



UNIVERSIDAD DEL AZUAY

FACULTAD DE CIENCIAS DE LA ADMINISTRACION

ESCUELA DE INGENIERIA DE SISTEMAS

TEMA:

Sistema de Helpdesk mediante el uso de entorno web

**Monografía previa a la obtención del Título de Ingeniería de
Sistemas**

Autor:

Pedro Esteban Astudillo León

Director:

Ing. Francisco Salgado

Cuenca – Ecuador

2012

Los conceptos estudiados, analizados, desarrollados e interpretados junto con las conclusiones del presente trabajo son de exclusiva responsabilidad del autor.

El Autor

Cuenca, febrero de 2012

Pedro Esteban Astudillo León

DEDICATORIA

Esta monografía está dedicada principalmente a mis padres, mi hermano, a mis tíos Hernán Astudillo, Genoveva Guzmán, Celfa León y José Méndez quienes siempre estuvieron junto a mí apoyándome incondicionalmente durante los momentos más difíciles, tanto dentro de mi carrera universitaria como en la vida misma; de igual forma, dedico este trabajo de investigación a los docentes de nuestra prestigiosa universidad, quienes aportaron con conocimientos técnicos para el progreso profesional, como también con consejos éticos y humanísticos para llegar a ser ciudadanos de criterio formado y alcanzar el éxito.

Pedro Astudillo

AGRADECIMIENTO

Agradezco a todos quienes hicieron posible la realización de este trabajo de investigación, a nuestros docentes quienes impartieron sus conocimientos en beneficio de nuestra preparación académica, en especial al Ing. Francisco Salgado quien me guió desde el principio de esta monografía; a la Ing. Ana Vásquez Gerente de Software de la Cooperativa JEP, en la cual laboro por el apoyo brindado al momento de seguir el curso de graduación y a su vez permitirme acceder a la información necesaria para la elaboración de esta investigación.

El Autor

INDICE DE CONTENIDOS

DEDICATORIA	iii
AGRADECIMIENTO.....	iv
INDICE DE CONTENIDOS	v
INDICE DE ILUSTRACIONES	viii
INDICE DE TABLAS	ix
RESUMEN	x
ABSTRACT.....	xjError! Marcador no definido.
INTRODUCCION	1
CAPITULO 1: RECOLECCIÓN Y LEVANTAMIENTO DE INFORMACIÓN PARA EL SISTEMA HELPDESK.....	3
1.1 Recolección de la información:.....	3
1.1.1 Procesos y operaciones del sistema de Helpdesk.....	4
1.1.1.1 Manejo de Incidentes	4
1.1.1.2 Manejo de Problemas	9
1.1.1.2.1 Proceso - Control de Problemas	13
1.1.1.2.2 Proceso - Control de Errores	17
1.2 Planteamiento de Mejoras a los Procesos Existentes	20
1.3 Especificación de Requisitos de Software	21
1.3.1 Introducción	21
1.3.1.1 Propósito.....	21
1.3.1.2 Ámbito del sistema.....	22
1.3.1.3 Definiciones.....	23
1.3.1.4 Visión General del Documento.....	23
1.3.2 Descripción General.....	23
1.3.2.1 Funciones del sistema.....	24
1.3.3 Requisitos específicos	24
1.3.3.1 Requisitos funcionales.....	24
1.3.3.2 Definición de Actores	25
1.3.3.2.1 Usuario.....	25
1.3.3.2.2 Soporte.....	25

1.3.4	Casos de Uso.....	25
1.3.4.1	Descripción de los Casos de Uso:.....	27
1.3.5	Requisitos de interfaces externos.....	34
1.3.5.1	Interfaces de Software.....	34
1.3.5.2	Interfaces de Usuario.....	35
CAPITULO 2. DISEÑO.....		36
2.1	Diagramado UML del Sistema de HelpDesk mediante el uso de Web Service.....	36
2.1.1	Introducción a UML.....	36
2.1.2	Diagrama de Clases.....	38
2.2	Diseño de la Base de Datos.....	39
	Características.....	40
	Características distintivas.....	42
	Tipos de compilación del servidor.....	42
	Especificaciones del código fuente.....	43
	Otras funcionalidades de las listas de correo.....	43
	Características adicionales:.....	44
2.2.1	Diseño del modelo Entidad Relación de la Base de Datos.....	45
2.2.2	Normalización del Modelo:.....	47
2.2.3	Definición del diccionario de datos.....	50
2.3	Diseño de la Aplicación Helpdesk mediante el uso de entorno web.....	57
2.3.1	Definición de los niveles de seguridad y privilegios.....	57
2.3.2	Plantilla a ser utilizada en el sitio web.....	58
2.3.2.1	Ventana de autenticación de usuarios.....	61
2.3.2.2	Páginas de Agregar.....	61
2.3.2.3	Páginas de Modificación.....	63
2.3.2.4	Base de Conocimiento.....	64
2.3.2.5	Bitácora.....	65
2.3.2.6	Incidentes Comunes.....	66
2.3.2.7	Solicitar Soporte.....	68
2.3.2.8	Calificar Requerimiento atendido.....	69
CAPITULO 3 CODIFICACION.....		70

3.1 Codificación de la aplicación web.....	70
3.1.1 Introducción a Html	70
3.1.2 Introducción a PHP	71
3.1.3 Introducción a Javascript.....	71
3.2 Creación de Plantillas Mediante Adobe Dreamweaver.....	72
3.2.1 Creación de Divs.....	73
3.2.2 Creación de CSS	75
3.3 Conexión con la base de datos.....	79
3.3.1 Creación de archivo de conexión.....	79
3.2 Verificación de autenticación.....	79
3.2.1 Código de verificación	80
3.2.2 Código de verificación de sesión iniciada	81
3.3 Código para la consulta de incidentes comunes.....	82
3.4 Código empleado para el envío de solicitud de soporte	84
3.5 Verificación de Requerimientos	86
3.6 Grabación de Bitácora	89
CAPITULO 4 INSTALACION DEL SISTEMA Y PRUEBAS	92
4.1 Instalación del servidor WAMPSEVER.....	92
4.2 Instalación del sistema	94
4.3 Pruebas del sistema.....	94
CONCLUSIONES Y RECOMENDACIONES.....	96

INDICE DE ILUSTRACIONES

Figura 1.1 Gestión de Incidente	7
Figura 1.2 Gestión de Problemas	10
Figura 1.3 Gestión de Configuraciones.....	11
Figura 1.4 Diagrama de Casos de Uso	26
Figura 2.1 Descripción de Diagramado UML.....	36
Figura 2.2 Diagrama de Clases	39
Figura 2.3 Diagrama Modelo Entidad Relación.....	46
Figura 2.4 Diagrama Modelo Relacional	49
Figura 2.5 Plantilla del Sistema Helpdesk.....	59
Figura 2.6 Ingreso de Usuarios.....	61
Figura 2.7 Agregar Personas	62
Figura 2.8 Agregar Incidente Común.....	62
Figura 2.9 Agregar Departamento	63
Figura 2.10 Modificación de Personas.....	63
Figura 2.11 Búsqueda Base de Conocimiento	¡Error! Marcador no definido.
Figura 2.12 Base de Conocimiento	64
Figura 2.13 Requerimientos de Soporte.....	65
Figura 2.14 Bitácora.....	66
Figura 2.14 Búsqueda de Incidentes.....	67
Figura 2.15 Selección de Incidente	67
Figura 2.16 Solución a incidente común.....	68
Figura 2.17 Solicitar Soporte	68
Figura 2.18 Calificar Requerimiento	69
Figura 2.19 Calificación	69
Figura 3.1 Creación de plantilla	73
Figura 4.1 Instalador Wampserver	92
Figura 4.2 Asistente de instalación de Wampserver.....	93
Figura 4.3 Aceptación de Licencia Wampserver	94

INDICE DE TABLAS

Tabla 1.1 Definición de Términos en la ERS	23
Tabla 1.2 Descripción del caso de uso 1	27
Tabla 1.3 Descripción del caso de uso 2	27
Tabla 1.4 Descripción del caso de uso 3	28
Tabla 1.5 Descripción del caso de uso 4	28
Tabla 1.6 Descripción del caso de uso 5	29
Tabla 1.7 Descripción del caso de uso 6	29
Tabla 1.8 Descripción del caso de uso 7	30
Tabla 1.9 Descripción del caso de uso 8	31
Tabla 1.10 Descripción del caso de uso 9	31
Tabla 1.11 Descripción del caso de uso 10	32
Tabla 1.12 Descripción del caso de uso 11	32
Tabla 1.13 Descripción del caso de uso 12	33
Tabla 1.14 Descripción del caso de uso 13	34
Tabla 2.1 Tabla tcargo	51
Tabla 2.2 Tabla tdepartamento	51
Tabla 2.3 Tabla tglosarioterminos	51
Tabla 2.4 Tabla tincidentecomun	52
Tabla 2.5 Tabla tnivelsoporte	52
Tabla 2.6 Tabla tpersona	53
Tabla 2.7 Tabla tpersonaincidentecomun.....	53
Tabla 2.8 Tabla tprivilegios	54
Tabla 2.9 Tabla trequerimiento	55
Tabla 2.10 Tabla tsoporte	56
Tabla 2.11 Tabla ttipoincidentecomun	56
Tabla 2.12 Tabla tusuarios.....	56

RESUMEN

La presente monografía tiene como objetivo proporcionar un sistema rápido y sencillo desarrollado en entorno web, para el acceso de empleados que tengan incidentes informáticos en cualquier instancia durante su jornada de trabajo normal y puedan encontrar ayuda inmediata, así como también acceder a solicitar soporte mediante el sistema y con respaldo de correo electrónico para constancia de que se requirió soporte.

Dentro del sistema se registran los requerimientos solicitados por parte de los empleados, así como el acceso a los mismos por parte del personal de soporte, buscando la solución a través de la Base de Conocimiento.

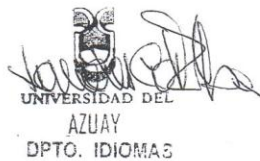
El personal de soporte llenará la bitácora, con la solución a los incidentes suscitados, mismo que será calificado por los empleados implicados de acuerdo a niveles de satisfacción.

ABSTRACT

The goal of the present research project is to provide a fast and simple system developed through a web setting that can be easily accessed by employees who have any type of informatics incident during their regular work day in order to find immediate help and ask for assistance through this system and with the support of an e-mail to keep record.

The system registers the employees' requests as well as the access to the support team, in search of solutions through the knowledgebase.

The support team will fill the blog with the solutions to the problems presented. The blog will be graded by the employees according to the levels of satisfaction.



Translated by,

Diana Lee Rodas

INTRODUCCION

Esta investigación se ha realizado con la finalidad de obtener un Sistema de Helpdesk para una empresa que cuente con incidentes de tipo informáticos, utilizando un entorno Web, para así brindar soporte óptimo al momento de suscitarse algún incidente

La primera etapa durante el desarrollo de esta investigación fue la recolección y levantamiento de información con la ayuda de una persona de sistemas quien me indicó los incidentes de tipo informático que se suscitan en el día a día de una empresa; además se utilizó la herramienta de internet y bibliografía para entender el proceso que debe seguir un sistema de Helpdesk.

Ya con la información se pudo determinar los requisitos principales que debe desempeñar el sistema.

En el sistema se definirán los permisos de acceso para los diferentes usuarios del mismo.

El sistema tenía que contar con una aplicación que sea capaz de brindar ayuda inmediata a usuarios al momento de suscitarse un incidente informático, pudiendo acceder a posibles soluciones rápidas, como solicitar ayuda mediante el sistema como tal.

A su vez al momento de solicitar soporte, el personal encargado de brindar el mismo, dentro del sistema estará verificando y atendiendo los incidentes que le

fueron encomendados y para una mayor seguridad también les llegará un correo electrónico con el requerimiento solicitado.

Llenarán una bitácora con el soporte brindado y a su vez los usuarios que recibieron ayuda podrán calificar a quien les atendió.

Una vez establecidos los objetivos a alcanzar, se procedió con la fase de análisis y diseño, en donde se definen los procesos a seguir, así como la base de datos y aplicaciones; para luego realizar la codificación respectiva, especificando las herramientas utilizadas.

Una vez finalizada la codificación, se procedió con la implementación y pruebas del mismo, con las cuales el software elaborado estará listo para ser manipulado por los usuarios finales, quienes probarán características y el funcionamiento del sistema.

CAPITULO 1: RECOLECCIÓN Y LEVANTAMIENTO DE INFORMACIÓN PARA EL SISTEMA HELPDESK

1.1 Recolección de la información:

Las empresas que manejan un sistema automatizado transaccional para la obtención de resultados e información mediante las operaciones diarias, y que a su vez cuentan con hardware de última tecnología tienden a estar sujetas a incidentes diariamente de tipo software, hardware, redes y otros; los cuales no son resueltos de manera inmediata causando insatisfacción a los usuarios del sistema y equipos de cómputo.

La mayoría cuentan con personal de sistemas que están en capacidad de dar soporte pero no registran la solución dentro de una base de conocimiento y tampoco hacen un escalamiento, si no todo es mecánico.

Cuando se ha resuelto un incidente o problema, no se almacena en una Base del Conocimiento para cuando se dé una reincidencia del mismo, si no sólo el personal que lo resolvió sabe cómo solucionarlo.

Dependiendo del tipo de incidente que sea, primero se está en conversación con el usuario y si se encuentra al nivel del personal de soporte que lo está atendiendo lo ayudará con la solución, de no le pasará la llamada hacia otro personal que tenga conocimiento acerca del problema del usuario; el

escalamiento es sólo de manera verbal y no se registra en algún sistema para que cuando se de reincidencia se pueda establecer comunicación directa con el personal en capacidad de solucionar el mismo.

1.1.1 Procesos y operaciones del sistema de Helpdesk

1. Definir el tipo de incidente a solucionar.
2. Buscar dentro de la Base de Conocimiento la o las posibles soluciones.
3. Dar soporte al usuario.
4. Registrar dentro de la Base del Conocimiento la solución que se dio al incidente suscitado.
5. Sacar reportes de tanto de los diferentes tipos de incidentes, áreas en las que más se registran incidentes, etc.

1.1.1.1 Manejo de Incidentes

El objetivo principal es restablecer el servicio inmediatamente, resolviendo cualquier incidente que cause interrupción en el mismo de la manera más rápida y eficaz posible para evitar que el usuario se vea afectado.

Las propiedades y funcionalidades del Manejo de Incidentes se resumen de la siguiente manera:

- **Incidencia**

Interrupción de los servicios TI (o Petición de servicio)

- Comunicada por usuario
- Generada automáticamente por aplicaciones

- **Service Desk**

Responsable directo de la gestión de las incidencias

- Centro de contacto de la organización TI
- 1ra línea de soporte

- **Registro y Clasificación**

Creación de un registro de incidente

- Prioridad = Impacto * Urgencia
- Categorización: asignación de tipo y personal de soporte

- **KDB (Knowledge Data Base) o Base de Conocimiento**

Análisis y diagnóstico

- Consulta Base de Datos de Conocimiento
- ¿Solución preestablecida?

¿Resuelto?

Si se conoce el método de solución:

- Se asigna los recursos necesarios

Si NO se conoce el método de solución:

- Se escala la incidencia a un nivel superior de soporte

- **Escalado**

Existen dos tipos de escalado en el proceso de resolución de una incidencia:

- Escalado funcional: Se recurre a técnicos de nivel superior
- Escalado Jerárquico: Entran en juego más altos responsables de la organización TI

¿Resuelto?

Si se conoce el método de solución:

- Se asigna los recursos necesarios

Si NO se conoce el método de solución:

- Se escala la incidencia a un nivel superior de soporte

- **Resolución y Cierre**

Cuando se ha resuelto el incidente satisfactoriamente:

- Registro del proceso en el sistema y en la Base de Datos de Conocimiento en donde se documentan todas las soluciones y se establecen los pasos a seguir para que se haga de forma eficiente, para que al momento de volverse a presentar el incidente ya va a estar documentado y esto hace que sea más fácil, rápida y eficiente su resolución.
- Si fuera necesario, generar una RFC (Petición de Cambio).

- **Monitorización y Seguimiento**

Todo el proceso debe ser controlado mediante la:

- Emisión de informes
- Actualización de las bases de datos asociadas
- Monitorización de los niveles de servicio

- **Interrelaciones**

Debe existir una estrecha relación entre Gestión de incidentes y otros procesos TI con el objetivo de:

- Mejorar el servicio y cumplir adecuadamente los SLA's
- Conocer la capacidad y disponibilidad de la infraestructura TI
- Planificar y realizar los cambios necesarios para la optimización y desarrollo del servicio TI

Para este proceso se tiene un diagrama que en cada una de sus fases maneja cuatro pasos:

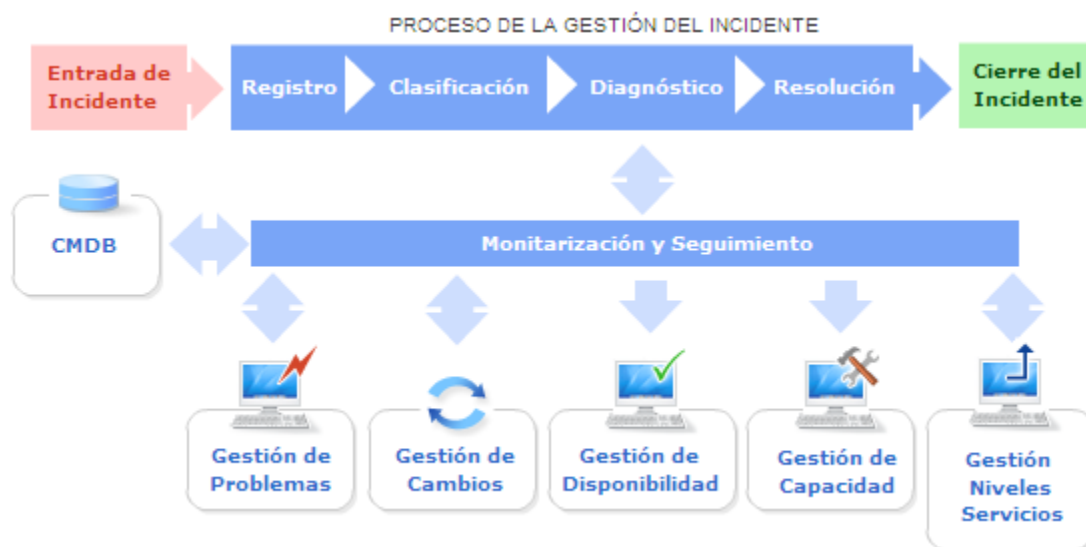


Figura 1.1 **Gestión de Incidente**

Gestión de Configuraciones: La CMDB juega un papel importante en la resolución de incidentes, nos muestra información acerca de los responsables de los componentes de configuración implicados; a su vez nos permite conocer todas las implicaciones que pueden tener en otros servicios el malfuncionamiento de un determinado CI (elemento de configuración).

Al resolver el incidente se deberá actualizar la CMDB en caso de que haya sido necesario cambiar o modificar ciertos elementos de configuración.

Gestión de Problemas: Ofrece ayuda a la Gestión de Incidentes informando sobre errores conocidos y posibles soluciones temporales, a su vez establece controles sobre la calidad de la información registrada por la Gestión de Incidentes para que ésta sea de utilidad en la detección de problemas y su posible solución.

Gestión de Cambios: La resolución de un incidente puede generar una RFC (petición de cambio) que se envía a la Gestión de Cambios. Por otro lado, un determinado cambio erróneamente implementado puede ser el origen de múltiples incidencias y la Gestión de Cambios debe mantener cumplidamente informada a la Gestión de Incidencias sobre posibles incidencias que los cambios realizados puedan causar en el servicio.

Gestión de Disponibilidad: Utilizará la información sobre la duración, el impacto y el desarrollo temporal de los incidentes para elaborar informes sobre la disponibilidad real del sistema.

Gestión de Capacidad: Se ocupará de incidentes causados por una insuficiente infraestructura (insuficiente ancho de banda, capacidad de proceso,...).

Gestión de Niveles de Servicio: La Gestión de Incidentes debe tener acceso a los SLA acordados con el usuario para poder determinar el curso de las acciones a adoptar. Por otro lado, la Gestión de Incidentes debe proporcionar periódicamente informes sobre el cumplimiento de los SLAs acordados.

1.1.1.2 Manejo de Problemas

Las principales funciones del manejo de problemas es la siguiente:

- Investigar las causas subyacentes a toda alteración, real o potencial, del servicio TI.
- Determinar posibles soluciones a las mismas.
- Proponer las peticiones de cambio (RFC) necesarias para restablecer la calidad del servicio.
- Realizar Revisiones Post Implementación (PIR) para asegurar que los cambios han surtido los efectos buscados sin crear problemas de carácter secundario.

La Gestión de Problemas puede ser:

- **Reactivo:** Analiza los incidentes ocurridos para descubrir su causa y propone soluciones a los mismos.

- **Proactivo:** Monitoriza la calidad de la infraestructura TI y analiza su configuración con el objetivo de prevenir incidentes incluso antes de que estos ocurran.

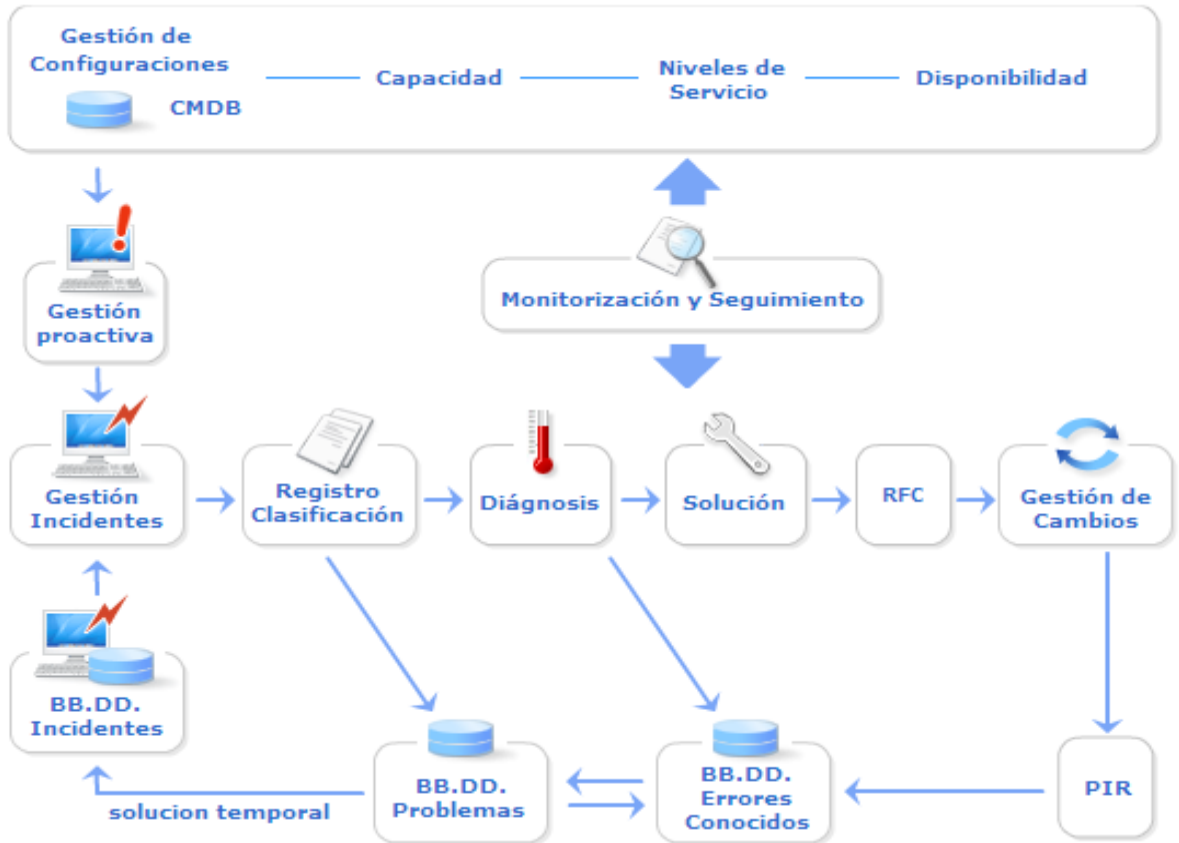


Figura 1.2 **Gestión de Problemas**



Figura 1.3 **Gestión de Configuraciones**

Gestión de Configuraciones: Se debe disponer de una Base de Datos de Configuraciones actualizada para determinar qué elementos de la configuración (CIs) están relacionadas con el problema en curso de investigación.

Es frecuente que el origen del problema provenga de la incompatibilidad entre diferentes elementos de hardware que pueden ser rápidamente solucionados descargando nuevas versiones de los controladores, firmware, etc.

Gestión de Incidentes: Es la fuente principal de la Gestión de Problemas. La Base de Datos de Incidentes ofrece valiosa información sobre los posibles orígenes del problema.

Por otro lado la Gestión de Problemas da soporte a la Gestión de Incidentes ofreciendo soluciones temporales que permitan minimizar el impacto de los incidentes en la calidad del servicio.

Gestión de Disponibilidad: Aporta información de gran importancia a la Gestión de Problemas que ayuda a determinar cuándo y cómo se ha interrumpido o deteriorado el nivel de servicio.

Por otro lado la Gestión de Problemas da soporte a la Gestión de Disponibilidad informando de las causas a la falta de acceso y poniendo en marcha medidas encaminadas a remediarlas.

Gestión de Capacidad: Muchos problemas tienen su origen en una falta de capacidad en la infraestructura TI, es por tanto indispensable que exista una estrecha colaboración entre la Gestión de la Capacidad y la de Problemas.

La Gestión de Capacidad ayuda a determinar las necesidades futuras de la infraestructura TI e impedir que se deterioren los futuros niveles de servicio.

Gestión de Niveles de Servicio: Nos sirve para determinar el impacto real de un problema específico sobre la calidad del servicio.

Una gestión proactiva requiere conocer en profundidad los SLAs acordados con los usuarios y determinar si estos pueden sobrecargar algunos componentes de la infraestructura TI. Por otro lado la Gestión de Problemas ha

de trabajar siempre con el objetivo último de que se cumplan adecuadamente los SLAs.

Gestión de Cambios: Cuando un problema se convierte en un error conocido la Gestión de Problemas debe elevar una RFC a la Gestión de Cambios con el objetivo de que el problema subyacente se resuelva definitivamente.

Tras la implementación del cambio la Gestión de Problemas debe llevar a cabo una revisión post-implementación (PIR) antes de cerrar definitivamente el problema.

1.1.1.2.1 Proceso - Control de Problemas

El principal objetivo del Control de Problemas es conseguir que estos se conviertan en Errores Conocidos para que el Control de Errores pueda proponer las soluciones correspondientes.

El Control de Problemas se compone en esencia de tres fases:

a. Identificación y Registro

Una de las tareas principales de la Gestión de Problemas es identificar los mismos. Las principales fuentes de información utilizadas son:

La base de datos de Incidentes: en principio cualquier incidente del que no se conocen sus causas y que se ha cerrado mediante algún tipo de solución temporal es potencialmente un problema. Sin embargo, se habrá de analizar si este incidente es aislado o su impacto en la estructura TI antes de elevarlo a la categoría de problema.

Análisis de la infraestructura TI: en colaboración con la Gestión de Disponibilidad y de Capacidad, la Gestión de Problemas debe analizar los diferentes procesos y determinar en qué aspectos se debe reforzar los sistemas y estructuras TI para evitar futuros problemas.

Deterioro de los Niveles de Servicio: el descenso del rendimiento puede ser una indicación de la existencia de problemas subyacentes que no se hayan manifestado de forma explícita como incidentes.

Todas las áreas de la infraestructura TI deben colaborar con la Gestión de Problemas para identificar problemas reales y potenciales informando a ésta de cualquier síntoma que pueda ser señal de un deterioro en el servicio TI.

El registro de problemas es, en principio, similar al de los incidentes aunque el énfasis debe hacerse no en los detalles específicos de los incidentes asociados sino más bien en su naturaleza y posible impacto.

El registro debe incorporar, entre otra, información sobre:

- Los CIs implicados.
- Causas del problema.
- Síntomas asociados.
- Soluciones temporales.
- Servicios involucrados.
- Niveles de prioridad, urgencia e impacto.
- Estado: activo, error conocido, cerrado.

b. Clasificación y Asignación de Recursos

La clasificación del problema engloba desde las características generales de éste, tales como si es un problema de hardware o software, que áreas funcionales se ven afectadas y detalles sobre los diferentes elementos de configuración (CIs) involucrados en el mismo.

Un factor esencial es la determinación de la prioridad del problema, que al igual que en el caso de los incidentes, se determina tanto a partir de la urgencia (demora aceptable para la solución del problema) como de su impacto (grado de deterioro de la calidad del servicio).

Al igual que en la Gestión de Incidentes la prioridad puede cambiar en el curso del ciclo de vida del problema, por ejemplo, si se encuentra una solución temporal al mismo que reduce considerablemente su impacto.

Una vez clasificado y determinada su prioridad se deben de asignar los recursos necesarios para su solución. Estos recursos deben ser suficientes

para asegurar que los problemas asociados son tratados eficazmente y así minimizar su impacto en la infraestructura TI.

c. Análisis y Diagnóstico: Error conocido

Los objetivos principales del proceso de análisis son:

- Determinar las causas del problema.
- Proporcionar soluciones temporales a la Gestión de Incidentes para minimizar el impacto del problema hasta que se implemente los cambios necesarios que lo resuelvan definitivamente.

Es esencial tener en cuenta que no siempre el origen del problema es un error de hardware o software. Es moneda frecuente que el problema este causado por:

- Errores de procedimiento.
- Documentación incorrecta.
- Falta de coordinación entre diferentes áreas.

Es también posible que la causa del problema sea un "bug" bien conocido de alguno de las aplicaciones utilizadas. Por lo tanto es conveniente establecer contacto directo con el entorno de desarrollo, en caso de aplicaciones desarrolladas "en la casa", o investigar en Internet información sobre errores conocidos aplicables al problema en cuestión.

Una vez determinadas las causas del problema éste se convierte en un Error Conocido y se remite al Control de Errores para su posterior procesamiento.

1.1.1.2.2 Proceso - Control de Errores

Una vez que el Control de Problemas ha determinado las causas de un problema es responsabilidad del Control de Errores el registro del mismo como error conocido.

Identificación y Registro de errores

El registro de los errores conocidos es de vital importancia para la Gestión de Incidentes pues debe llevar asociado, siempre que esto sea posible, algún tipo de solución temporal que permita minimizar el impacto de los incidentes asociados.

Análisis y Solución

Se deben investigar diferentes soluciones para el error evaluando en cada momento:

El posible impacto de las mismas en la infraestructura TI.

- Los costes asociados.
- Sus consecuencias sobre los SLAs.

En algunos casos, en los que el impacto del problema puede tener consecuencias graves en la calidad del servicio, pueden emitirse una RFC de emergencia para su procesamiento urgente por la Gestión de Cambios.

Una vez determinada la solución óptima al problema y antes de elevar una RFC a la Gestión de Cambios han de tenerse en cuenta las siguientes consideraciones:

- ¿Es conveniente demorar la solución? Ya sea porque se prevén cambios significativos en la infraestructura TI a corto plazo o por el escaso impacto del problema en cuestión.
- ¿Es la solución temporal aportada suficiente para mantener unos niveles de calidad de servicios aceptable?
- ¿Los beneficios justifican los costes asociados?

Sea cual sea la respuesta, toda la información sobre el error y su solución se registrará en las bases de datos asociadas. En el caso en el que se considere que el problema necesita ser solucionado se emitirá una RFC. Será responsabilidad de la Gestión de Cambios la implementación de los cambios de infraestructura propuestos.

Revisión Post Implementación y Cierre

Antes de dar el problema por resuelto y cambiar su estado a “cerrado” se debe analizar el resultado de la implementación de la RFC elevado a la Gestión de Cambios (PIR).

Si los resultados de esta PIR son los deseados y se pueden cerrar todos los incidentes relacionados con este problema se considera concluido el proceso y se emiten los informes correspondientes.

El objetivo de la Gestión de Problemas no es otro que el de mejorar el funcionamiento de la infraestructura TI y para evaluar su eficacia es imprescindible realizar un continuo seguimiento de los procesos relacionados y evaluar su rendimiento.

En particular una buena gestión de problemas debe traducirse en una:

- Disminución del número de incidentes y una más rápida resolución de los mismos.
- Mayor eficacia en la resolución de problemas.
- Gestión proactiva que permita identificar problemas potenciales antes de que estos se manifiesten o provoquen una seria degradación de la calidad del servicio.

La correcta elaboración de informes permite evaluar el rendimiento de la Gestión de Problemas y aporta información de vital importancia a otras áreas de la infraestructura TI.

Entre la documentación generada cabría destacar:

Informes de Rendimiento de la Gestión de Problemas: donde se detalle el número de errores resueltos, la eficacia de las soluciones propuestas, los tiempos de respuesta y el impacto en la Gestión de Incidentes.

Informes de Gestión Proactiva: donde se especifiquen las acciones ejercidas para la prevención de nuevos problemas y los resultados de los análisis realizados sobre la adecuación de las estructuras TI a las necesidades de la empresa.

Informes de Calidad de Productos y Servicios: donde se evalúe el impacto en la calidad del servicio de los productos y servicios contratados y que eventualmente puedan permitir adoptar decisiones informadas sobre cambios de proveedores, etc.

Una eficaz Gestión de Problemas también requiere determinar claramente quienes son los responsables de cada proceso. Sin embargo, en pequeñas organizaciones es recomendable no segmentar en exceso las responsabilidades para evitar los costes asociados: sería poco eficaz y contraproducente asignar unos recursos humanos desproporcionados al proceso de identificación y solución de problemas.

1.2 Planteamiento de Mejoras a los Procesos Existentes

Las mejoras que se pueden implementar dentro de los procesos existentes para el Helpdesk dentro de una empresa automatizada, está en que se acceda al sistema de Helpdesk por medio de un entorno Web por parte del usuario quien

a primera instancia podrá encontrar la solución inmediata a incidentes comunes, y en caso de no hacerlo enviar por medio de correo electrónico la petición a personal capacitado, el cual se comunicará inmediatamente y a través de una Base de Conocimiento irá descartando las posibles soluciones a la incidencia hasta encontrar la adecuada y lograr solucionar el percance del usuario.

En caso de que este personal de soporte no logre encontrar la solución a la incidencia se efectuará el respectivo escalamiento vía correo electrónico hasta llegar al personal capacitado que logre dar solución a la incidencia.

El personal encargado de monitorizar la satisfacción de los usuarios al recibir el soporte respectivo al momento de ocurrirles un incidente o problema, podrá hacerlo por medio del entrono Web en el cual podrá encontrar gráficos estadísticos, por departamento, área, agencia, etc.; a más de poder consultar por rango de fechas, ver el porcentaje de avance en caso de ser un problema.

El personal de soporte en sus diferentes rangos cada vez que se presente un incidente o problema, llenará la Base de Conocimiento, dependiendo de lo que se presente, lo cual será por medio de una bitácora de soporte al iniciar, estar en proceso y finalizar la solución del incidente o problema.

1.3 Especificación de Requisitos de Software

Las Especificaciones de Requisitos de Software (ERS) para el sistema de HelpDesk serán mediante el formato de Especificación de Requisitos de Software según la versión de 1998 del estándar IEEE830.

1.3.1 Introducción

En esta sección se encuentra la Especificación de Requisitos de Software para el Sistema de HelpDesk; todo su contenido será realizado con la colaboración personal administrativo, soporte y usuarios finales.

1.3.1.1 Propósito

El propósito de las especificaciones de requisitos del software es tener una definición, clara y precisa de las funcionalidades y condiciones técnicas del sistema a desarrollar.

Esta sección está dirigida a los usuarios del sistema, tanto el personal de soporte como el personal que solicita ayuda por medio del sistema.

A su vez será el medio por el cual mantendrán una comunicación las partes involucradas para el desarrollo del Sistema de Helpdesk por medio de entrono web.

La especificación estará sujeta a revisiones por los grupos de usuarios, según las diversas versiones generadas por los cambios solicitados.

1.3.1.2 Ámbito del sistema

El sistema estará contemplado de la siguiente manera:

- Gestionar Usuarios.
- Gestionar Incidentes.
- Gestionar Bitácora o requerimientos.

- Gestionar Departamentos.
- Gestionar Cargos.

El sistema deberá gestionar los usuarios de manera que los usuarios de soporte quienes estarán a cargo de la administración del sistema puedan agregar, modificar, eliminar o consultar a los mismos.

A su vez gestionarán tanto los departamentos, cargos para su creación, modificación y consulta; en lo que respecta con la bitácora o requerimientos realizarán la consulta, modificación de la misma que es en donde los usuarios que solicitan ayuda llenan su requerimiento.

En lo que respecta a los incidentes igual cada vez que se dé uno que sea repetitivo será creado como incidente común y de igual forma se creará, modificará o eliminará en caso de que ya no se den causas para el mismo.

1.3.1.3 Definiciones

Gestionar	Dirigir o Administrar una empresa
ERS	Especificación de los requisitos de software
Incidentes	Son pequeños problemas que se suscitan en cualquier momento afectando al usuario de un sistema, hardware o algún programa utilitario.
Bitácora:	Parte del sistema en el cual se registran los incidentes suscitados por parte del personal de la empresa, y a su vez donde guardan la ayuda dada y la solución a los incidentes por parte de soporte.

Tabla 1.1 Definición de Términos en la ERS

1.3.1.4 Visión General del Documento

El contenido del documento está formado de tres secciones. En la que nos encontramos que es la primera es la Introducción que nos da una visión general de la ERS. En la segunda sección se da una descripción general del sistema conociendo sus principales funciones, datos asociados, requisitos técnicos que afectan al desarrollo del mismo. En la tercera sección se define a detalle los requisitos que debe cumplir el sistema.

1.3.2 Descripción General

Esta sección describe de una forma más abstracta el sistema, dentro de la cual están las áreas más importantes del sistema que recibirán soporte del mismo; además las funciones que el sistema debe cumplir, información empleada, condiciones y demás factores que afecten durante el desarrollo del sistema.

1.3.2.1 Funciones del sistema

El sistema proporciona la ayuda necesaria para que los usuarios que tengan un incidente común puedan consultar dentro del mismo y hallar la solución lo más rápido sin necesidad de soporte.

El sistema proporciona un medio por el cual el usuario solicita ayuda al personal de soporte vía correo electrónico describiendo en breves rasgos el incidente suscitado.

El sistema proporciona una Base de Conocimiento mediante la cual el personal de soporte va descartando las posibles soluciones al incidente suscitado.

El sistema proporciona una bitácora en la cual se registrará cada incidente suscitado, con la fecha y hora exacta, así como el usuario que solicitó, personal de soporte que atendió la misma, la solución y si es que se necesitó de un escalamiento.

1.3.3 Requisitos específicos

Esta es la sección más importante de la ERS porque se detallan los requisitos más a fondo. Gracias a esto los analistas pueden realizar un sistema que satisfaga las necesidades y requisitos de los usuarios, así como la realización de pruebas para lograr la satisfacción de los usuarios.

1.3.3.1 Requisitos funcionales

Los requisitos funcionales estarán agrupados por medio de Casos de Uso, que darán una perspectiva a través de los actores, y dará una visión a nivel general del alcance que tendrá el sistema.

1.3.3.2 Definición de Actores

1.3.3.2.1 Usuario

Se refiere a quienes son los actores directos afectados por las incidencias o problemas que tengan de tipo software, hardware, utilitarios, redes, etc.; los mismos que ingresaran vía web al sistema de HelpDesk consultarán la solución a la incidencia común y en caso de no hallarla accederán a la solicitud de asistencia al personal de soporte vía correo electrónico.

1.3.3.2.2 Soporte

Es el personal encargado de comunicarse inmediatamente después de recibir el correo electrónico por parte del usuario para brindar la ayuda necesaria accediendo a la Base de Conocimiento y en caso de no encontrar solución a la incidencia recurrir al escalamiento vía correo electrónico a un personal de Nivel Superior.

1.3.4 Casos de Uso

Los Casos de Uso son una secuencia de interacciones se desarrollan entre un sistema y sus actores en respuesta a un evento que inicia uno de los actores sobre el sistema.

Los diagramas de caso de uso nos sirven para entender los requerimientos que presenta el sistema en base a eventos que se producen dentro del ámbito en el que se desenvuelven los actores y el sistema.

Diagrama de casos de uso Sistema de Helpdesk mediante el uso de entorno web

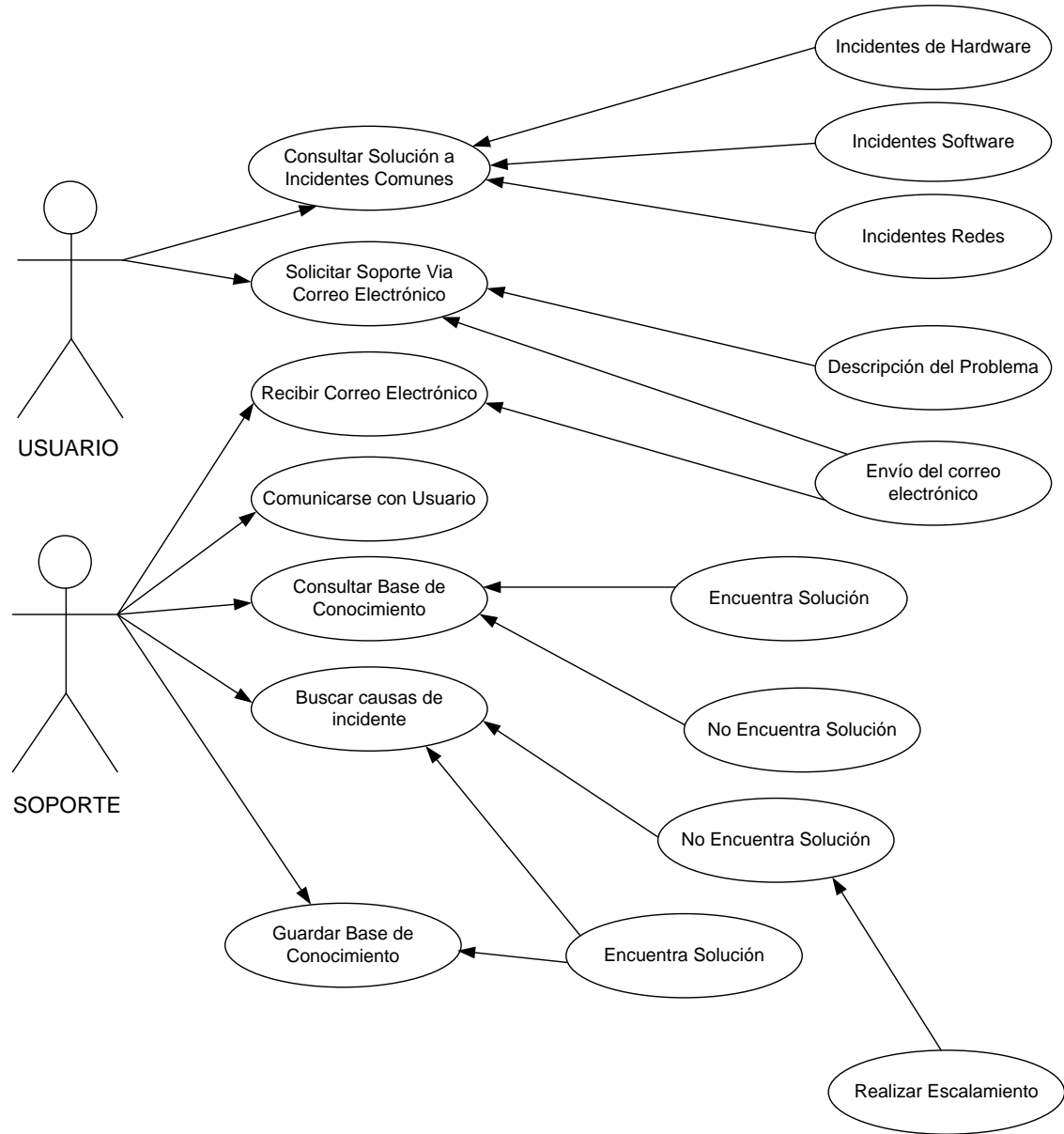


Figura 1.4 Diagrama de Casos de Uso

1.3.4.1 Descripción de los Casos de Uso:

Caso de Uso 1	Consultar Solución a Incidentes Comunes
Actor:	Usuario
Descripción:	El Usuario buscará la solución a incidencia suscitada dentro del sistema o lugar de trabajo.
Prioridad:	Obligatorio
Requisitos Asociados	
R1.1 El Usuario encontrará en primera instancia ayuda para incidencias comunes que pueda resolverlas el mismo sin necesidad de recurrir a personal de soporte, escogiendo el tipo de incidente común.	

Tabla 1.2 Descripción del caso de uso 1

Caso de Uso 2	Incidentes de Hardware
Actor:	Usuario
Descripción:	El Usuario buscará la solución a incidencia de tipo hardware suscitada.
Prioridad:	Obligatorio
Requisitos Asociados	
R2.1. El usuario buscará la solución a su incidencia dentro de las opciones que muestre el sistema en las que son de tipo hardware.	

Tabla 1.3 Descripción del caso de uso 2

Caso de Uso 3	Incidentes de Software
Actor:	Usuario
Descripción:	El Usuario buscará la solución a incidencia de tipo software suscitada.
Prioridad:	Obligatorio
Requisitos Asociados	
R3.1. El usuario buscará la solución a su incidencia dentro de las opciones que muestre el sistema en las que son de tipo software.	

Tabla 1.4 Descripción del caso de uso 3

Caso de Uso 4	Incidentes de Redes
Actor:	Usuario
Descripción:	El Usuario buscará la solución a incidencia de tipo utilitario suscitada.
Prioridad:	Obligatorio
Requisitos Asociados	
R4.1. El usuario buscará la solución a su incidencia dentro de las opciones que muestre el sistema en las que son de tipo redes.	

Tabla 1.5 Descripción del caso de uso 4

Caso de Uso 5	Solicitar Soporte Vía Correo Electrónico
Actor:	Usuario
Descripción:	El Usuario solicitará la ayuda al incidente suscitado vía correo electrónico.
Prioridad:	Obligatorio
Requisitos Asociados	
R5.1. El solicitará por medio de correo electrónico la solución a la incidencia suscitada al personal de soporte describiendo en breves rasgos la incidencia suscitada la cual no le permite seguir normalmente el curso de su trabajo.	

Tabla 1.6 Descripción del caso de uso 5

Caso de Uso 6	Envío del correo electrónico
Actor:	Usuario
Descripción:	El usuario procederá a enviar el correo electrónico.
Prioridad:	Obligatorio
Requisitos Asociados	
R6.1. Una vez llenado lo requerido el usuario procederá a enviar el correo electrónico pulsando el botón enviar, para solicitar el soporte necesario	

Tabla 1.7 Descripción del caso de uso 6

Caso de Uso 7	Recibir Correo Electrónico
Actor:	Soporte
Descripción:	El personal de soporte recibe el correo electrónico del usuario que solicitó ayuda.
Prioridad:	Obligatorio
Requisitos Asociados	
R7.1. A través del sistema llega el correo electrónico al personal de soporte por la necesidad de resolver el incidente.	

Tabla 1.8 Descripción del caso de uso 7

Caso de Uso 8	Comunicarse con Usuario
Actor:	Soporte
Descripción:	El personal de soporte se comunica inmediatamente con el usuario para ayudarlo con el incidente suscitado.
Prioridad:	Obligatorio
Requisitos Asociados	
R8.1. Una vez leído la petición del usuario por medio del correo electrónico el personal de soporte procede a comunicarse inmediatamente para ayudarlo a resolver el incidente.	
R8.2 La comunicación se podrá realizar vía VNC, telefónica o chat.	

Tabla 1.9 Descripción del caso de uso 8

Caso de Uso 9	Consultar Base Conocimiento
Actor:	Soporte
Descripción:	El personal de soporte procederá a consultar la Base de Conocimiento para dar solución al incidente.
Prioridad:	Obligatorio
Requisitos Asociados	
R9.1. Por medio del sistema el personal del soporte tendrá acceso a la Base de Conocimiento en la cual irá buscando todas las opciones que se encuentren almacenadas hasta llegar a solucionar el problema de forma inmediata.	

Tabla 1.10 Descripción del caso de uso 9

Caso de Uso 10	Encuentra Solución
Actor:	Soporte
Descripción:	Se procederá a dar por terminado el soporte al usuario una vez encontrada y aplicada la solución a la incidencia.
Prioridad:	Obligatorio
Requisitos Asociados	
R10.1. Una vez encontrada la solución y aplicada la misma al usuario implicado se procede a dar el cierre de soporte.	
R10.2 A su vez el personal de soporte llenará una Bitácora de Actividades indicando el usuario al cual atendió, fecha y hora del suceso, así como el tipo de incidente y el proceso para solucionar el mismo.	

Tabla 1.11 Descripción del caso de uso 10

Caso de Uso 11	Buscar causas de incidentes
Actor:	Soporte
Descripción:	El personal de soporte luego de no encontrar solución dentro de la Base de Conocimiento, comenzará a buscar las posibles causas de la incidencia.
Prioridad:	Obligatorio
Requisitos Asociados	
R11.1. Una vez desechadas todas las posibles causas y dentro de las mismas encontrada y aplicada la solución al incidente suscitado por el usuario se procede al cierre del caso de incidencia.	

Tabla 1.12 Descripción del caso de uso 11

Caso de Uso 12	Guardar Base Conocimiento
Actor:	Soporte
Descripción:	El personal de soporte guarda en la Base de Conocimiento el proceso de la solución aplicada al incidente.
Prioridad:	Obligatorio
Requisitos Asociados	
R12.1 Por medio del sistema el personal de soporte que resolvió el incidente presentado por el usuario, accederá a la Base de Conocimiento para guardar el proceso de solución del incidente.	

R12.2 A su vez procederá a llenar una Bitácora de Actividades indicando el personal de soporte al que realizó escalamiento, usuario del incidente, fecha y hora de lo ocurrido, así como el proceso de solución.

Tabla 1.13 Descripción del caso de uso 12

Caso de Uso 13	No encuentra solución
Actor:	Soporte
Descripción:	El personal de soporte luego de no encontrar la solución al incidente procederá a realizar un escalamiento, solicitando ayuda a alguien de mayor nivel de conocimiento sobre el incidente.
Prioridad:	Obligatorio
Requisitos Asociados	
R13.1 Se realiza un escalamiento implícito de personal de soporte hacia un nivel superior y a su vez esto es llenado en la Bitácora de Actividades.	

Tabla 1.14 Descripción del caso de uso 13

1.3.5 Requisitos de interfaces externos

En esta parte del capítulo se describen los requisitos que afecten a las diferentes interfaces para todos los usuarios que van a hacer uso del sistema.

1.3.5.1 Interfaces de Software

La interfaz para poder visualizar la aplicación web debe ser un navegador de internet como Opera, Google Chrome, Firefox o Safari, Internet Explorer tiene conflictos con el menú y no se puede mostrar correctamente, no se recomienda utilizar este navegador de internet.

1.3.5.2 Interfaces de Usuario

El sistema de Helpdesk mediante el uso de entorno web ofrecerá una interfaz intuitiva, equilibrada, fácil de aprender y manejar.

Deberá contar con una capacidad de respuesta rápida, consistente, estética para el usuario, muy eficiente, tener un alto grado de usabilidad; el usuario deberá poder familiarizar rápidamente en cuestión de pocas horas con el sistema.

Las interfaces de los usuarios se manejarán en un entorno Web y deberán seguir los estándares acordados dentro del área de desarrollo.

CAPITULO 2. DISEÑO

2.1 Diagramado UML del Sistema de HelpDesk mediante el uso de Web Service.

2.1.1 Introducción a UML

El Lenguaje Unificado de Modelado (UML) es el lenguaje de modelado más utilizado en la actualidad, es de tipo gráfico que nos sirve para visualizar, especificar, construir y documentar un sistema. UML ofrece un estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes reutilizables.

El diagramado UML posee 13 tipos de diagramas divididos en 13 grupos:

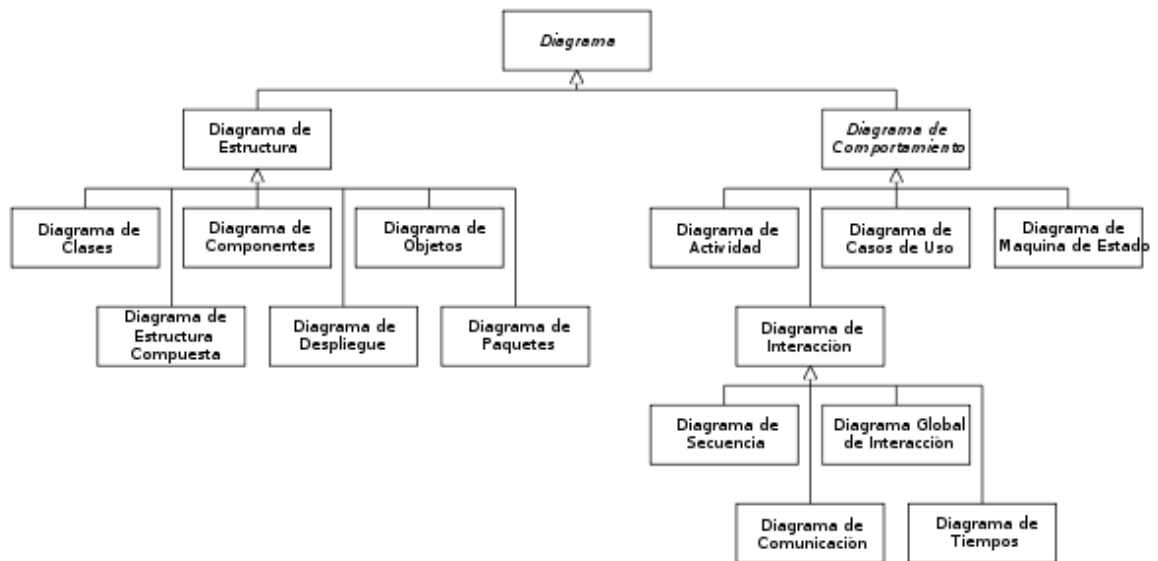


Figura 2.1 Descripción de Diagramado UML

Los **Diagramas de Estructura** enfatizan en los elementos que deben existir en el sistema modelado:

- Diagrama de clases
- Diagrama de componentes
- Diagrama de objetos
- Diagrama de estructura compuesta (UML 2.0)
- Diagrama de despliegue
- Diagrama de paquetes

Los **Diagramas de Comportamiento** enfatizan en lo que debe suceder en el sistema modelado:

- Diagrama de actividades
- Diagrama de casos de uso
- Diagrama de estados

Los **Diagramas de Interacción** son un subtipo de diagramas de comportamiento, que enfatiza sobre el flujo de control y de datos entre los elementos del sistema modelado:

- Diagrama de secuencia
- Diagrama de comunicación, que es una versión simplificada del Diagrama de colaboración (UML 1.x)
- Diagrama de tiempos (UML 2.0)

- Diagrama global de interacciones o Diagrama de vista de interacción (UML 2.0)

Para la realización de los diagramas nombrados anteriormente se utilizará Microsoft Visio 2007, dentro de la sección Software y Base de Datos en Diagrama de Modelo de UML.

2.1.2 Diagrama de Clases

El diagrama de clases describe la estructura del sistema por medio de las clases, con sus atributos, operaciones y relaciones que tienen entre sí. Son utilizados durante el análisis y diseño de los sistemas, de una manera más conceptual con la información que se manipulará dentro del mismo y sus componentes.

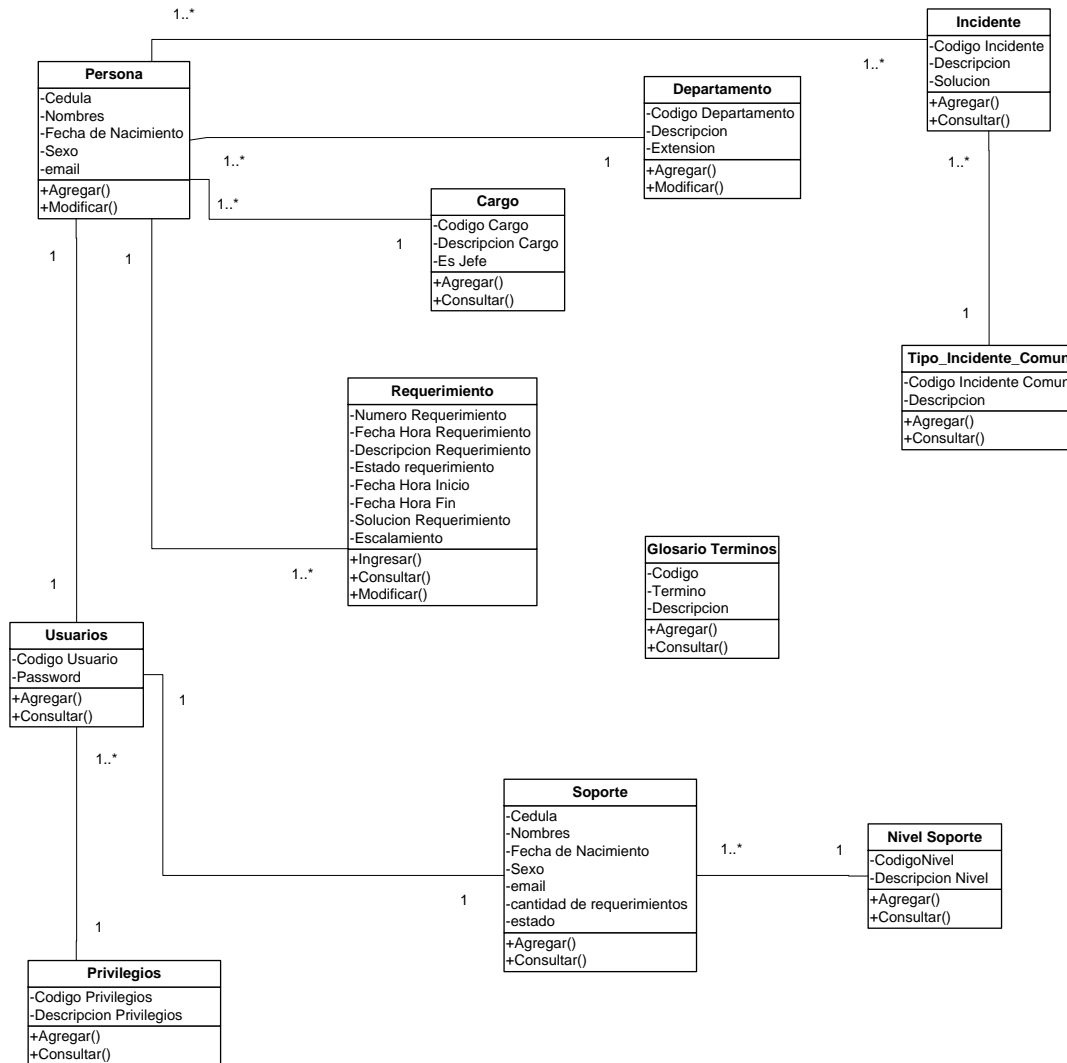


Figura 2.2 Diagrama de Clases

2.2 Diseño de la Base de Datos

Para el diseño, desarrollo e implementación de la Base de Datos he optado por el Gestor de Base de Datos MySQL, mismo que es relacional; licenciado bajo la licencia **GNU GPL**, a su vez es multihilo y multiusuario lo cual nos permite gran carga de transacciones de una forma eficiente.

Aunque MySQL AB distribuye una versión comercial, en lo único que se diferencia de la versión libre, es en el soporte técnico que se ofrece, y la posibilidad de integrar este gestor en un software propietario, ya que de otra manera, se vulneraría la licencia GPL.

Características

Inicialmente, MySQL carecía de elementos considerados esenciales en las bases de datos relacionales, tales como integridad referencial y transacciones. A pesar de ello, atrajo a los desarrolladores de páginas web con contenido dinámico, justamente por su simplicidad.

Poco a poco los elementos de los que carecía MySQL están siendo incorporados tanto por desarrollos internos, como por desarrolladores de software libre. Entre las características disponibles en las últimas versiones se puede destacar:

- Amplio subconjunto del lenguaje SQL. Algunas extensiones son incluidas igualmente.
- Disponibilidad en gran cantidad de plataformas y sistemas.
- Posibilidad de selección de mecanismos de almacenamiento que ofrecen diferente velocidad de operación, soporte físico, capacidad, distribución geográfica, transacciones...
- Transacciones y claves foráneas.
- Conectividad segura.

- Replicación.
- Búsqueda e indexación de campos de texto.

MySQL es un sistema de administración de bases de datos. Una base de datos es una colección estructurada de tablas que contienen datos. Esta puede ser desde una simple lista de compras a una galería de pinturas o el vasto volumen de información en una red corporativa. Para agregar, acceder a y procesar datos guardados en un computador, usted necesita un administrador como MySQL Server. Dado que los computadores son muy buenos manejando grandes cantidades de información, los administradores de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones.

MySQL es un sistema de administración relacional de bases de datos. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido.

MySQL es software de fuente abierta. Fuente abierta significa que es posible para cualquier persona usarlo y modificarlo. Cualquier persona puede bajar el código fuente de MySQL y usarlo sin pagar. Cualquier interesado puede estudiar el código fuente y ajustarlo a sus necesidades. MySQL usa el GPL (GNU General Public License) para definir qué puede hacer y que no puede hacer con el software en diferentes situaciones. Si usted no se ajusta al GPL o requiere introducir código MySQL en aplicaciones comerciales, usted puede comprar una versión comercial licenciada.

Características distintivas

Las siguientes características son implementadas únicamente por MySQL:

- Múltiples motores de almacenamiento (MyISAM, Merge, InnoDB, BDB, Memory/heap, MySQL Cluster, Federated, Archive, CSV, Blackhole y Example en 5.x), permitiendo al usuario escoger la que sea más adecuada para cada tabla de la base de datos.
- Agrupación de transacciones, reuniendo múltiples transacciones de varias conexiones para incrementar el número de transacciones por segundo.

Tipos de compilación del servidor

Hay tres tipos de compilación del servidor MySQL:

- Estándar: Los binarios estándares de **MySQL** son los recomendados para la mayoría de los usuarios, e incluyen el motor de almacenamiento InnoDB.
- Max (No se trata de MaxDB, que es una cooperación con SAP): Los binarios incluyen características adicionales que no han sido lo bastante probadas o que normalmente no son necesarias.

- MySQL-Debug: Son binarios que han sido compilados con información de depuración extra. No debe ser usada en sistemas en producción porque el código de depuración puede reducir el rendimiento.

Especificaciones del código fuente

MySQL está escrito en una mezcla de C y C++. Hay un documento que describe algunas de sus estructuras internas en:

<http://dev.mysql.com/doc/internals/en>

Otras funcionalidades de las listas de correo

- Anuncios: informan sobre nuevas versiones de MySQL y programas relacionados.
- MySQL: lista principal de discusión de MySQL y sql.
- Bugs: permite a la gente buscar y arreglar bugs.
- Temas internos: para gente que trabaja con el código de MySQL. Es el fórum para discutir sobre el desarrollo de MySQL.
- MySQLdoc: para gente que trabaja en documentación.
- Pruebas de rendimiento: para gente interesada en temas de rendimiento no sólo de MySQL, sino de otros motores de bases de datos.
- Empaquetadores: para discusiones sobre empaquetamiento y distribución de MySQL.
- Java: discusiones sobre MySQL Server y Java.

Otras listas de correo son: MyODBC, Herramientas GUI, Cluster, Dotnet, PlusPlus y Perl.

Adicional a las listas de correo, se encuentra el soporte de IRC de la comunidad MySQL. Además, hay soporte a través de foros agrupados en categorías tales como: Migración, Uso de MySQL, Conectores MySQL, Tecnología MySQL y Negocios.

Características adicionales:

- Usa GNU Automake, Autoconf, y Libtool para portabilidad.
- Uso de multihilos mediante hilos del kernel.
- Usa tablas en disco b-tree para búsquedas rápidas con compresión de índice.
- Tablas hash en memoria temporales.
- El código MySQL se prueba con Purify (un detector de memoria perdida comercial) así como con Valgrind, una herramienta GPL.
- Completo soporte para operadores y funciones en cláusulas select y where.
- Completo soporte para cláusulas group by y order by, soporte de funciones de agrupación.
- **Seguridad:** ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor.
- Soporta gran cantidad de datos. MySQL Server tiene bases de datos de hasta 50 millones de registros.
- Se permiten hasta 64 índices por tabla (32 antes de MySQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de

columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2).

- Los clientes se conectan al servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows se pueden conectar usando named pipes y en sistemas Unix usando ficheros socket Unix.
- En MySQL 5.0, los clientes y servidores Windows se pueden conectar usando memoria compartida.
- MySQL contiene su propio paquete de pruebas de rendimiento proporcionado con el código fuente de la distribución de MySQL.

2.2.1 Diseño del modelo Entidad Relación de la Base de Datos

El modelo Entidad Relación es una herramienta para el modelado de datos de un sistema de información. Estos modelos expresan entidades relevantes para un sistema de información así como sus interrelaciones y propiedades.

El modelo de datos entidad-relación está basado en una percepción del mundo real que consta de una colección de objetos básicos, llamados entidades, y de relaciones entre esos objetos.

A Continuación se detalla el Modelo Entidad Relación desarrollado en MySQL Workbench:

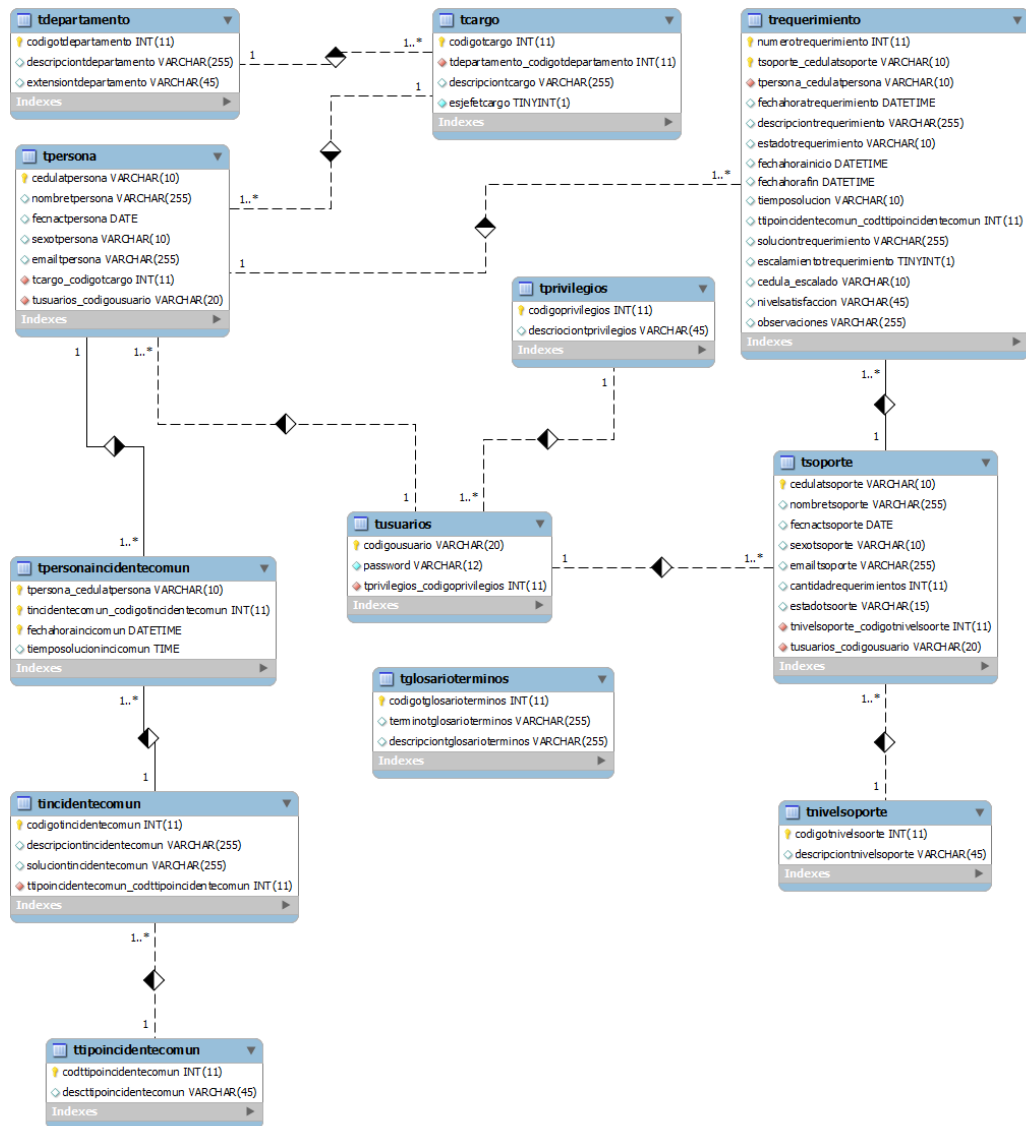


Figura 2.3 Diagrama Modelo Entidad Relación

2.2.2 Normalización del Modelo:

El proceso de normalización de bases de datos consiste en aplicar una serie de reglas a las relaciones obtenidas tras el paso del modelo entidad-relación al modelo relacional.

El proceso de normalización de bases de datos consiste en aplicar una serie de reglas a las relaciones obtenidas tras el paso del modelo entidad-relación al modelo relacional.

Las bases de datos relacionales se normalizan para:

- Evitar la redundancia de los datos.
- Evitar problemas de actualización de los datos en las tablas.
- Proteger la integridad de los datos.

En el modelo relacional es frecuente llamar tabla a una relación, aunque para que una tabla sea considerada como una relación tiene que cumplir con algunas restricciones:

- Cada tabla debe tener su nombre único.
- No puede haber dos filas iguales. No se permiten los duplicados.
- Todos los datos en una columna deben ser del mismo tipo.

A continuación se detalla el modelo relacional o normalizado desarrollado en Visio 2007:

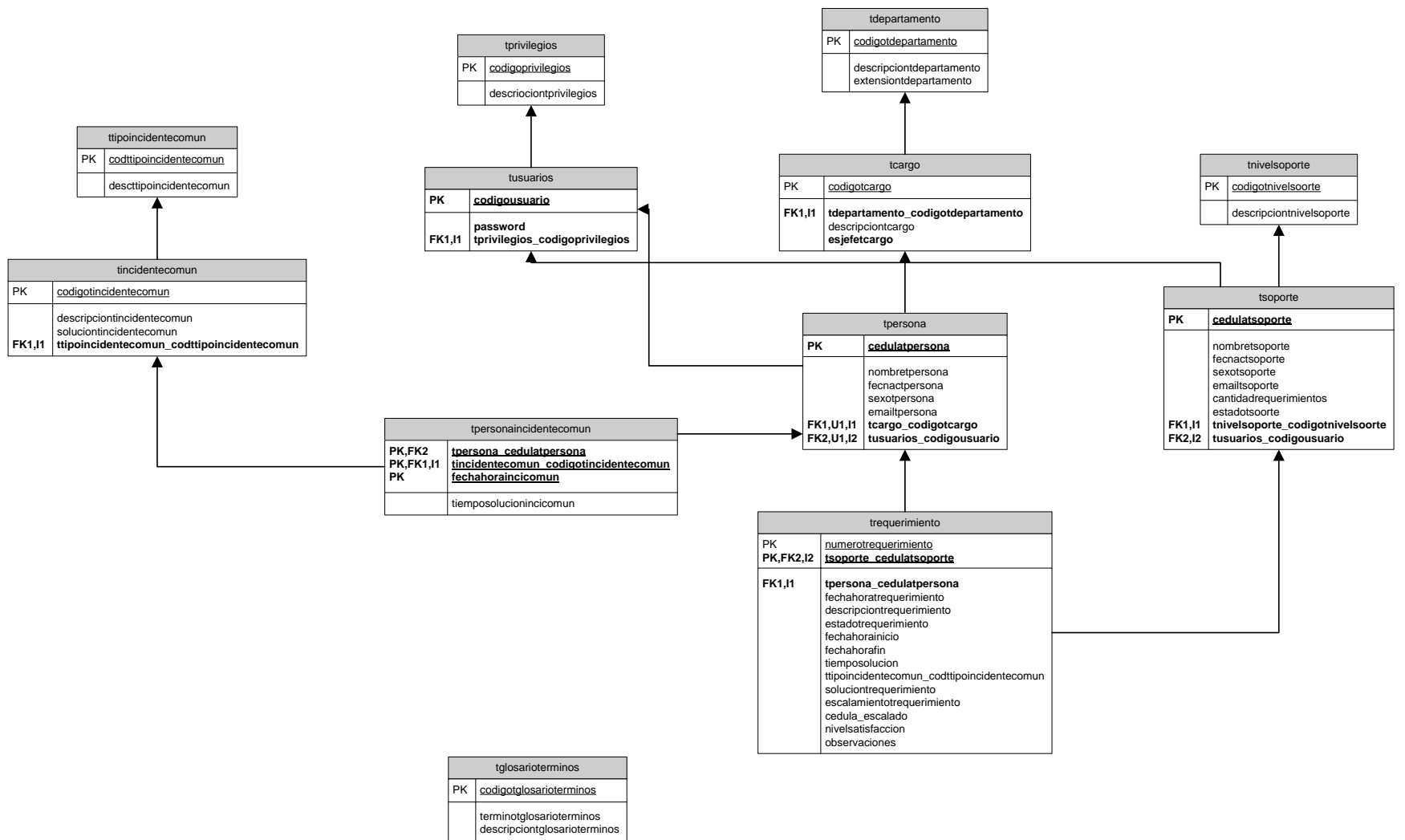


Figura 2.4 Diagrama Modelo Relacional

2.2.3 Definición del diccionario de datos

Un diccionario de datos es un conjunto de metadatos que contiene las características lógicas y puntuales de los datos que se van a utilizar en el sistema que se programa, incluyendo nombre, descripción, alias, contenido y organización.

Identifica los procesos donde se emplean los datos y los sitios donde se necesita el acceso inmediato a la información, se desarrolla durante el análisis de flujo de datos y auxilia a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño.

En un diccionario de datos se encuentra la lista de todos los elementos que forman parte del flujo de datos de todo el sistema. Los elementos más importantes son flujos de datos, almacenes de datos y procesos. El diccionario de datos guarda los detalles y descripción de todos estos elementos.

A continuación se detalla el diccionario de datos:

tcargo				
Nombre Campo	Tipo	Llave	Nulo	Descripción
codigotcargo	int(11)	PK	No	Campo identificador del cargo
tdepartamento_codigotdepartamento	int(11)	FK	No	Llave foránea proveniente de la tabla tdepartamento
descripciontcargo	varchar(255)		Yes	Descripción del cargo
esjefetcargo	tinyint(1)		No	Campo que describe si es jefe

Tabla 2.1 Tabla tcargo

tdepartamento				
Nombre Campo	Tipo	Llave	Nulo	Descripción
codigotdepartamento	int(11)	PK	No	Campo identificador del departamento
descripciontdepartamento	varchar(255)		Yes	Descripción del departamento
extensiontdepartamento	varchar(45)		Yes	Número telefónico del departamento

Tabla 2.2 Tabla tdepartamento

tglosarioterminos				
Nombre Campo	Tipo	Llave	Nulo	Descripción
codigotglosarioterminos	int(11)	PK	No	Campo que identifica al término
terminotglosarioterminos	varchar(255)		Yes	Término informático
descripciontglosarioterminos	varchar(255)		Yes	Pequeña explicación a lo que hace referencia el término informático

Tabla 2.3 Tabla tglosarioterminos

Tincidentecomun				
Nombre Campo	Tipo	Llave	Nulo	Descripción
codigotincidentecomun	int(11)	PK	No	Campo que identifica al incidente común
descripciontincidentecomun	varchar(255)		Yes	Descripción del incidente
soluciontincidentecomun	varchar(255)		Yes	Solución del incidente
ttipoincidentecomun_codttipoincidentecomun	int(11)	FK	No	Llave foránea venida de la tabla ttipoincidentecomun

Tabla 2.4 Tabla *tincidentecomun*

tnivelsoporte				
Nombre Campo	Tipo	Llave	Nulo	Descripción
codigotnivelsoorte	int(11)	PK	No	Campo que identifica al nivel de soporte
descripciontnivelsoporte	varchar(45)		Yes	Indica si es nivel normal o avanzado

Tabla 2.5 Tabla *tnivelsoporte*

tpersona				
Nombre Campo	Tipo	Llave	Nulo	Descripción
cedulatpersona	varchar(10)	PK	No	Campo que identifica a la persona
nombretpersona	varchar(255)		Yes	Nombre de la persona
fecnactpersona	date		Yes	Fecha de Nacimiento
sexotpersona	varchar(10)		Yes	Sexo
emailtpersona	varchar(255)		Yes	Correo electrónico de la persona
tcargo_codigotcargo	int(11)	FK	No	Llave foránea que viene desde la tabla tcargo
tusuarios_codigousuario	varchar(20)	FK	No	LLlave foránea que viene desde la tabla tusuarios

Tabla 2.6 Tabla tpersona

tpersonaincidentecomun				
Nombre Campo	Tipo	Llave	Nulo	Descripción
tpersona_cedulatpersona	varchar(10)	PK	No	Llave que viene a armar la llave primaria desde la tabla tpersona
tincidentecomun_codigotincidentecomun	int(11)	PK	No	Llave que viene a armar la llave primaria desde la tabla tincidentecomun
fechahoraincicomun	datetime	PK	No	Llave que viene a armar la llave primaria, contiene la fecha-hora del incidente
tiemposolucioninccomun	time		Yes	Tiempo que demoró en solucionar el

				incidente
--	--	--	--	-----------

Tabla 2.7 Tabla tpersonaincidentecomun

tprivilegios				
Nombre Campo	Tipo	Llave	Nulo	Descripción
codigoprivilegios	int(11)	PK	No	Campo que identifica al privilegio de acceso al sistema
descripciontprivilegios	varchar(45)		Yes	Descripción del privilegio al sistema

Tabla 2.8 Tabla tprivilegios

trequerimiento				
Nombre Campo	Tipo	Llave	Nulo	Descripción
numerotrequerimiento	int(11)	PK	No	Campo que identifica al número del requerimiento solicitado
tsoporte_cedulatsoporte	varchar(10)	PK	No	Llave que viene desde la tabla tsoporte y forma parte de la llave primaria
tpersona_cedulapersona	varchar(10)	FK	No	Cédula de la persona que solicitó el requerimiento de soporte
fechahoratrequerimiento	datetime		Yes	Fecha y hora a la que se hizo el requerimiento
descripciontrequerimiento	varchar(255)		Yes	Descripción del incidente
estadotrequerimiento	varchar(10)		Yes	Estado en el que se encuentra
fechahorainicio	datetime		Yes	Fecha hora en la que inició el soporte

fechahorafin	datetime		Yes	Fecha hora en la que finalizó el soporte
tiemposolucion	varchar(10)		Yes	Tiempo que tomó la solución del incidente
ttipoincidentecomun_codttipoincidentecomun	int(11)		Yes	Tipo de incidente
soluciontrequerimiento	varchar(255)		Yes	Solución que se le dio al incidente
escalamientotrequerimiento	tinyint(1)		Yes	Si es que tuvo escalamiento el incidente
cedula_escalado	varchar(10)		Yes	Cédula del personal de soporte a quien escaló el incidente
nivelsatisfaccion	varchar(45)		Yes	Nivel de Satisfacción del soporte recibido
observaciones	varchar(255)		Yes	Observaciones hechas por el usuario que recibio soporte

Tabla 2.9 Tabla trequerimiento

tsoporte				
Nombre Campo	Tipo	Llave	Nulo	Descripción
cedulatsoporte	varchar(10)	PK	No	Campo que identifica al personal de soporte
nombretsoporte	varchar(255)		Yes	Nombre del personal de soporte
fecnactsoporte	date		Yes	Fecha de nacimiento del personal de soporte
sexotsoporte	varchar(10)		Yes	Sexo
emailtsoporte	varchar(255)		Yes	Correo electrónico del personal de soporte

cantidadrequerimientos	int(11)		Yes	Número de requerimientos asignados
estadotsoorte	varchar(15)		Yes	Estado de soporte
tnivelsoporte_codigotnivelsoorte	int(11)	FK	No	Llave foránea venida desde la tabla tnivelsoporte
tusuarios_codigousuario	varchar(20)	FK	No	Llave foránea venida desde la tabla tusuarios

Tabla 2.10 Tabla tsoporte

ttipoincidentecomun				
Nombre Campo	Tipo	Llave	Nulo	Descripción
codttipoincidentecomun	int(11)	PK	No	
descttipoincidentecomun	varchar(45)		Yes	

Tabla 2.11 Tabla ttipoincidentecomun

tusuarios				
Nombre Campo	Tipo	Llave	Nulo	Descripción
codigousuario	varchar(20)	PK	No	Campo que identifica al usuario del sistema
password	varchar(12)		No	Clave para el ingreso al sistema
tprivilegios_codigoprivilegios	int(11)	FK	No	Llave foránea que viene desde la tabla tprivilegios

Tabla 2.12 Tabla tusuarios

2.3 Diseño de la Aplicación Helpdesk mediante el uso de entorno web

La aplicación Sistema Helpdesk mediante el uso de entorno web se la desarrollará en php, y estará dirigida al soporte de incidentes mediante correo electrónico.

2.3.1 Definición de los niveles de seguridad y privilegios

El sistema de helpdesk cuenta con tres tipos de usuarios, los mismos que son: personal de soporte, personal empleados, y director.

El director es el encargado de supervisar que se esté cumpliendo el servicio de helpdesk a cabalidad, que el personal de soporte esté brindando la ayuda necesaria, llenado de bitácora, así como también que los usuarios califiquen el soporte recibido.

Tendrá acceso a consultas y en base de las mismas podrá controlar a personal de soporte y empleados.

El personal de soporte son usuarios del departamento de sistemas que tienen acceso a toda la aplicación para realizar cualquier mantenimiento u operación, siendo estas las siguientes:

- Mantenimiento de Personas.
- Mantenimiento de Departamentos.
- Ingreso de Incidentes Comunes.
- Llenado de Bitácora.

- Consulta Base de Conocimiento.

El personal de empleados son usuarios normales y jefes que tendrán acceso a la ayuda online y a su vez a la solicitud de soporte, sus operaciones son las siguientes:

- Consulta de incidentes comunes por tipo para encontrar una solución rápida.
- Solicitud de soporte vía correo electrónico llenando la descripción y recibiendo la misma inmediatamente una vez haya llegado el correo electrónico al personal de soporte.
- Calificación de soporte recibido.

2.3.2 Plantilla a ser utilizada en el sitio web

Para el diseño del sitio Web se ha elaborado una plantilla que será utilizada en todas las interfaces de los usuarios, llegando a diferenciarse solamente por los menús dependiendo del tipo de usuario.

La plantilla abarca los estándares para la satisfacción del usuario final.

Consta de 4 partes la plantilla:

- Cabecera.
- Menú.
- Contenido.

- Pie de página.

La estructura es la siguiente:



Figura 2.5 Plantilla del Sistema Helpdesk

El menú tendrá las siguientes opciones dependiendo del tipo de usuario que ingrese:

Director:

- Incidentes Comunes
 - Usuarios
- Requerimientos
 - No Atendidos
 - Atendidos
 - Iniciados

- Proceso
 - Finalizados
 - Escalados
- Calificación Soporte
 - General
 - Niveles
- Tiempos
- Calificación
- Salir

Soporte:

- Base de Conocimiento
- Bitácora
- Personas
 - Ingreso
 - Modificación
- Incidentes Comunes
 - Ingreso
- Departamentos
 - Ingreso
 - Modificación
- Salir

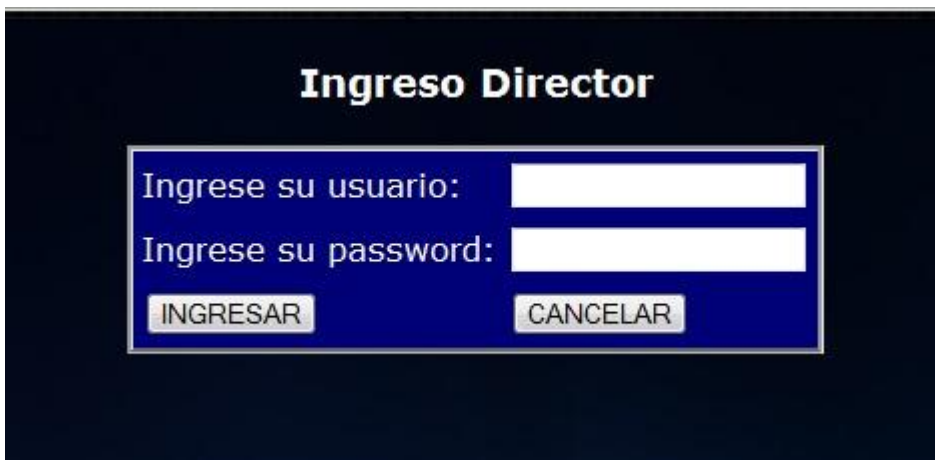
Usuarios:

- Incidentes Comunes
 - Hardware
 - Software
 - Redes
- Solicitar Soporte

- Requerimiento
 - Calificar
- Salir

2.3.2.1 Ventana de autenticación de usuarios.

La aplicación al momento de autenticarse para el ingreso a la misma tiene la misma ventana para los tres tipos de usuarios, lo único diferente es el título:



The image shows a login window titled "Ingreso Director". It features a dark blue background with white text. The window contains two input fields: "Ingrese su usuario:" and "Ingrese su password:". Below the input fields are two buttons: "INGRESAR" and "CANCELAR".

Figura 2.6 Ingreso de Usuarios

2.3.2.2 Páginas de Agregar.

Las páginas de agregar son para incidentes comunes, personas y departamentos y están disponibles dentro del menú de soporte, varían y su diseño es el siguiente:

Agregar personas:

Agregar Persona

Ingrese el número de cédula:	<input type="text"/>
Ingrese el nombre:	<input type="text"/>
Ingrese la fecha de nacimiento:	<input type="text"/> ...
Sexo:	--Escoja Sexo-- ▾
Correo Electrónico:	<input type="text"/>
Escoja Cargo:	--Escoja Cargo-- ▾
<input type="button" value="GRABAR"/>	<input type="button" value="CANCELAR"/>

Figura 2.7 Agregar Personas

Agregar Incidente Común:

Agregar Incidente Común

Describa Incidente:	<input type="text"/>
Describa Solucion:	<input type="text"/>
Escoja Tipo de incidente	--Tipo Incidente-- ▾
<input type="button" value="Grabar"/>	<input type="button" value="Cancelar"/>

Figura 2.8 Agregar Incidente Común

Agregar Departamento:

Agregar Departamento

Ingrese nombre departamento:	<input type="text"/>
Ingrese extension:	<input type="text"/>
<input type="button" value="GRABAR"/>	<input type="button" value="CANCELAR"/>

Figura 2.9 Agregar Departamento

2.3.2.3 Páginas de Modificación.

Las páginas de modificación funcionan similarmente, con filtros de búsqueda en la primera sección, luego en la segunda se verán los datos a modificar:

Modificación de Personas

Correo:	<input type="text" value="viejopeteras@gmail.com"/>
Cargo:	<input type="text" value="JEFE DE CONTABILIDAD"/>
Nuevo Cargo:	<input type="text" value="---Seleccione el cargo---"/>
<input type="button" value="MODIFICAR"/>	<input type="button" value="CANCELAR"/>

Figura 2.10 Modificación de Personas

2.3.2.4 Base de Conocimiento.

El usuario de soporte accede e ingresa una palabra clave para la búsqueda de la solución dentro de la Base de Conocimiento, dentro de dos signos porcentuales %palabra_clave%:

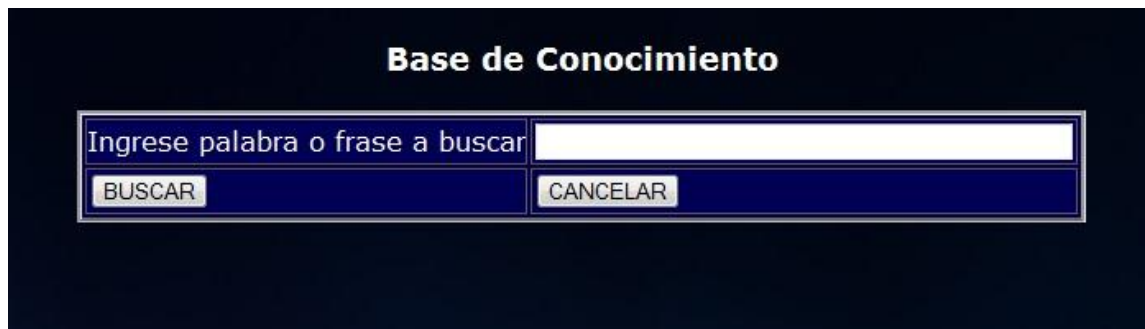
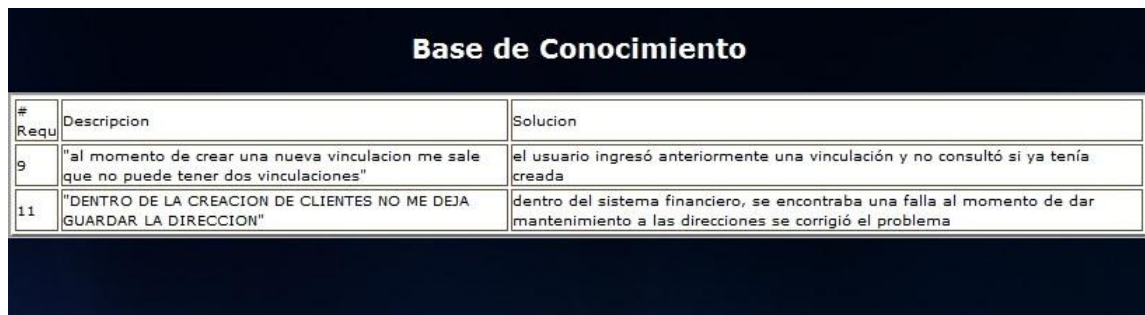


Figura 2.11 Búsqueda Base de Conocimiento

Después le aparecerá la o las posibles soluciones al incidente que le está dando seguimiento:



# Requ	Descripcion	Solucion
9	"al momento de crear una nueva vinculacion me sale que no puede tener dos vinculaciones"	el usuario ingresó anteriormente una vinculación y no consultó si ya tenía creada
11	"DENTRO DE LA CREACION DE CLIENTES NO ME DEJA GUARDAR LA DIRECCION"	dentro del sistema financiero, se encontraba una falla al momento de dar mantenimiento a las direcciones se corrigió el problema

Figura 2.12 Base de Conocimiento

2.3.2.5 Bitácora

Los usuarios de soporte luego de haber brindado ayuda a la persona con incidentes, y después de revisar la Base de Conocimiento, o encontrar la solución por medio propio procederán a llenar la bitácora, para que quede constancia del soporte brindado y también seguir alimentando más la Base de Conocimiento:

Primero contarán con una grilla en la cual aparecen el o los requerimientos solicitados:

Requerimientos de Soporte					
# Requ	Usuario	Fecha	Incidente	Extension	Llenar
8	JAIME RENE ASTUDILLO ORTIZ	2012-02-05 20:58:37	la maquina se demora en encender y en iniciar la sesion	10040	Ver
14	JAIME RENE ASTUDILLO ORTIZ	2012-02-05 21:48:43	no me permite grabar al nuevo cliente, me sale error de constraint	10040	Ver
4	MARIA EUGENIA SALINAS QUEVEDO	2012-02-05 20:54:10	LA IMPRESORA NO IMPRIME SE ENCUENTRA TRABADA	10040	Ver

Figura 2.13 Requerimientos de Soporte

Luego de haberse comunicado con el usuario afectado, llenará la bitácora realizando un clic en el enlace ver, que le llevará a la siguiente página en la que se encuentra la bitácora:

Llenar Bitácora

Numero de Requerimiento:	8
Cedula personal:	0101031250
Fecha del Requerimiento:	2012-02-05 20:58:37
Descripcion del Requerimiento:	"la maquina se demora en encender y en iniciar la sesion"
Estado del Requerimiento:	--Seleccione Estado--
Fecha Hora inicio:	<input type="text"/> ...
Fecha Hora finalizacion:	<input type="text"/> ...
Escoja el tipo de incidente:	--Escoja Tipo Incidente--
Describe la Solución:	<input type="text"/>
Escalamiento:	--ESCALAMIENTO--
<input type="button" value="GRABAR"/>	<input type="button" value="CANCELAR"/>

Figura 2.14 Bitácora

2.3.2.6 Incidentes Comunes

Al momento que un usuario requiera soporte primero tiene la opción de escoger entre los tipos de incidentes, de ahí tienen la misma pantalla, solamente deben ingresar una palabra o frase clave para buscar la solución al incidente suscitado:

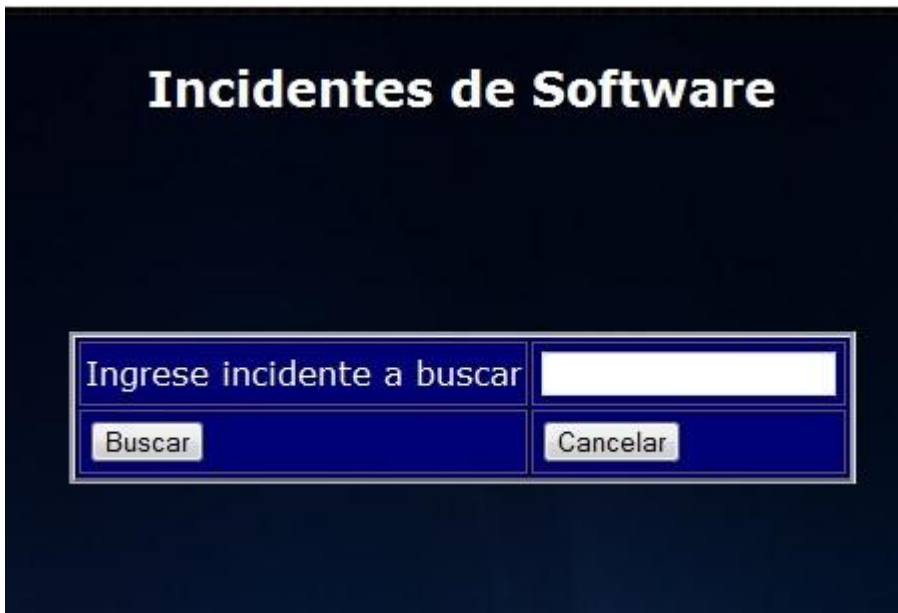


Figura 2.14 **Búsqueda de Incidentes**

Luego buscará la palabra clave y se saldrá una o varias opciones para el tipo de incidente que está buscando solucionar, así como luego podrá buscar la solución definitiva:

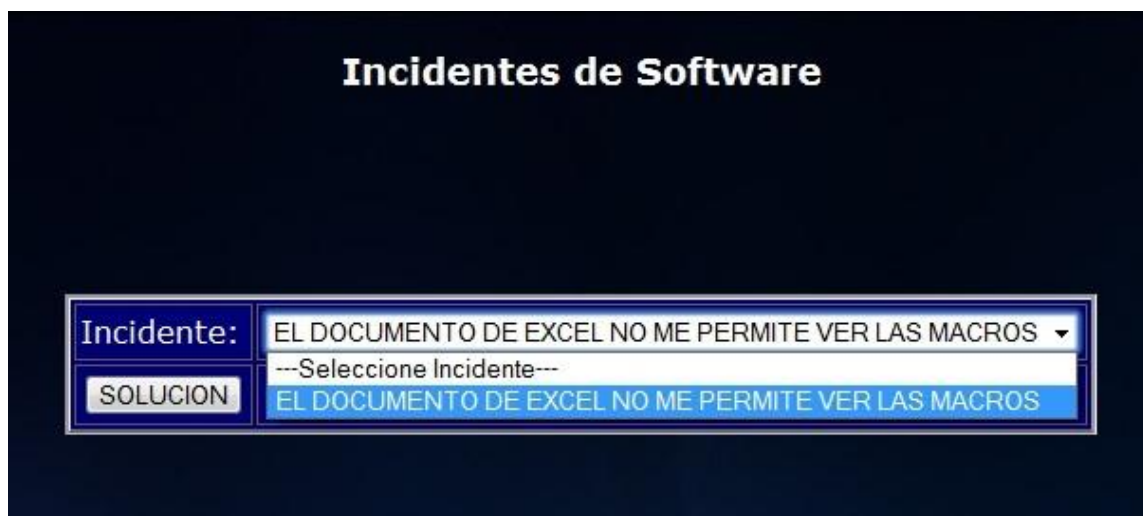


Figura 2.15 **Selección de Incidente**

Incidentes de Software

Solucion:	"IR A OPCIONES, SEGURIDAD, HABILITAR MACROS"
GRABAR	Volver al formulario

Figura 2.16 Solución a incidente común

2.3.2.7 Solicitar Soporte

Si es que el usuario no logró encontrar la solución a su incidente en la parte anteriormente explicada, procederá a enviar la solicitud de soporte mediante el sistema, detallando su inconveniente en breves rasgos; el cual inmediatamente se registrará en el sistema y estará visible para el personal de soporte y a su vez se enviará la petición vía correo electrónico:

Solicitar Soporte

Describa su incidente:	
SOLICITAR	CANCELAR

Figura 2.17 Solicitar Soporte

2.3.2.8 Calificar Requerimiento atendido

Luego de haber solicitado el soporte y ser atendido por el personal especializado, el usuario deberá calificar la ayuda recibida:

Calificar Requerimiento					
# Requ	Soporte	Fecha	Incidente	Solucion	Calificar
11	ESTEBAN SANTIAGO LEON MUÑOZ	2012-02-05 20:53:55	"DENTRO DE LA CREACION DE CLIENTES NO ME DEJA GUARDAR LA DIRECCION"	dentro del sistema financiero, se encontraba una falla al momento de dar mantenimiento a las direcciones se corrigió el problema	Calificar

Figura 2.18 Calificar Requerimiento

Seleccionará el requerimiento atendido para calificar:

Calificar Soporte Recibido	
Nivel de calificación:	--Seleccione Nivel--
Anote alguna observación:	--Seleccione Nivel--
	1 -- MALO
	2 -- REGULAR
	3 -- BUENO
	4 -- MUY BUENO
5 -- EXCELENTE	
GRABAR	Cancelar

Figura 2.19 Calificación

CAPITULO 3 CODIFICACION

3.1 Codificación de la aplicación web

3.1.1 Introducción a HTML

El HTML, Hyper Text Markup Language (Lenguaje de marcación de Hipertexto) es el lenguaje de marcas de texto utilizado normalmente en la www (World Wide Web). Fue creado en 1986 por el físico nuclear Tim Berners-Lee; el cual tomo dos herramientas preexistentes: El concepto de Hipertexto (Conocido también como link o ancla) el cual permite conectar dos elementos entre si y el SGML (Lenguaje Estándar de Marcación General) el cual sirve para colocar etiquetas o marcas en un texto que indique como debe verse. HTML no es propiamente un lenguaje de programación como C++, Visual Basic, etc., sino un sistema de etiquetas. HTML no presenta ningún compilador, por lo tanto algún error de sintaxis que se presente éste no lo detectará y se visualizara en la forma como éste lo entienda.

El entorno para trabajar HTML es simplemente un procesador de texto, como el que ofrecen los sistemas operativos Windows (Bloc de notas), UNIX (el editor vi o ed) o el que ofrece MS Office (Word). El conjunto de etiquetas que se creen, se deben guardar con la extensión .htm o .HTML

Estos documentos pueden ser mostrados por los visores o "browsers" de páginas Web en Internet, como Netscape Navigator, Mozilla Firefox, Safari, Opera y Microsoft Internet Explorer.

También existe el HTML Dinámico (DHTML), que es una mejora de Microsoft de la versión 4.0 de HTML que le permite crear efectos especiales como, por ejemplo,

texto que vuela desde la página palabra por palabra o efectos de transición al estilo de anuncio publicitario giratorio entre página y página.

3.1.2 Introducción a PHP

PHP es un acrónimo recursivo que significa *PHP Hypertext Pre-processor* (inicialmente *PHP Tools*, o, *Personal Home Page Tools*). Fue creado originalmente por Rasmus Lerdorf en 1994; sin embargo la implementación principal de PHP es producida ahora por The PHP Group y sirve como el estándar de facto para PHP al no haber una especificación formal. Publicado bajo la PHP License, la Free Software Foundation considera esta licencia como software libre.

Puede ser desplegado en la mayoría de los servidores web y en casi todos los sistemas operativos y plataformas sin costo alguno. El lenguaje PHP se encuentra instalado en más de 20 millones de sitios web y en un millón de servidores, el número de sitios en PHP ha compartido algo de su preponderante dominio con otros nuevos lenguajes no tan poderosos desde agosto de 2005. El sitio web de Wikipedia está desarrollado en PHP. Es también el módulo Apache más popular entre las computadoras que utilizan Apache como servidor web.

3.1.3 Introducción a Javascript

JavaScript es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas, en bases de datos locales al navegador...⁴ aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

JavaScript se diseñó con una sintaxis similar al C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo Java y JavaScript no están relacionados y tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. JavaScript se interpreta en el agente de usuario, al mismo tiempo que las sentencias van descargándose junto con el código HTML.

3.2 Creación de Plantillas Mediante Adobe Dreamweaver

Para facilitar la digitación del código HTML, javascript, PHP, se ha escogido Adobe Dreamweaver, que nos ayuda enormemente al momento de la creación de plantillas en las cuales se basarán las páginas del sistema Web.

Para la creación de una plantilla nos vamos al menú Archivo → Nuevo, y dentro de varias opciones escogemos Plantilla en blanco, aquí tenemos varios modelos de plantillas, pero para la creación del sistema decidimos por HTML y sin ningún diseño para uno colocar las divisiones basadas en divs, cabeceras, etc.

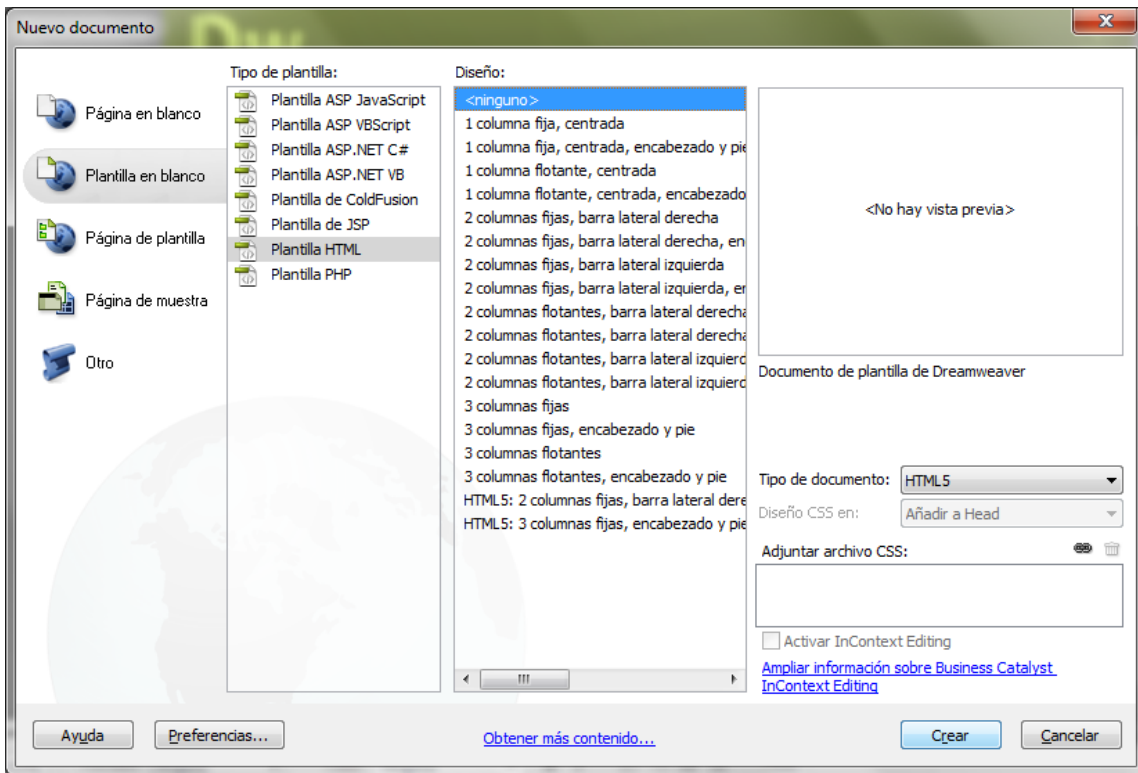


Figura 3.1 Creación de plantilla

3.2.1 Creación de Divs

Ya dentro de la plantilla, procedemos con el diseño de los divs que contendrán la cabecera, el contenedor, el menú y el pié que se aplicarán a las páginas basadas en la plantilla, todo esto estará contenido dentro del body:

```
<body>
<div class="container">
  <div class="div1">
    <h1 class="font_div1"> SISTEMA HELPDESK
  </h1>
</div>
<div class="div2">
  <p>&nbsp;</p>
  <p>&nbsp;</p>
  <ul id="MenuBar1" class="MenuBarVertical">
    <li><a href=" ../index.php">Home</a> </li>
    <li><a href=" ../indexadministrador.php"
class="MenuBarItemSubmenu">Director</a>
      <ul>
        <li><a href=" ../ingresodirector.php">Ingreso</a></li>
      </ul>
    </li>
    <li><a href="#" class="MenuBarItemSubmenu">Soporte</a>
      <ul>
        <li><a href=" ../ingresosistemas.php">Ingreso</a></li>
      </ul>
    </li>
    <li><a href="#" class="MenuBarItemSubmenu">Usuarios</a>
      <ul>
        <li><a href=" ../ingresousuarios.php">Ingreso</a></li>
      </ul>
    </li>
    <li><a href="#">Acerca de...</a></li>
```

```

</ul>
<div class="div3">
  <!-- TemplateBeginEditable name="EditRegion3" -->
  <!-- TemplateEndEditable -->
</div>
</div>
</div>
<div class="divfooter">
  <br>
  <p class="font_footer">SISTEMA HELPDESK, Pedro Esteban Astudillo,
Universidad del Azuay - Copyright© 2012</p>
</div>
<script type="text/javascript">
var MenuBar1 = new Spry.Widget.MenuBar("MenuBar1",
{imgRight:"../SpryAssets/SpryMenuBarRightHover.gif"});
</script>
</body>

```

3.2.2 Creación de CSS

Para que todas las páginas tengan el mismo tamaño de letras, tablas, etc. se ha creado un código de css:

```

<style type="text/css">
body {
  background-image:url(../Imagenes/fondo%20negro.jpg);
  margin: 0;
  padding: 0;
  font-family: Verdana, Arial, Helvetica, sans-serif;
  font-size: 100%;
  line-height: 1.4;

```

```

        color: #FFFFFF;
    }

.tables{
    border:groove;
    border-bottom-color:#BBB;
    background-color:#007;
}

.div1 {
    left:180px;
    background-image:url(../imagenes/fondo8.jpg);
    position:static;
    visibility: visible;
    height: 110px;
    width: 1022px;
    border-left:groove;
    border-left-color:#EEE;
    border-right:groove;
    border-right-color:#EEE;
    border-top:groove;
    border-top-color:#EEE;
    border-bottom:groove;
    border-bottom-color:#EEE;
}

.font_div1 {
    font:"Palatino Linotype", "Book Antiqua", "Palatino", "serif";
    font:bold;
    color:#FFFFFF;
    text-align:center;
}

.div2 {

```

```

background-image:url(../Imágenes/fondo10.jpg);
visibility:visible;
height:600px;
width:200px;
position:absolute;
border-left:groove;
border-left-color:#EEE;
border-right:groove;
border-right-color:#EEE;
border-bottom:groove;
border-bottom-color:#EEE;
}
.div3 {
background-image:url(../Imágenes/fondo3.jpg);
top:5px;
left:207px;
height:595px;
width:815px;
visibility:visible;
position:absolute;
border-right:groove;
border-right-color:#EEE;
border-bottom:groove;
border-bottom-color:#EEE;
}
.container {
width: 960px;
background: #FFFFFF;
margin: 0 auto; /* el valor automático de los lados, unido a la anchura,
centra el diseño */
}
.font_index {

```



```

        color:#FFFFFF;
        text-align:center;
        font:bold;
    }
    .divfooter {
        background-color:#111;
        width:1022px;
        height:80px;
        left:194px;
        top:606px;
        position:relative;
        border-left:groove;
        border-left-color:#EEE;
        border-right:groove;
        border-right-color:#EEE;
        border-top:groove;
        border-top-color:#EEE;
        border-bottom:groove;
        border-bottom-color:#EEE;
    }

    .font_footer {
        font:"Palatino Linotype", "Book Antiqua", "Palatino", "serif";
        font:bold;
        color:#FFFFFF;
        text-align:center;
        font-size:10px;
        font-style:italic;
    }
    td {
        padding:2px;
        width:auto;

```

```

}
.textos {
    text-transform:uppercase;
}
</style>

```

3.3 Conexión con la base de datos

3.3.1 Creación de archivo de conexión

Para poder acceder al sistema Web se debe ingresar digitando usuario y password, pero a su vez se necesita una conexión, la cual se realizó creando un archivo de conexión que va a ser llamado por las diferentes páginas al momento de realizar consultas:

```

<?php
$conexion=mysql_connect("localhost","root","endgame1320");
if(!$conexion)
{
echo("<center><h1>Error Al Conectarse al Servidor</h1></center>");exit;
}
else
{
mysql_select_db("helpdesk",$conexion);
}
?>

```

3.2 Verificación de autenticación

Para que usuarios avanzados, o personal de soporte que tengan conocimientos sobre programación web no puedan acceder a las diferentes páginas se trabajará con las sesiones para verificar que estén iniciados sesión o de no se redigirá a la página de inicio; para ello primero creamos un archivo .php que verifica que el usuario haya ingresado bien su clave y contraseña, y después se le redigirá a la página de index del respectivo usuario, soporte o director y ahí dentro de esa y el resto de páginas se validará que esté en sesión activa:

3.2.1 Código de verificación

```
<?php

$usuario=$_POST['txtusuario'];
$clave=$_POST['passusuario'];

include("conexion.php");

$sql=mysql_query("select codigousuario, password, tprivilegios_codigoprivilegios,
cedulatpersona, tcargo_codigotcargo
from `helpdesk`.`tusuarios`, `helpdesk`.`tpersona`
where codigousuario=tusuarios_codigousuario
and tprivilegios_codigoprivilegios in ('3','4')
and codigousuario='$usuario'
and password='$clave'", $conexion);

while ($regusuarios=mysql_fetch_array($sql))
{
$cedula=$regusuarios['cedulatpersona'];
};
```

```

$filas=mysql_affected_rows($conexion);

echo $filas;

if ($filas>0)
{
session_start();
$_SESSION['usuario']=$usuario;
$_SESSION['clave']=$clave;
$_SESSION['cedula']=$cedula;
$_SESSION['access'] = "true";
header("location:indexusuarios.php");
}
else
{
echo "<center>Nombre de Usuario o contrase&ntilde;a incorrecta</center>";
$_SESSION['access'] = "false";
}
?>

```

3.2.2 Código de verificación de sesión iniciada

Una vez ingresada a la página de inicio, de cada tipo de usuario, se validará que esté la sesión iniciada, de no se le regresará a la página de inicio:

```

<?php
//inicio de la sesion
session_start();
if($_SESSION['access']=="true") {

```

```

        echo "<img src='imagenes/help_desk.jpg' alt=" name='img_usuarios'
width='444' height='162' hspace='180' vspace='180'>";
    }
else {
    session_start();
    session_destroy();
    header("location:index.php");
    exit();
}
?>

```

3.3 Código para la consulta de incidentes comunes

Al momento que los usuarios acceden a la consulta de incidentes comunes se procede a consultar datos de la base mediante código php:

```

echo "<select name='incidenteredes'>";
echo "<option value='nada'>---Seleccione Incidente---</option>";
$registros=mysql_query("SELECT codigotincidentecomun,
descripcionincidentecomun
from tincidentecomun
where ttipoincidentecomun_codttipoincidentecomun=3 and
descripcionincidentecomun like '$incidente", $conexion)
or die ("Problemas en el select de la forma:".mysql_error());
while ($reg=mysql_fetch_array($registros))
{
echo"<option
value='\$reg[codigotincidentecomun]\>\$reg[descripcionincidentecomun]</opt
ion>";
}
echo"</select>";

```

Una vez solicitado el incidente, se envía la consulta, para lo cual se debe obtener el valor de los objetos html que tienen los parámetros de búsqueda para el select:

```
$cedula=$_SESSION['cedula'];
$codigo=$_POST['incidenteredes'];
echo "<center>";
echo "<h2> Incidentes de Redes </h2>";
echo "<table class='tablesc'>";
echo "<form name='fsolucionred' id='fsolucionred' method='post'
action='grabarincidente.php'>";
echo "<tr>";
echo "<td>";
echo "<label> Solucion: </label>";
echo "</td>";
echo "<td>";
$solucion=mysql_query("SELECT codigotincidentecomun,
soluciontincidentecomun from tincidentecomun
where codigotincidentecomun='$codigo", $conexion)
or die ("Problemas en el select de la forma:".mysql_error());
while ($reg=mysql_fetch_array($solucion))
{
echo "<input type='hidden' name='codincidente' id='codincidente'
value=\"\$reg[codigotincidentecomun]\"/>";
echo "<textarea cols='50' rows='5' readonly
name='txtsolucion'>\"\$reg[soluciontincidentecomun]\"</textarea>";
}
echo "</tr>";
echo"<tr>";
echo"<td>";
echo "<input type='submit' value='GRABAR'></input>";
echo"</td>";
```

```
echo "<td>";
echo"<input type=button value='Volver al formulario'
onclick=javascript:history.back()>";
echo "</td>";
echo"</tr>";
echo "</form>";
echo "</table>";
echo "</center>";
```

3.4 Código empleado para el envío de solicitud de soporte

Dentro de la pantalla de solicitud de soporte luego de que el usuario pulsa el botón solicitar se llama a código php, el mismo que verifica a quién debe enviar la petición, así como llenar la tabla requerimiento y enviar el respectivo correo electrónico:

Variables a utilizar

```
$fechahora = date("Y-m-d H:i:s");
$cedula=$_SESSION['cedula'];
$problema=$_REQUEST['txtproblema'];
$cantidad=0;
$cedsoporte=0;
```

Sentencia que escoge el personal de soporte designado

```
$soporte=mysql_query("SELECT cantidadrequerimientos,
cedulatsopORTE,
nombretsopORTE,
```

```

emailtsoporte
FROM tsoporte
where estadotsoorte='DISPONIBLE'
and tnivelsoporte_codigotnivelsoorte=1
ORDER BY rand() LIMIT 1",$conexion)
or die ("Problemas en el select de la forma:".mysql_error());
while ($regsoporte=mysql_fetch_array($soporte))
{
$email=$regsoporte['emailtsoporte'];
$cedsoporte=$regsoporte['cedulatsoporte'];
$nomsoporte=$regsoporte['nombretsoporte'];
$cantidad=$regsoporte['cantidadrequerimientos'];
$cantidad++;
}

```

Código que actualiza la cantidad de requerimientos, inserta el requerimiento y envía el correo

Dependiendo del número de requerimientos que tenga el personal de soporte se irá incrementando el mismo y si llega a 4 pues pasará a estado de "NO DISPONIBLE", hasta que se libere de un requerimiento

```

//si es que la cantidad es igual a 4 le actualiza a estado NO DISPONIBLE al
personal de soporte, si es que es menor a 4 solamente no se encuentra el código
de actualización de estado el resto es el mismo
if (($cantidad==4) and ($cedsoporte!=0)){
$estado='NO DISPONIBLE';
mysql_query("update tsoporte
set cantidadrequerimientos='$cantidad',
estadotsoorte='$estado'
where cedulatsoporte='$cedsoporte'", $conexion)
or die ("Problemas en el update de la forma:".mysql_error());

```



```

//inserta el nuevo requerimiento
mysql_query("INSERT INTO trequerimiento (
    tsoporte_cedulatsoporte,
    tpersona_cedulatpersona,
    fechahoratrequerimiento,
    descripciontrequerimiento)
VALUES      ('$cedsoporte',
            '$cedula',
            '$fechahora',
            '$problema')", $conexion)
or die ("Problemas en el insert de trequerimiento: ".mysql_error());

//envio de mail
require("class.phpmailer.php");
$mail = new PHPMailer();
$mail->Host = "localhost";
$mail->From = $mailusua;//"pietro_e_20@yahoo.com";
$mail->FromName = $nomusua;//"Pedro Astudillo";
$mail->Subject = "Solicitud de Soporte";
$mail->AddAddress($email);
$body = "Se ha generado el requerimiento de soporte numero :\n<br>";
$body .= $requerimiento. "\n<br>";
$body .= "Solicitado por ".$nomusua. ", favor comunicarse a la extension:
".$ext. "\n<br>";
$body .= $problema. "\n<br>";
$body .= "<font style='font-style:bold'> Dar seguimiento inmediato</font>\n<br>";
$body .= "<font color='red'>Gracias por su colaboracion</font>";
$mail->Body = $body;
$mail->AltBody = "Hola amigo\nprobanda PHPMailer\n\nSaludos";
if(!$mail->Send()) {
    echo "Error al enviar correo: " . $mail->ErrorInfo;
}

```

```
}
```

3.5 Verificación de Requerimientos

El personal de soporte en el sistema estará esperando que le llegue la solicitud del requerimiento, para lo cual se empleó el siguiente código:

```
$sql = mysql_query("select numerotrequerimiento,
nombretpersona,
fechahoratrequerimiento,
descripcionrequerimiento,
extensiontdepartamento
from `helpdesk`.`trequerimiento`,
`helpdesk`.`tpersona`,
`helpdesk`.`tcargo`,
`helpdesk`.`tdepartamento`
where tpersona_cedulatpersona=cedulatpersona
and tcargo_codigotcargo=codigotcargo
and tdepartamento_codigotdepartamento=codigotdepartamento
and ifnull(estadotrequerimiento,0)!='FINALIZADO'
and tsoporte_cedulatsوپorte='$cedsoporte','$conexion)
or die ("Problemas en el select de la forma:".mysql_error());
echo"<center>";
echo "<h3> Requerimientos de Soporte </h3>";
echo "<table class='grid'>";
echo "<tr>";
echo "<td>";
echo "<label> # Requ </label>";
echo "</td>";
echo "<td>";
```

```

echo "<label> Usuario </label>";
echo "</td>";
echo "<td>";
echo "<label> Fecha </label>";
echo "</td>";
echo "<td>";
echo "<label> Incidente </label>";
echo "</td>";
echo "<td>";
echo "<label> Extension </label>";
echo "</td>";
echo "<td>";
echo "<label> Llenar </label>";
echo "</td>";
echo "</tr>";
while($row = mysql_fetch_array($sql)){
echo "<tr>";
echo "<td>";
echo $row['numerotrequerimiento'];
echo "</td>";
echo "<td>";
echo $row['nombretpersona'];
echo "</td>";
echo "<td>";
echo $row['fechahoratrequerimiento'];
echo "</td>";
echo "<td>";
echo $row['descripcionrequerimiento'];
echo "</td>";
echo "<td>";
echo $row['extensiontdepartamento'];
echo "</td>";

```

```

echo "<td>";
echo "<a href='llenabitacora.php?id=$row[numerotrequerimiento]'
style='icon:url(../imagenes/modificar.jpg)'>Ver</a>";
echo "</td>";
echo "</tr>";
}
echo"</table>";
echo "</center>";

```

3.6 Grabación de Bitácora

Después de que el personal de soporte haya o no podido resolver el inconveniente, y por ende haber o no escalado el incidente, debe grabar la bitácora, para lo cual se utilizó el siguiente código:

Variables a utilizar:

```

$numreq=$_POST['txtnumreq'];
$estado=$_POST['sestado'];
$finicio=$_POST['dthorainicio'];
$ffin=$_POST['dthorafin'];
$tipoincidente=$_POST['stipoincidente'];
$solucion=$_POST['txtsolucion'];
$escalamiento=$_POST['sescalamiento'];
$cedpersona=$_POST['txtcedpersona'];
$incidente=$_POST['txtdescripcion'];
$fechahora=$_POST['txtfecrequ'];

```

Si es que existió escalamiento, actualizará el requerimiento, colocando que fue escalado y creará un nuevo requerimiento utilizando el mismo código empleado

para la inserción de requerimiento explicado anteriormente, pero con la diferencia que al personal de soporte que designará el requerimiento será de un nivel superior; de no ser así actualizará el requerimiento colocando su solución y demás campos requeridos; así como la notificación vía correo electrónico de que el requerimiento fue atendido:

```
$bitacora=mysql_query("update trequerimiento
set estadotrequerimiento='$estado',
fechahorainicio='$finicio',
fechahorafin='$ffin',
ttipoincidentecomun_codttipoincidentecomun='$tipoincidente',
soluciontrequerimiento='$solucion',
escalamientotrequerimiento='$escalamiento'
where numerotrequerimiento='$numreq','$conexion)
or die ("Problemas en el select de la forma:".mysql_error());

if ($estado=='FINALIZADO'){
$soporte=mysql_query("update tsoporte
set cantidadrequerimientos=(cantidadrequerimientos-1),
estadotsoorte='DISPONIBLE'
where cedulatsoporte='$cedsoporte','$conexion)
or die ("Problemas en el select de la forma:".mysql_error());
}
```

```
echo"<center> <h1> Requerimiento Actualizado Normalmente</h1> </center>";
$email=mysql_query("select emailtpersona from tpersona where
cedulatpersona='$cedpersona','$conexion)
or die ("Problemas en el select de la forma:".mysql_error());
require("class.phpmailer.php");
$mail = new PHPMailer();
$mail->Host = "localhost";
$mail->From = "pietro_e_20@yahoo.com";
```

```

$mail->FromName = "Pedro Astudillo";
$mail->Subject = "Mensaje de Prueba";
$body = "Su requerimiento ha sido atendido se encuentra :\n\n<br>";
$body .= $estado."<br>";
$body .= "<font color='red'>Gracias por haber solicitado soporte</font>\n\n<br>";
$mail->Body = $body;
$mail->AltBody = "Hola amigo\n\nprobando PHPMailer\n\nSaludos";
while ($regmail=mysql_fetch_array($email))
{
$mail->AddAddress($regmail[0]);
if(!$mail->Send()){
echo "Error: " . $mail->ErrorInfo;
}
else{
echo "<center> Correo de estado de requerimiento enviado </center>";
$mail->ClearAddresses();
}
}
}

```

CAPITULO 4 INSTALACION DEL SISTEMA Y PRUEBAS

4.1 Instalación del servidor WAMPSEVER

Wampserver, es un servidor que contiene apache que es necesario para la programación en php, así como también contiene la base de datos MySQL, mismo que fue instalado en un servidor físico para poder crear la base de datos que vamos a utilizar.

Hacer doble clic en el instalador de Wampserver




Nombre	Fecha de modifica...	Tipo	Tamaño
 WampServer2.2a-x64.exe	2011-10-07 22:47	Aplicación	25.069 KB

Figura 4.1 Instalador Wampserver

Configurar la instalación de los archivos del servidor Wampserver con la ayuda del asistente de instalación y continuamos hasta finalizar.

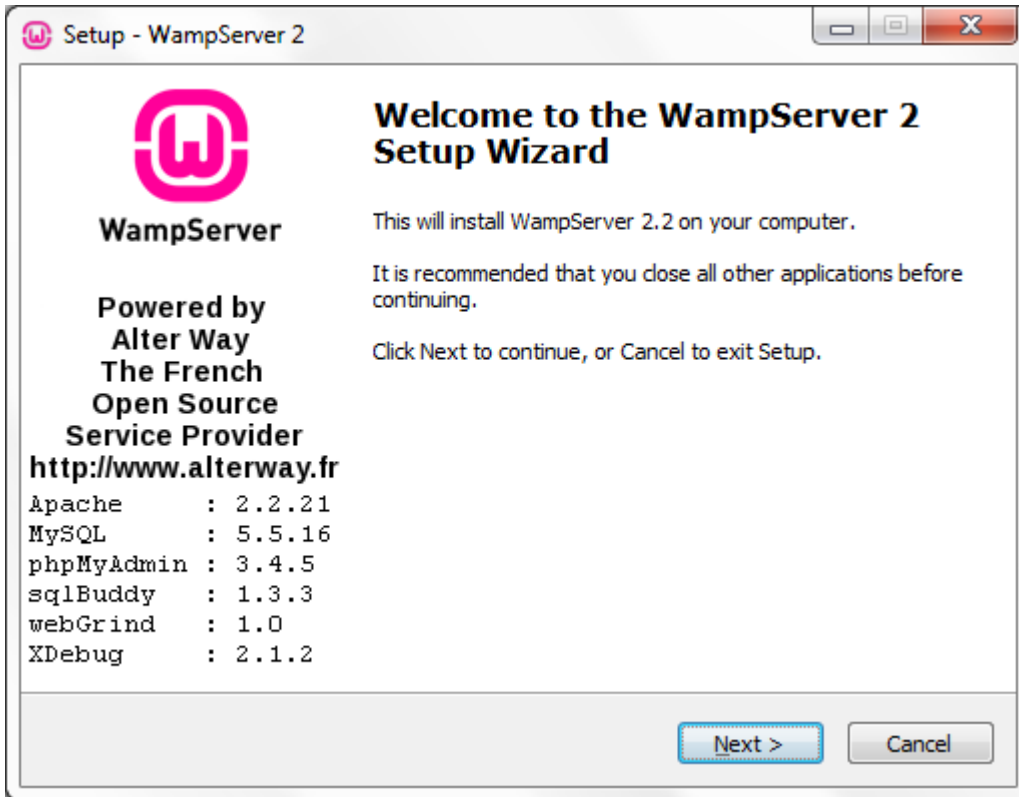


Figura 4.2 Asistente de instalación de Wampserver

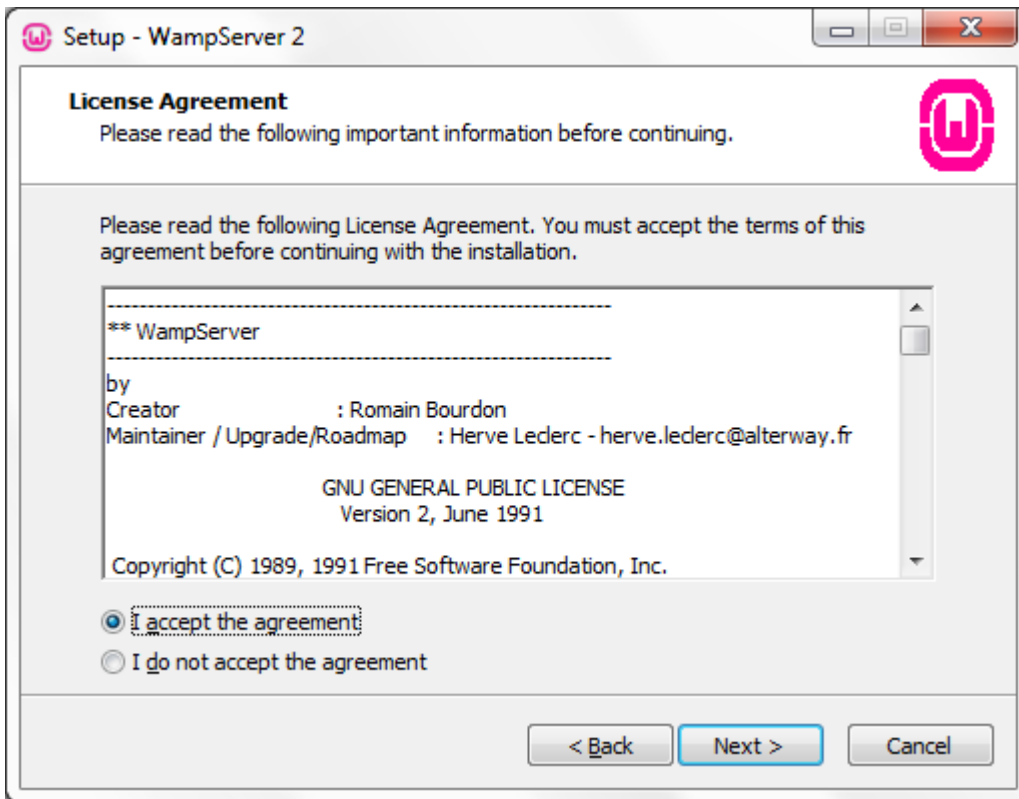


Figura 4.3 Aceptación de Licencia Wampserver

4.2 Instalación del sistema

El sistema creado en entorno web no se debe instalar porque no es un archivo ejecutable, si no todos los directorios que contienen los archivos HTML, php, imágenes empleadas, js, css, deben ser colocados en el directorio www en donde se instaló el servidor Wampserver, que por lo general es el disco C y al cual se accederán por medio de una dirección web, misma que será asignada al servidor físico del sistema web.

C:\wamp\www\helpdesk

4.3 Pruebas del sistema

Para realizar las pruebas del sistema se me concedió por unos dos días acceso a un servidor pequeño de pruebas de la Cooperativa JEP, mismo que se encontraba en la red interna.

Se pidió a personal de soporte y usuarios normales que accedan al sistema con sus usuarios y claves otorgadas para que soliciten el soporte, accedan a la ayuda de incidentes comunes, llenen la bitácora y a su vez accedan a las consultas.

El personal de soporte rápidamente accedió a los requerimientos, llenó la bitácora, y en los casos que se ejemplificó hicieron escalamiento.

Así también los usuarios pudieron calificar el soporte recibido, y a quien se le dio el rol de director se le pidió que acceda a las consultas y reportes, quedando satisfecho de los resultados.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones.

La automatización de soporte recibido por parte de usuarios en empresas grandes, debería ser una prioridad, ya que de esta manera se puede agilizar el trabajo de los empleados y ayudar a los mismos a que puedan por sí solos ir conociendo las soluciones a problemas simples informáticos que se les presenta en el día a día.

El acceso a la Base de Conocimiento por parte del personal de soporte hará que cada vez, el mismo personal nombrado tenga ya conocimiento de la solución al incidente suscitado y sea mucho más rápida su corrección.

El llenado de la Bitácora de requerimientos solicitados por parte de empleados, hace que cada vez la Base de Conocimiento se extienda más hasta tener suficiente información y varias vías para la corrección de incidentes; así como la determinación de que tan repetitivo sea un incidente para registrarlo dentro de los comunes.

El uso de la herramienta Adobe Dreamweaver CS5 fue un factor determinante para el desarrollo del sistema, ya que facilita enormemente la elaboración de páginas web, con plantillas, mermando la repetición de código html en las diferentes páginas, limitándose uno a sólo programar lo necesario una vez creada la plantilla.

MySQL es el gestor de base de datos más utilizado para la programación web, a más de eso tiene una gran ventaja de que es gratis, y no se necesita más si no saber utilizar la correcta sintaxis para la programación en php.

Recomendaciones.

El manejo de incidentes es sólo una parte de la metodología ITIL, se podría ahondar mucho más la investigación y manejar también los problemas más graves que requieren un mayor tiempo de espera, así como las configuraciones y cambios que se deba dar en caso de ser necesario; todo esto registrándolo en el sistema y haciendo que sea más completo.

Se recomienda a la institución que cambie su sistema de helpdesk, ya que está mal estructurada su base de datos y además, no se asigna un incidente a su respectivo personal de soporte según sea el caso si no se aplican las llamadas por parte de los empleados convirtiéndose a su vez en un cuello de botella.

Se recomienda utilizar las variables de sesión ya que de esa manera podemos saber qué usuario accedió al sistema, a qué hora y que no puedan acceder a una página sin los respectivos privilegios y sin haber iniciado sesión en el sistema.

BIBLIOGRAFIA

Libros.

- MAINT CIA. LTDA. HIDALGO M., Ing. Jorge E Pinilla Biblioteca de Infraestructura de Tecnologías de la Información 06/12/2006.
- SCHMULLER, JOSEPH, Aprendiendo UML en 24 horas, Pearson Latinoamérica, México, año de publicación 2002.
- KROENKE, DAVID M, Procesamiento de bases de datos: fundamentos, diseño e implementación/ Pearson Educación. México. 8 ed. 2003.
- LOPEZ QUIJADO, JOSE, Domine PHP y MYSQL. 2da Edición, Ra-Ma Editorial, año de publicación 2010.
- HEURTEL, O., PHP 5.3 DESARROLLAR UN SITIO WEB DINAMICO E INTERACTIVO, ENI EDICIONES, año de publicación 2010.

Referencia electrónica.

- OSIATIS. (s.f.). *ITIL-Gestión de Servicios TI*. Recuperado el 7 de Noviembre de 2011, de http://itil.osiatis.es/Curso_ITIL/.
- PHP. (23 de 11 de 2011). *mysql_fetch_array*. Recuperado el 23 de 11 de 2011, de *mysql_fetch_array*: <http://uk3.php.net/manual/es/function.mysql-fetch-array.php>
- PHP. (23 de 11 de 2011). *mysql_fetch_array*. Recuperado el 23 de 11 de 2011, de *mysql_fetch_array*: <http://uk3.php.net/manual/es/function.mysql-fetch-array.php>
- PHP. (20 de 12 de 2011). *mysql_fetch_row*. Recuperado el 20 de 01 de 2012, de *mysql_fetch_row*: <http://uk3.php.net/manual/es/function.mysql-fetch-row.php>
- MySQL. (12 de 11 de 2011). *Documentation*. Recuperado el 03 de 01 de 2012, de *Documentation*: <http://dev.mysql.com/doc/refman/5.0/en/limit-optimization.html>
- MySQL. (17 de 01 de 2012). *Documentation*. Recuperado el 17 de 01 de 2012, de *Documentation*: <http://dev.mysql.com/doc/refman/5.0/es/datetime.html>