salida.txt, obteniendo lo siguiente:  
  
La carpeta de red está montada.  
  
Iniciando copia Online. Transferir las llamadas offline al server + Llamada actual.  
  
Llamada al webservice..  
  
Finalizado.
```

Como se puede observar, ha detectado que la unidad de red ya estaba montada y ha procedido a copiar las llamadas offline y la llamada actual sobre el servidor Windows. Donde se encuentra el webservice. Finalmente, llama al método del webservice para procesar las llamadas.

A continuación, se muestra un ejemplo de la llamada al webservice donde:

1º) Se invoca mediante una ristra que contiene una estructura compleja en XML, al webservice y a su método “Estado\_Escucha” (método encargado de procesar las llamadas que se encuentren en la carpeta compartida)



```
Oficina [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Llamada al webservice..
--2017-06-18 23:38:21-- http://genesys/WS_PROCESARLLAMADAS_WEB/awws/WS_Procesar
Llamadas.awws?xml=%3Csoap%3AEnvelope%20xmlns%3Asoap%3D%22http%3A%2F%2Fschemas.xml
lsoap.org%2Fsoap%2Fenvelope%2F%22%20xmlns%3Axs%3D%22http%3A%2F%2Fwww.w3.org%2F1
999%2FXMLSchema-instance%22%20xmlns%3Axsd%3D%22http%3A%2F%2Fw%2E%80%A6LSchema%22
%3E%3Csoap%3ABody%3E%3C%2Fsoap%3ABody%3E%3C%2Fsoap%3AEnvelope%3E&action=urn%3AWS
_ProcesarLlamadas%2FEstado_Escucha
Resolving genesys... 192.168.1.34, 232.115.74.0
Connecting to genesys[192.168.1.34]:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 490 [text/xml]
WS_ProcesarLlamadas.awws?xml=<soap:Envelope xmlns:soap="http://schemas.xmls
ap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2003/XMLSchema-instance" xmlns:xsd="http://w
w.w3.org/2003/XMLSchema"><soap:Body><soap:Body></soap:Body></soap:Envelope>&action=urn:WS_ProcesarLlamadas%2FEstado_Escucha: File name too long

Cannot write to "WS_ProcesarLlamadas.awws?xml=<soap:Envelope xmlns:soap="http://schemas.xmls
ap.org/soap/envelope/" xmlns:xsi="http://www.w3.org/2003/XMLSchema-instance" xmlns:xsd="http://w
w.w3.org/2003/XMLSchema"><soap:Body><soap:Body></soap:Envelope>&action=urn:WS_ProcesarLlamadas%2FEstado_Escucha" (File name too long).
Finalizado.
[root@Oficina Samba]#
[root@Oficina Samba]#
[root@Oficina Samba]# _
```

Ilustración 68: Prueba - Llamada al webservice

2º) Como se puede observar, se realiza la petición a la dirección [http://genesys/WS/PROCESARLLAMADAS\\_WEB/...](http://genesys/WS/PROCESARLLAMADAS_WEB/...) Donde “genesys” es el nombre de red de la máquina de ejemplo, no una dirección IP ([WINS]). Se observa que funciona correctamente el servidor “WINS”, indicando “*resolving genesys*” en la captura, significa que realiza la consulta en la tabla de nombres del servidor “WINS”, obteniendo su dirección IP.

3º) Establece la conexión real con el servidor de nombre “genesys” y realiza la consulta.

El script “Procesar\_Registro.sh” realiza correctamente sus funciones, tal como indica el procedimiento de la Ilustración 44: Captura de llamada mediante Asterisk y procesamiento del webservice.

## Prueba del Dialplan

La prueba consiste en realizar la llamada entre dos teléfonos IP, una vez aplicada la personalización del Dialplan.

Durante la prueba surge un fallo, el terminal al cual se llama no recibe tono y el termina que está llamándole recibe un *error 500*. Después de analizar el log del debugger en el servidor Asterisk, se detecta una latencia en un punto concreto del Dialplan, que coincide con el script “Procesar\_Resgistro.sh”.

La latencia de este script provoca el **timeout** de la centralita telefónica Asterisk, al considerar que la llamada entrante se ha detenido en el camino crítico de ejecución del Dialplan. La documentación ([TOUT]) especifica que el código de error es aleatorio al tratarse de una excepción.

El problema radica en el tiempo de espera para montar la carpeta de red compartida y llamar al webservice. Al estar todas las tareas dentro de un solo script, provoca el timeout. Para solventar este problema se recurre a la ejecución del script en segundo plano.

La ejecución de un script en segundo plano en un sistema Linux es simple, únicamente añadir al final de cada script el símbolo “&”.

Después de aplicar la solución no ha habido más problemas de timeout o interrupciones inesperadas en el Dialplan. Los terminales tienen tono y al descolgar el sonido es bidireccional. El registro de la llamada en fichero funciona correctamente. Por lo tanto, se ha superado la prueba.

## Prueba de Backup Online/Offline (interrelación entre Scripts y webservice)

El propósito de la prueba es ejecutar el script y ver si cumple con los requisitos cuando suceden las siguientes situaciones:

- 1) Resultado con el Webservice detenido y carpeta de red desmontada.
  - a. Crea una copia offline de seguridad en Linux en la carpeta “*Llamadas\_Offline*”.
  
- 2) Resultado con el webservice deshabilitado y carpeta de red habilitada.
  - a. Realiza la copia online de la llamada. Además mueve los ficheros de las llamadas offline en Linux a la carpeta compartida en red, donde se encuentra la llamada actual. El servidor tendría mínimo una llamada esperando al webservice. En Linux no hay llamadas offline.
  
- 3) Resultado con el webservice habilitado y carpeta de red montada.
  - a. Deposita la llamada actual y, si hubiera, las llamadas del backup offline en la carpeta de red. El webservice obtiene la información de cada llamada. Procesa los datos para obtener toda la información del cliente accediendo a BBDD, si está registrado. En ambos casos, toma la llamada y la mueve a la carpeta “**Backup**”, almacenando de forma organizada por día, mes y año las llamadas procesadas. Finalmente, actualiza el fichero “*XML\_RSS.xml*”. Si la

jornada ha finalizado (después de las 00:00), crea una copia de seguridad del fichero RSS en el Backup. En Linux no hay llamadas offline.

- 4) Resultado con el webservice habilitado y carpeta de red desmontada.
  - a. Desde Linux el script no detecta la carpeta montada al no encontrar el fichero "**Baliza\_linux.lnx**". Por lo tanto, realiza el backup offline dentro de la carpeta "**Llamadas\_Offline**". A pesar de llamar al webservice, este no encontrará llamada alguna y no ocurrirá nada más.

### Prueba de ejecución del webservice.

- 1) No hay llamadas en la carpeta de red y webservice activo.
  - a. No ocurre nada.
- 2) Hay llamadas en la carpeta de red y webservice detenido.
  - a. Quedarán a la espera del webservice.
- 3) Hay llamadas en la carpeta y el webservice está activo.
  - a. Ejecuta la rutina de procesamiento para todas las llamadas. Se crea una nueva entrada en el fichero RSS por cada llamada en la carpeta compartida.

### Prueba de integración

Con esta prueba se quiere demostrar que todos los sistemas funcionan y están comunicados correctamente. Se realizarán las siguientes acciones:

1. Un cliente accede al portal web y solicita darse de alta. Rellena el formulario de alta y lo envía. Al cliente le llega un email que confirma su registro en SRIC.
2. El cliente accede al portal web con su usuario y contraseña. Acto seguido registra una incidencia.
3. Un empleado de Kernel Informática S.L inicia sesión con su usuario. Independiente del rol que tenga.
4. El cliente decide llamar a Kernel Informática S.L.
5. Durante la llamada, el teléfono de la oficina de Kernel Informática S.L sonará y el sistema de registro de incidencia de clientes (SRIC) mostrará la información en la parte superior derecha. Indicando el nombre del cliente, descripción del estado de sus incidencias y una breve descripción.

Si todas las tareas las realiza correctamente, se puede afirmar que la integración entre el sistema Linux con la centralita software Asterisk, se comunica correctamente con el webservice. El webservice trabajará correctamente las llamadas, si ha actualizado rápidamente el fichero RSS. Finalmente, ha de confirmarse que SRIC queda informado de la llamada cada poco tiempo mediante el RSS y la tabla de base de datos que contiene el registro de llamadas.

Al realizar la prueba en el servidor con un entorno real de operaciones, se confirma el éxito de la integración. Se ha realizado todas las pruebas y el sistema Linux con Asterisk, el webservice y SRIC han cumplido con los objetivos de la prueba.

Adicionalmente, se confirma el buen rendimiento. Desde que se ha realizado la llamada telefónica de prueba hasta que ha aparecido en el visor de llamadas de SRIC, ha transcurrido menos de un segundo, con toda la información del cliente. Confirmando que funciona.

## Resolución de problemas

Durante las pruebas han surgido problemas inesperados. Se destacan los siguientes inconvenientes por su nivel de dificultad:

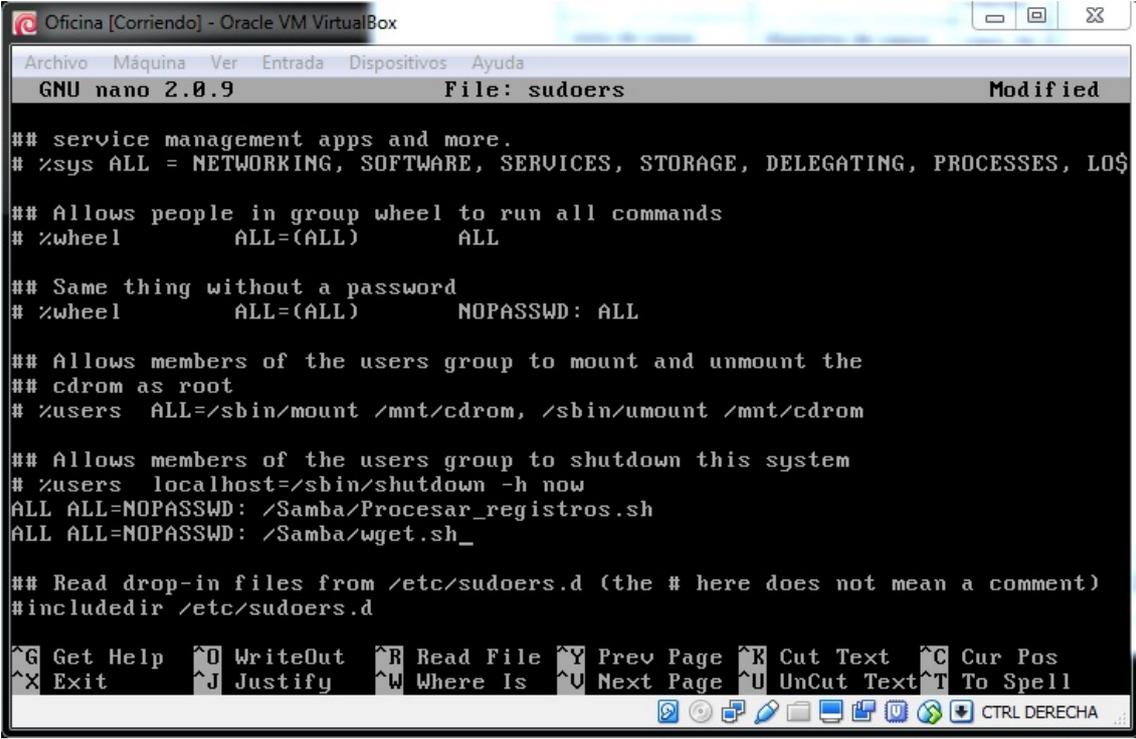
### Los scripts no se ejecutan

El problema surgía inesperadamente al realizar una llamada de prueba entre dos terminales, con el fin de verificar que las órdenes indicadas por los scripts se realizaban correctamente.

Los scripts no se ejecutaban. Sin embargo, desde la consola de Linux funcionaba. Después de indagar en profundidad se detecta que el problema está al ejecutar los scripts con privilegio de administrador, “sudo”. Cada vez que el ejecutable de Asterisk, lanzaba el script, el sistema le pedía la contraseña del administrador.

#### Solución:

Dentro del sistema Linux, se permite indicar en el fichero de configuración “/etc/sudoers”, que scripts del sistema no necesitan escribir la contraseña del administrador al ser ejecutados mediante el comando “sudo”.



```
Oficina [Corriendo] - Oracle VM VirtualBox
GNU nano 2.0.9 File: sudoers Modified
## service management apps and more.
# %sys ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOGS
## Allows people in group wheel to run all commands
# %wheel ALL=(ALL) ALL
## Same thing without a password
# %wheel ALL=(ALL) NOPASSWD: ALL
## Allows members of the users group to mount and unmount the
## cdrom as root
# %users ALL=/sbin/mount /mnt/cdrom, /sbin/umount /mnt/cdrom
## Allows members of the users group to shutdown this system
# %users localhost=/sbin/shutdown -h now
ALL ALL=NOPASSWD: /Samba/Procesar_registros.sh
ALL ALL=NOPASSWD: /Samba/wget.sh_
## Read drop-in files from /etc/sudoers.d (the # here does not mean a comment)
#includedir /etc/sudoers.d
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^U Next Page ^U UnCut Text ^T To Spell
CTRL DERECHA
```

Ilustración 69: Edición de permisos para ejecutar los scripts

Una vez hecho el cambio, los scripts se ejecutan correctamente desde el dialplan.

## Cuelgue inesperado de las llamadas

Durante la programación de la centralita Asterisk se realizaron varias llamadas con diferentes configuraciones.

Al haber creado varios scripts para controlar tareas concretas, se decidió unificarlos todos en un solo script "Procesar\_registro.sh". El cual contiene un algoritmo de control para asegurar la conexión de red, copiar las llamadas (online-offline), etc.

A partir de este script, surgen aleatoriamente problemas en las llamadas de terminar a terminal. Asterisk envía el *error 500*.

Después de analizar y depurar el único script que se ejecutaría en el Dialplan de Asterisk, no se detecta que ocurre. Hasta que se analiza todos los mensajes del depurador que genera el servidor Asterisk internamente. Durante la entrada de una llamada se lanza una excepción "TIMEOUT" que ejecuta la orden "HANGUP" (Colgar), en la referencia ([TOUT]) se indica la documentación que identifica la excepción.

El problema proviene de Asterisk, el script es correcto. Asterisk espera un tiempo por la ejecución del script "Procesar\_registro.sh", debido a que tiene que atender la llamada entrante, se considera que es un camino crítico y prioritario. Es por ello que sucede el timeout.

### **Solución:**

El script se ejecuta en Linux. La mejor manera de evitar que espere es lanzar un hilo independiente para que ejecute el script. Esto se consigue añadiendo al comando "sh" el símbolo "&" al final de cada comando que lanza "Procesar\_registro.sh". Quedando de la siguiente manera:

➤ `sudo sh wget.sh &`

Posteriormente se realizaron varias llamadas de prueba y en ningún caso surgió el error 500. Los terminales sonaron y se podían descolgar y hablar bidireccionalmente.

## El script StartWin.sh no monta la carpeta compartida con el servidor WebDev

Al ejecutar el script, la carpeta que comparte el servidor WebDev no es montada.

### Solución:

Existen dos razones principales.

La carpeta de red indicada en el script se ha modificado. Para conocer si se accede correctamente a la ruta, se accede directamente por el navegador. Si no muestra la ventana que solicita los credenciales (usuario y contraseña) para acceder a la carpeta compartida, significará que la ruta no es correcta.

La segunda razón es que los permisos de la carpeta compartida no son correctos:

- Hay que comprobar que el usuario y contraseña son correctos
- Dentro de los permisos de compartición de la carpeta hay que asegurar que el "usuario del sistema" y "todos" tienen acceso de lectura y escritura. En caso contrario podría no aparecer la carpeta, o bien, nunca poder acceder leer o escribir en ella.

## El script Wget.sh no invoca al webservice

Cuando se ejecuta este script, muestra en la consola de Linux un error de conexión.

**Solución:** El problema está en la ruta del webservice. Hay que indicar la dirección IP del servidor WebDev. En este caso hay que averiguar la dirección IP del servidor WebDev cada vez que se reinicie dicho servidor, en caso de no tener una IP fija.

### Solución:

Para evitar tener que modificar el script cada vez que una máquina reinicie o cambie de dirección de red, se encontró una solución, configurando el fichero `/etc/nsswitch.conf`.

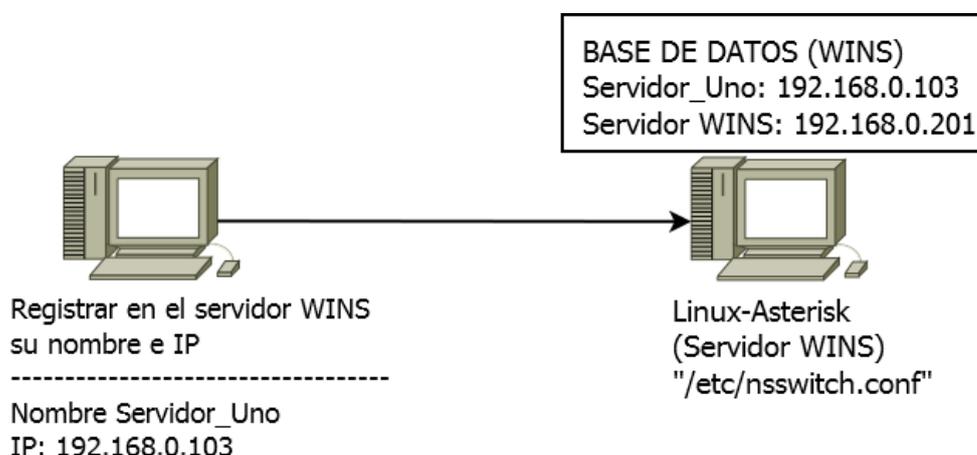


Ilustración 70: Servidor WINS

Este fichero de configuración define el orden de búsqueda en la resolución de los nombres de red. Es necesario introducir el parámetro “wins”, que es un servidor de nombres de Microsoft para NetBIOS que mantiene una tabla de correspondencia entre el nombre de las máquinas y su dirección IP.

En definitiva, no hace falta conocer la dirección IP del servidor WebDev, únicamente el nombre de la máquina.

Por ejemplo, si hay que llamar desde Linux al webservice que está en el Servidor\_Uno, antes había que escribir lo siguiente:

```
“wget \\192.168.0.103\Webservice”
```

Ahora se realiza de la siguiente manera:

```
“wget \\Servidor\_Uno\webservice”
```

### Error de acceso a la carpeta de red

Inicialmente se configuró el sistema para que el webservice accediera a leer las llamadas que se encontraban en una carpeta de red (realizado con un servidor Samba) en Linux. Asterisk depositaba las llamadas en esta carpeta.

El webservice nunca podía acceder a la carpeta de red para leer los ficheros. La ruta venía dada por [\\IP\\_servidor\Carpeta\\_Samba](http://IP_servidor\Carpeta_Samba). Sin embargo la carpeta existía y se podía acceder a ella remotamente sin ningún problema de permisos o credenciales.

#### **Solución:**

Se detectó el problema, provenía del servidor Apache (WAMP). El fichero de configuración “**httpd.conf**” define las rutas visibles, donde se puede acceder y sus atributos.

Por definición, en apache no se puede indicar como carpeta accesible a una dirección UNC (define una sintaxis común para especificar la localización de un recurso de red), por ejemplo, [\\dirección\\_IP](http://dirección_IP) es una ruta UNC, carece letra de unidad.

Por esta razón no era accesible y esta fue la razón por el cual se descartó la solución median Samba desde Linux y se apostó por una carpeta compartida en el servidor WebDev con Windows 7. Desde Linux (dentro de virtualbox) a través del script “startWin.sh”, monta la carpeta de red en una carpeta local dentro del propio sistema Linux, siendo muy práctico y sencillo de utilizar a la hora de volcar los ficheros de las llamadas generados desde Asterisk, que se ejecuta en el mismo sistema Linux.

Esta decisión sirvió de inspiración para invocar posteriormente el webservice desde el sistema Linux y sólo cuando hay una llamada entrante. Evitando de este modo la llamada activa cada cierto intervalo de tiempo que se había planificado en un principio.

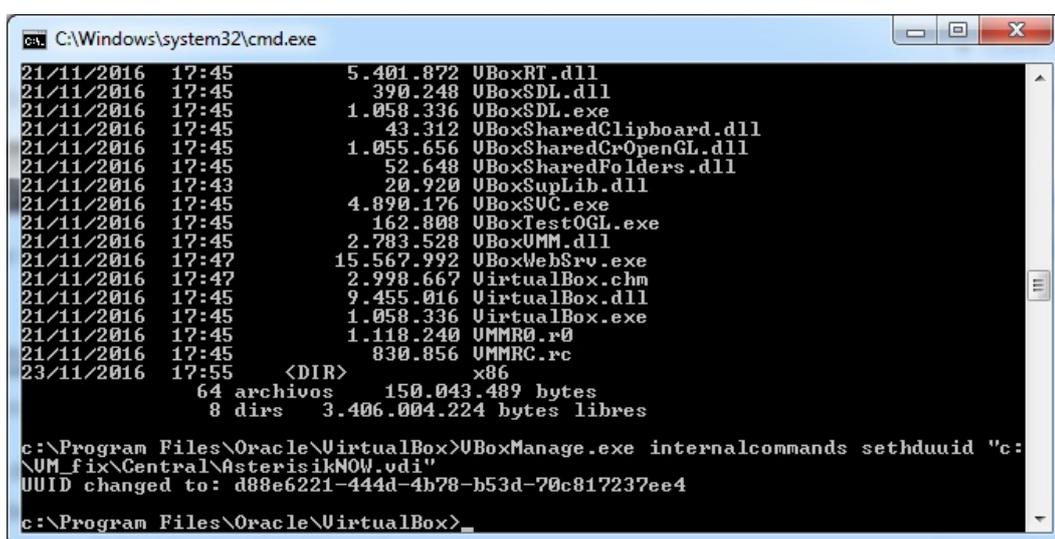
## Error UUID al hacer una copia de una máquina Virtualbox

Durante las pruebas de implantación se ha copiado el fichero virtualbox que representa al disco duro que contiene el sistema Linux CentOS, Asterisk y FreePBX.

El problema es que el software Virtualbox detecta que es una copia por medio de su UUID y no permite utilizarla en una nueva máquina virtual. Muestra un mensaje de error indicando que el identificador del disco duro ya pertenece a otra máquina.

### Solución:

Es necesario acceder a la consola de Windows y ejecutar el comando que aparece en la siguiente ilustración.



```
ca. C:\Windows\system32\cmd.exe
21/11/2016 17:45      5.401.872 UBoxRT.dll
21/11/2016 17:45      390.248 UBoxSDL.dll
21/11/2016 17:45      1.058.336 UBoxSDL.exe
21/11/2016 17:45         43.312 UBoxSharedClipboard.dll
21/11/2016 17:45      1.055.656 UBoxSharedCrOpenGL.dll
21/11/2016 17:45         52.648 UBoxSharedFolders.dll
21/11/2016 17:43         20.920 UBoxSupLib.dll
21/11/2016 17:45      4.890.176 UBoxSUC.exe
21/11/2016 17:45         162.808 UBoxTestOGL.exe
21/11/2016 17:45      2.783.528 UBoxUMM.dll
21/11/2016 17:47     15.567.992 UBoxWebSrv.exe
21/11/2016 17:47      2.998.667 VirtualBox.chm
21/11/2016 17:45      9.455.016 VirtualBox.dll
21/11/2016 17:45      1.058.336 VirtualBox.exe
21/11/2016 17:45      1.118.240 UMMR0.r0
21/11/2016 17:45      830.856 UMMRC.rc
23/11/2016 17:55      <DIR>          x86
                64 archivos    150.043.489 bytes
                8 dirs      3.406.004.224 bytes libres

c:\Program Files\Oracle\VirtualBox>UBoxManage.exe internalcommands sethduid "c:\
\UM_fix\Central\AsteriskNOW.vdi"
UUID changed to: d88e6221-444d-4b78-b53d-70c817237ee4

c:\Program Files\Oracle\VirtualBox>
```

Ilustración 71: Cambio de UUID del disco duro virtual.

Acto seguido al crear la nueva máquina virtual no hay mensaje de error.

## Análisis de costos

A continuación se confecciona el desglose de costes de este proyecto para la venta directa al cliente.

1. **Autor:**
  - Francisco Javier Montero Vega
2. **Departamento:**
  - Ingeniería del software
3. **Descripción**
  - Título: Análisis y Diseño de un sistema software para el registro y seguimiento de las incidencias de los clientes
4. **Presupuesto total del proyecto**
  - 38620 euros.
5. **Costes laborales**

Duración del proyecto: 12 meses

Coste por hora trabajada: 23 euros/hora

Horas de trabajo del proyecto: 950 horas

Coste total= 950horas x 23 euros/hora = 21850 euros

### 6. Costes materiales

Costes materiales necesarios para la ejecución del proyecto:

Precio del Portatil HP pavilion g6 + mantenimiento = 2000 euros

Teléfono Samsung S4 GT-I9505 = 300 euros

Disco duro externo Seagate 1tb = 85 euros

Licencia de Windows = Incluida en portátil

Vida útil del portatil = 4 años = 48 meses

Vida útil del teléfono = 2 años = 24 meses

Vida útil del disco duro = 1 años = 12 meses

Cálculos de amortización:

Coste del portátil:

$2.000 \text{ euros} * 0,25 \text{ uso} * 12 \text{ meses proyecto} / 48 \text{ meses} = 83,33 \text{ euros}$

Coste del teléfono:

$300 \text{ euros} * 0,10 \text{ uso} * 12 \text{ meses proyecto} / 24 \text{ meses} = 15 \text{ euros}$

Coste del disco duro:

$85 \text{ euros} * 0,50 \text{ uso} * 12 \text{ meses proyecto} / 12 \text{ meses} = 42,50 \text{ euros}$

Coste total = 83,33euros + 15euros + 42,50euros = 140,83 euros

## **7. Gastos de viajes y dietas**

Gastos de transporte:

Número de visitas acordadas: 7

Distancia al cliente: 60 km

Precio medio de combustible: 0,93euros/litro

Consumo: 7 litros cada 100km

7 visitas \* 60km = 420km totales

Coste: 420km \* 7 litros /100km \* 0,93 euros/litro = 27,34 euros

Gastos de dietas:

Presupuesto: 50 euros

Coste total = 27,34euros + 50euros = 77,34 euros

## **8. Material fungible**

Coste medio de papeles: 3 euros/mes

Coste medio de tinta de impresora: 15 euros/mes

Coste de papeles = 3 euros/mes \* 12 meses = 36 euros

Costes de tinta de impresora= 10 euros/mes\*12meses = 120 euros

Coste total = 156 euros

## **9. Gastos de documentación**

Coste por fotocopia: 0,035 euros

Número de páginas aproximado: 160

Encuadernación: 6 euros

Número de ejemplares: 5

Coste por ejemplar = 160 página \* 0,035 euro/página \* 6 euros= 33,60 euros

Coste total = 33,60 euros \* 5 ejemplares = 168 euros

## **10. Costes indirectos**

Tasa de costes indirectos: 10%

## **11. Beneficio empresarial**

Porcentaje de beneficio de la empresa: 50% sobre los costes del proyecto

Tabla de Costes totales:

Presupuesto Coste Totales

Costes laborables	21850 euros
Costes materiales	140,83 euros
Gastos de viajes y dietas	73,44 euros
Material fungible	156 euros
Gastos de documentación	168 euros
Costes indirectos	15%
Beneficio empresarial	50%
<b>Total:</b>	<b>38620 euros</b>

## Conclusiones y trabajos futuros

### Conclusiones

El producto obtenido cumple con todos los objetivos propuesto en la definición del proyecto. Su administración y gestión de incidencias también cumplen con las funcionalidades mínimas necesarias que se requieren en este tipo de aplicativos. Añadiendo características interesantes, como por ejemplo, el sistema de conocimiento con búsquedas avanzadas, el cálculo automático de fechas estimadas en función de reglas temporales y dentro del calendario laboral personalizado, las notificaciones visuales ante caducidad de fechas desde el sistema de conocimiento, o bien, la capacidad de informar en tiempo real de una llamada entrante, con más información que el simple nombre del cliente.

El trabajo realizado ha abordado un problema real de la empresa Kernel Informática S.L y se ha afrontado desde cero, con muchas incógnitas por delante que se han ido disipando a medida que se obtenían los requerimientos mediante técnicas de observación, comunicación con los empleados de Kernel informática S.L, documentación y entrevistas. Todo en base a las técnicas vistas en ingeniería del software para la obtención de requerimientos.

Durante cuatro años trabajé en Kernel Informática S.L. Concretamente en atención técnica al cliente de la sección de robótica farmacéutica y he vivido de primera mano la captura, seguimiento y resolución de las incidencias. Entiendo la gran importancia de cumplir con los tiempos de respuesta de solución y resolución, la necesidad de comunicarse con el resto de compañeros, redactar soluciones que ayuden en incidencias futuras y organizar el conocimiento. Toda esta experiencia personal me ha servido de inspiración a la hora de afrontar cualquier duda durante el desarrollo de este proyecto.

El proyecto ha seguido la planificación inicial para el desarrollo de un sistema de registro y seguimiento de incidencias pero vista la complejidad de todo el estudio e implantación del servidor Asterisk y del webservice, han hecho que el proyecto haya alcanzado una magnitud mucho mayor.

Destaco que no ha sido sencillo abordar un problema que inicialmente no aparentaba complejidad pero que a medida que se intentaba avanzar se iba complicando. En este sentido hago especial hincapié al problema de la centralita telefónica virtual Movistar, donde ha sido necesario un gran esfuerzo de investigación y búsqueda, ya que a pesar de que abunda la información de centralitas telefónicas virtuales, apenas hay información para capturar el número de teléfono de una llamada entrante mientras suena. Fue el hecho de probar varios caminos diferentes el que produjo el logro de configurar la centralita telefónica software Asterisk.

El aprendizaje de Asterisk fue muy duro ya que había que personalizar sus ficheros de configuración. Por lo tanto había que aprender cómo se programaba. Vuelvo a recalcar que hay mucha documentación pero muy poco que explique cómo bifurcar llamadas o ideas de cómo obtener el número de teléfono solamente. En definitiva, abreviando los problemas más destacados fueron:

- Tras editar los ficheros de configuración el sistema Asterisk los sobrescribía. La razón es que utiliza un framework, el cual sobrescribe los ficheros de configuración. La solución era bifurcar el flujo de órdenes de una llamada entrante en un punto concreto de su configuración y de esta manera poder insertar código desde otro fichero personalizado.
- Para extraer el número de teléfono durante el flujo de una llamada entrante, se recurre a usar un script que hace uso de comandos Linux. Se genera un fichero con el número de teléfono obtenido de las variables de entorno global de la programación de Asterisk.
- Otro problema fue que el webservice tenía que conocer siempre la ip de la máquina Linux, sino no podía conectar mediante Samba para recibir la llamada capturada. Finalmente, configuré Linux para obtener la ip de una máquina, por su nombre de red mediante un servidor de nombres WIND.
- ¿Cuándo se ejecuta el método que procesa la llamada en el webservice? ¿Cómo sabe si tiene una llamada por procesar? Se estudió un sistema de hilos para llamar al webservice cada poco tiempo, pero nunca llegó a funcionar porque el lenguaje Wlanguage de Webdev no tiene hilos persistentes, solo en la versión para móviles. Además, un webservice no se puede llamar a sí mismo a no ser recursivamente y sin condición de parada, es inviable, por lo tanto tenía que invocarse externamente. La solución fue añadir otro script en Linux para invocar al webservice cada vez que entraba una llamada en el código personalizado del Dialplan de Asterisk. De este modo es muy eficiente al no haber espera activa.
- Hubo que configurar los permisos y credenciales de una carpeta de red en Windows. Un script en Linux se encarga de montar dicha carpeta y depositar ahí las llamadas. El problema es que desde el script no hay manera de saber si está o no montada la carpeta de red por lo que hubo que buscar una manera. Se añadió un fichero "baliza" llamado "Baliza\_linux.lnx" en la carpeta compartida de red en Windows. Cuando el script monte la carpeta de red preguntará si existe el fichero, en caso afirmativo es que ha montado la carpeta de red Windows en Linux con éxito.
- Los scripts desarrollados en Linux habían pasado las pruebas, pero al ejecutarlos en el Dialplan de Asterisk ocurría un error inesperado y colgaba las llamadas. Hubo que hacer una traza completa usando el debugger propio de Asterisk, se detectó un timeout en el Dialplan. Los scripts son pesados, tardan en ejecutarse y frena la ejecución de órdenes críticas de Asterisk. Por lo que cuelga las llamadas. La solución fue lanzar los scripts en segundo plano con "&". Inicialmente no funcionó, porque el usuario Linux "Asterisk" es quien los

ejecuta mediante la orden "sudo". Debido a que se lanza el comando "sudo" el sistema Linux obliga a que inserte la clave del root. Solución, acceder a la configuración de root para que el sistema no le solicite la clave root al usuario de Linux "Asterisk".

A partir de este punto, es posible capturar y transferir las llamadas al webservice e invocarlo posteriormente.

Se ha tenido en cuenta pequeños detalles para garantizar que todas las llamadas de los clientes han sido registradas. Tanto desde Linux como en el webservice, se ha hecho un esfuerzo adicional para no perder ni una sola llamada, implementando soluciones offline en ambos. Permitiendo continuar con la captura de llamadas entrantes en modo offline y a través de los scripts lanzados desde Asterisk restaurar todas las llamadas offline, desde que detecte que existe comunicación con el webservice. Ninguna llamada se pierde. Además el webservice genera el registro de llamadas y crear una copia de seguridad automática de las llamadas por cada jornada de trabajo.

El webservice mientras toma la información de los clientes, cuando estos llaman, además alimenta su propia agenda independiente del sistema de incidencias, para que en el caso de trasladar el servidor Asterisk a otra máquina, el webservice pueda trabajar desde su base de datos local alimentando el RSS en cada llamada.

El entorno de desarrollo de PCSoft (Webdev) facilitó las herramientas de depuración y el acceso automático a la base de datos tanto del sistema de incidencias como del webservice.

Es la primera vez que implemento un webservice y que configuro un servidor Asterisk. Pero sin embargo tengo que decir que es una gran satisfacción completar la integración entre Asterisk desde Linux, el webservice y el sistema de seguimiento y registro de incidencias. Con el fin de obtener la notificación de la llamada de un cliente en tiempo real.

Destaco que es la primera vez que implemento un RSS. En este caso es versión 2.0. El sistema que lee cada 2 segundos el RSS por red me ha gustado porque no ofrece pérdida de rendimiento y ejecuta su objetivo perfectamente. Informar de la llamada entrante en tiempo real.

He de señalar un dato importante y es que hasta la fecha Movistar no ofrece entre sus servicios de centralita telefónica en la nube para empresas, la opción de una agenda corporativa. Que significa que solo hay una agenda para toda la empresa y sus terminales. En este proyecto se ha conseguido implementar una agenda corporativa mediante la comunicación del webservice con la base de datos compartida con el sistema de incidencias. Adicionalmente no se limita sólo a mapear por su número telefónico y extraer el nombre del cliente, ya que además adjunta información actualizada del estado de sus incidencias y avisos importantes respecto al cliente. Por lo tanto, todo empleado podrá no solo detectar quien llama sino estar informado del estado del cliente antes de descolgar.

## Trabajos futuros

Partiendo de todo lo hecho, se puede llevar a cabo una serie de mejoras de ampliación de su funcionalidad, para que sea más completo y flexible, con el fin de cubrir un amplio rango de necesidades. Las mejoras se irían repartiendo por distintas fases de implementación para conseguir un producto mucho más completo y que permita una mayor distribución.

Se contemplan diversos puntos a mejorar en el sistema, de los cuales los más destacados son los siguientes:

1. Actualmente se importa el cliente, su dirección y su contacto desde un fichero del sistema Visual Fox Pro. Como trabajo futuro, seguir importando los demás datos relevantes, tales como las incidencias y partes de cada cliente.
2. Implementar un sistema de seguridad para nuevos usuarios que deseen darse de alta en el sistema:
  - Que el sistema les envíe un email con un enlace, el cual tendrá que acceder para finalizar el registro. Si no fuera así, el registro no se hará efectivo. Actualmente se controla con un número máximo de usuarios no validados por el administrador.
  - Insertar un código captcha en el formulario de registro.
3. Generación de informes internos por cada incidencia. El objetivo es plasmar en un documento su seguimiento en detalle. Incluirá todos sus comentarios asociados, los estados transitados desde su creación hasta su cierre y los tiempos entre cada cambio de estado.
4. Aumentar el número de informes estadísticos.
5. Debido a problemas ya sufridos con cierta frecuencia, la empresa se queda sin comunicación ante fallos de la centralita Movistar. Se plantea, ya que se ha implantado y configurado dentro de la red interna de la oficina, una centralita virtual Asterisk, facilita enormemente tener contratado una línea SIP-Trunk (con número de teléfono asociado). El objetivo es tener su propia centralita telefónica software, interna a la oficina, tal como lo hacían en los inicios de Kernel Informática S.L. Las posibilidades son:
  - Número alternativo de empresa en caso de problema en la centralita principal Movistar.
  - El coste de los proveedores actuales ronda los 16 euros/mes incluido el número de teléfono para recibir llamadas desde fijos, móviles o softphone externos a la empresa.

6. El sistema formado por la centralita Asterisk y el webservice puede ser mejorado haciendo uso de una de las características del sistema de base de datos HFSQL de WebDev, las tablas Cliente/Servidor. Únicamente hay que sustituir las tablas actuales HFSQL clásico, por tablas Cliente/Servidor HFSQL. Este cambio permitirá el acceso remoto a la base de datos. Por lo tanto, la información de las tablas serán accesibles desde internet. Sólo necesitaría la dirección IP y el credencial de conexión.
7. La arquitectura empleada, basada en capas, permite que se pueda implementar sobre otras plataformas, como por ejemplo plataformas móviles ya que toda la lógica de negocio la provee los servicios web, conjuntamente con los patrones de diseño, que permitirán la realización de este tipo de aplicativos. Es por ello que proponemos su desarrollo en una posterior versión del producto.

## Referencias

- [Asterisk]. (s.f.). *Documentación de Asterisk*. Obtenido de <https://wiki.asterisk.org/wiki/display/AST/Home> ; <https://www.voip-info.org/wiki-Asterisk> ;<https://www.voip-info.org/wiki/view/Asterisk+call+notification>
- [Booch06]. (2006). El Lenguaje Unificado de Modelado. En J. Rumbaugh, I. Jacobson, & G. Booch.
- [Dialplan]. (2013). *Explicación del Dialplan de Asterisk*. Obtenido de <http://www.axelko.com/techblog/2013/11/curso-asterisk-iv-el-dialplan/>
- [Ed97]. (1997). Análisis Estructurado Moderno. En E. Yourdon.
- [ITILv3]. (s.f.). *ITIL® v3 Information Technology Infrastructure Library Foundation v3* . Obtenido de <http://infoitilv3.weebly.com/incidentes.html>
- [OOWS]. (s.f.). OOWS: (Artículo) Un Método de Producción de Software en Ambientes Web. En J. Fons, O. Pastor, P. Valderas, & M. Ruiz. Departamento de Sistemas Informáticos y Computación Universidad Politécnica de Valencia.
- [PBX]. (s.f.). *Wikipedia - PBX*. Obtenido de <https://es.wikipedia.org/wiki/PBX>
- [Press10]. (2010). Pressman - Un enfoque práctico. En R. S. Pressman.
- [RFC-SIP]. (s.f.). *Protocol SIP: Session Initiation Protocol*. Obtenido de <https://tools.ietf.org/id/draft-rosenberg-sipping-siptrunk-00.txt> ; <https://www.ietf.org/rfc/rfc3261.txt>
- [RSS]. (s.f.). *RSS (Really Simple Syndication)*. Obtenido de <https://es.wikipedia.org/wiki/RSS>
- [Servicio]. (s.f.). *Análisis de expectativas del servicio al cliente*. Obtenido de <http://www.americalearningmedia.com/edicion-031/349-analisis/5934-ique-expectativas-de-servicios-al-cliente-tienen-los-consumidores>
- [Somm05]. (2005). Sommerville - Ingeniería del Software. En I. Sommerville. Pearson - Addison Wesley.
- [TOUT]. (s.f.). *Timeout Dialplan Asterisk*. Obtenido de <https://wiki.asterisk.org/wiki/display/AST/SIP+Retransmissions>
- [UML]. (s.f.). *The Unified Modeling Language*. Obtenido de <http://www.uml-diagrams.org/>
- [VFPv9]. (s.f.). *Controlador OLEDB para Visual Fox Pro 9*. Obtenido de <https://www.microsoft.com/en-us/download/details.aspx?id=14839>
- [WINS]. (s.f.). *Windows Internet Naming Service*. Obtenido de <http://man7.org/linux/man-pages/man5/nsswitch.conf.5.html>
- [WS]. (s.f.). *Servicios Web (Webservices)*. Obtenido de [https://es.wikipedia.org/wiki/Servicio\\_web](https://es.wikipedia.org/wiki/Servicio_web) ; <https://es.wikipedia.org/wiki/Interoperabilidad>