

Universidad de las Ciencias Informáticas

Facultad 1



***Título:** Sistema Experto para la gestión de la Base de Conocimientos de NovaDesk,*

Trabajo de Diploma para optar por el Título de Ingeniero en Ciencias Informáticas.

Autores:

Susana Sánchez Ortiz

Alfredo Pérez Benitez

Tutores:

Ing. Yaima Oval Riverón

Ing. Santiago Almira Ramírez

Ciudad de La Habana

Mayo 2011

Declaración de autoría

Declaramos ser autores de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los 25 días del mes de mayo del año 2011

Susana Sánchez Ortiz

Firma del Autor

Alfredo Pérez Benitez

Firma del Autor

Ing. Yaima Oval Riverón

Firma del Tutor

Ing. Santiago Almira Ramírez

Firma del Tutor

Resumen

En la actualidad se cuenta con sistemas inteligentes que permiten un mejor control de cualquier proceso de la vida diaria. Estos sistemas han surgido por el avance de una ciencia conocida como Inteligencia Artificial. Una de las ramas de esta ciencia es los Sistemas Expertos.

La aplicación Web NovaDesk para el registro y control de las incidencias de Nova presta servicios de utilidad que fueron mejorados en su versión 2.0, sin embargo necesita gestionar su Base de Conocimientos para lograr una mejor atención a los usuarios.

En el presente trabajo se realiza un estudio de los Sistemas Expertos, sus características, razones de uso, clasificaciones, componentes, así como, lenguajes y herramientas para su desarrollo. Como resultado se implementa un Sistema Experto basado en reglas que permite identificar y diagnosticar los problemas con los que se enfrenta un usuario en su trabajo con el sistema Operativo Nova, mediante la gestión de la base de conocimientos de NovaDesk.

En el trabajo se realiza además un análisis del comportamiento del Sistema Experto de NovaDesk mediante su verificación, validación y evaluación; comprobándose el correcto funcionamiento del Sistema Experto implementado.

Palabras claves: Base de Conocimientos, Inteligencia Artificial, NovaDesk, Sistema Experto.

ÍNDICE

ÍNDICE.....	4
INTRODUCCIÓN	5
CAPÍTULO 1: “LOS SISTEMAS EXPERTOS”	9
1.1 GESTIÓN DE LA BASE DE CONOCIMIENTOS.	9
1.2 INTELIGENCIA ARTIFICIAL.....	10
1.3 SISTEMAS EXPERTOS	10
1.4 CARACTERÍSTICAS DE LOS SISTEMAS EXPERTOS.....	11
1.5 RAZONES PARA EL USO DE LOS SISTEMAS EXPERTOS.....	11
1.6 COMPONENTES DE UN SISTEMA EXPERTO	12
1.7 DESARROLLO DE UN SISTEMA EXPERTO	13
1.8 TIPOS DE SISTEMAS EXPERTOS.....	14
1.8.1 <i>Sistemas Basados en Probabilidad</i>	15
1.8.2 <i>Sistemas Basados en Casos</i>	15
1.8.3 <i>Sistemas Basados en Reglas</i>	16
1.8.4 <i>Comparación entre los Sistemas basados en Reglas y los Sistemas basados en Probabilidad</i>	17
1.9 LOS SISTEMAS EXPERTOS EN EL MUNDO	18
1.10 LOS SISTEMAS EXPERTOS EN CUBA	18
1.11 LOS SISTEMAS EXPERTOS EN LA UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS (UCI).....	19
1.12 LENGUAJES UTILIZADOS EN LA CONSTRUCCIÓN DE SISTEMAS EXPERTOS	19
1.13 HERRAMIENTAS Y SHELLS UTILIZADOS EN LA CONSTRUCCIÓN DE SISTEMAS EXPERTOS	21
1.14 OTRAS HERRAMIENTAS Y LENGUAJES A EMPLEAR.	23
1.14.1 <i>Lenguajes de programación Web</i>	24
1.14.2 <i>Herramientas para el desarrollo Web</i>	24
1.14.3 <i>Metodología para el desarrollo del software</i>	25
CAPÍTULO 2: “SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTOS DE NOVADESK”	26
2.1 SISTEMA EXPERTO PROPUESTO.....	26
2.2 PROCESO DE DESARROLLO DEL PROYECTO BASADO EN SXP	30
2.3 MODELO DE DOMINIO.....	31
2.4 LISTA DE RESERVA DEL PRODUCTO (LRP).....	31
2.5 HISTORIAS DE USUARIO.	32
2.6 PLAN DE RELEASE.	48
2.7 BASE DE DATOS.....	48
2.8 DISEÑO CON METÁFORAS.....	49
2.9 DIAGRAMA DE COMPONENTES.	51
CAPÍTULO 3: “DISEÑO Y REALIZACIÓN DE PRUEBAS”	52
3.1 CASOS DE PRUEBA	52
CAPÍTULO 4: “ANÁLISIS DEL COMPORTAMIENTO DEL SISTEMA EXPERTO”	61
4.1. VERIFICACIÓN DEL SISTEMA EXPERTO	61
4.1.1 <i>Verificación del cumplimiento de las especificaciones</i>	61
4.1.2 <i>Verificación de los mecanismos de inferencia</i>	62
4.1.3 <i>Verificación de la base de conocimientos</i>	62
4.2. VALIDACIÓN DEL SISTEMA EXPERTO.....	63
4.2.1 <i>Principales características del proceso de validación</i>	64
4.2.2 <i>Validación del Sistema Experto implementado</i>	69
4.3 EVALUACIÓN DEL SISTEMA EXPERTO.....	71
CONCLUSIONES GENERALES.....	73
RECOMENDACIONES	74
REFERENCIAS BIBLIOGRÁFICAS	75
BIBLIOGRAFÍA CONSULTADA	77
ANEXOS.....	79
GLOSARIO DE TÉRMINOS.....	80

Introducción

El siglo XXI ha provocado un auge en el desarrollo de las Tecnologías de la Información y las Comunicaciones (TIC), cada día la tecnología se encuentra en constante transformación. En la actualidad se cuenta no solo con programas que contribuyen con la automatización de funciones para mejorar la vida de la humanidad, sino con sistemas inteligentes que sustituyen las funciones de un humano; programas que reúnen características y comportamiento similares al de la inteligencia humana, capaces de alcanzar metas complejas en entornos complejos. Estos sistemas permiten un mayor control de cualquier proceso de la vida diaria: actividades que solo eran posible realizar por humanos se están desarrollando por un software.

La Inteligencia Artificial (IA) se dedica a la creación de máquinas que puedan "pensar". Uno de los mayores éxitos alcanzados por la inteligencia artificial han sido los Sistemas Expertos basados en reglas heurísticas. Estos surgen en los años 70 con el objetivo de incorporar en los sistemas informáticos las "reglas de experiencia" y de "buen juicio". Estos sistemas en ningún momento sustituyen o reemplazan la actividad del profesional.

Cuba enfrenta el reto de informatizar su sociedad con el propósito de integrarse plenamente a la infraestructura de la información y hacer uso óptimo de las nuevas tecnologías, lo que permitirá lograr incrementos sustanciales en la productividad, calidad y la eficiencia en toda la actividad tanto industrial como de servicios. El crecimiento de las comunicaciones con las nuevas tecnologías de la información, no permite que ninguna esfera económica o social pueda desarrollarse si no es con la presencia de la informática; por ello la nación ha incursionado en el mundo del desarrollo de software, sin menguar ante los modelos privativos que se imponen a nivel mundial. Se ha lanzado a romper todas las barreras que pudieran parar la estrategia de informatización de la sociedad cubana aprobada en 1997 [1] y en la cual se trabaja arduamente.

El proceso de migración al software libre se hace cada vez más cotidiano y va alcanzando mayor popularidad a nivel mundial, pues constituye una alternativa para el desarrollo de la informática, tanto en el marco público como en el privado. La migración al software libre es una vía factible para Cuba por ser un país bloqueado y del tercer mundo, lo que constituye una alternativa para superarse económicamente.

La Universidad de las Ciencias Informáticas (UCI) constituye el pilar fundamental del avance tecnológico en la Isla de Cuba, en la cual se han desarrollado diversos software para la informatización de los procesos de la sociedad cubana y es la protagonista del proceso de migración que se lleva a cabo en Cuba. Entre las facultades que conforman la UCI, la Facultad 1 con sus proyectos de software libre asume un rol importante en el proceso de migración; entre estos proyectos cuenta con Nova, que se enfoca en el desarrollo de las diferentes partes del sistema operativo del mismo nombre. El sistema operativo Nova es una distribución de GNU/Linux, desarrollada para entorno de escritorio orientado a usuarios inexpertos que hayan experimentado un proceso de migración de Microsoft Windows a

GNU/Linux, listo para hacerse extensivo a cada ordenador donde sea necesario un sistema operativo libre cubano.

Con el objetivo de gestionar los reportes de las posibles incidencias por parte de los usuarios de Nova, se plantea en el trabajo de diploma titulado “Aplicación Web para brindar soporte a Nova: NovaDesk” [2] la necesidad de contar con un sistema que brinde asistencia de soporte técnico, capaz de ofrecer el servicio mediante las nuevas tecnologías sin la necesidad de realizarlo personalmente, dando lugar a la aplicación Web NovaDesk 1.0. Esta aplicación se obtuvo posterior al estudio del trabajo de diploma “Servicio de Soporte Técnico utilizando la tecnología Service-Desk” [3] donde se plantea la necesidad de emplear una tecnología Web como los Service-Desk, para mejorar la calidad del servicio de soporte técnico.

Con el objetivo de incorporarle nuevas funcionalidades a NovaDesk 1.0, de solucionar todos los problemas presentes en la estrategia organizativa al realizar la atención a los clientes y de controlar los componentes de la infraestructura donde se encuentre instalado el sistema operativo Nova; se plantea en el trabajo de diploma titulado “Herramienta para el Soporte de los Servicios NovaDesk 2.0” [4] la necesidad de alcanzar mayor nivel de prestación de servicio en la aplicación Web NovaDesk dando lugar a NovaDesk 2.0 que es ahora una personalización de GLPI (*Gestionnaire libre de parc informatique*).

Las funcionalidades primarias de NovaDesk 2.0 van a estar enmarcadas en la gestión de incidencias, posibilitando su control a través de reportes realizados por los usuarios y del empleo de la mensajería instantánea. Brinda consultoría sobre la base de conocimientos donde se encuentran preguntas frecuentes. Posee notificación vía correo de la información de un determinado reporte, define estadísticas y tiempo de trabajo de un reporte al administrador. NovaDesk 2.0 cuenta con una sala de chat para que los técnicos compartan sus conocimientos y aclaren sus dudas entre sí; además, presenta funcionalidades que permiten el inventario de hardware y la publicación de noticias relacionadas al tema de soporte, entre otras necesarias para brindar una buena atención y soporte al sistema operativo Nova.

Sin embargo, aunque el sistema presta servicios de utilidad que fueron mejorados en su versión 2.0 aún presenta deficiencias en la estrategia organizativa al realizar la atención a los clientes. Los técnicos responsables de atender las incidencias son estudiantes que prestan el servicio en los horarios de tiempo de máquina o en momentos extras en los que se encuentren conectados a la aplicación, provocando retardo en la solución a las mismas. Dichos estudiantes no poseen, en ocasiones, el conocimiento, ni tienen la experiencia necesaria para dar la atención y respuesta que requiere un usuario. La comunicación de forma escrita no satisface las expectativas para lograr una buena comunicación: frecuentemente los usuarios no saben exponer el problema que enfrentan y los técnicos no saben cómo dirigirse ante el desconocimiento del mismo.

Estos problemas planteados anteriormente se ven arraigados pues los usuarios en su mayoría no

consultan la base de conocimientos de NovaDesk por no tener costumbre de buscar la ayuda que necesitan; además, los técnicos al no tener el conocimiento ni la experiencia suficiente, se ven en la necesidad de recurrir a esta y como la información con la que cuenta la base de conocimientos es amplia puede ser engorrosa su búsqueda, lo que trae como consecuencia un retardo a la hora de dar respuesta a las incidencias de los usuarios.

En encuestas realizadas a estudiantes de los primeros años de enseñanza: primero, segundo y tercer año, se obtuvo como resultado que los mismos se sienten capaces de resolver por sí solos muchos de los problemas que presentan durante su trabajo con el sistema operativo Nova, pero esto lo realizarían con más frecuencia si encontraran fácilmente la guía para hacerlo; además, esta sería una forma para que los técnicos de NovaDesk se preparen más en la solución de problemas complejos que no puedan resolver los usuarios por sí solos.

Los problemas declarados anteriormente ocasionan que los usuarios no recurran de forma frecuente a la aplicación para registrar sus incidencias, y que el sistema no cumpla con su principal objetivo: prestar servicio de soporte.

Lo anterior demuestra una contradicción entre el estado real y el estado deseado, de donde surge el siguiente **problema científico**: ¿Cómo realizar la gestión de la Base de Conocimiento de NovaDesk que permita identificar y diagnosticar problemas?

Por lo que el **objeto de estudio** del trabajo son los Sistemas Expertos, enmarcado en el **campo de acción**: los Sistemas Expertos para la gestión de la base de conocimientos.

En correspondencia con el problema planteado se propone como **objetivo general**: Desarrollar un Sistema Experto de diagnóstico e identificación de problemas para NovaDesk. Derivándose los siguientes **objetivos específicos**:

- Sistematizar los aspectos teóricos relacionados con los Sistemas Expertos.
- Proponer un Sistema Experto que posibilite la gestión de la base de conocimientos de NovaDesk.
- Implementar la solución propuesta.
- Realizar las pruebas correspondientes al Sistema Experto implementado.
- Analizar el comportamiento del Sistema Experto implementado.

Para darle cumplimiento a los objetivos que se exponen anteriormente se hace necesario llevar a cabo el desarrollo de las siguientes tareas:

1. Fundamentación teórica relacionada con los Sistemas Expertos.
2. Análisis y diseño de la propuesta de un Sistema Experto para la gestión de la base de conocimientos de NovaDesk.
3. Implementación del Sistema Experto.
4. Realización de las pruebas correspondientes al Sistema Experto implementado.

5. Verificación, validación y evaluación del Sistema Experto Implementado

Con la realización del presente trabajo investigativo, se **defiende la idea** que con la implementación de un Sistema Experto se permitirá realizar la gestión de la Base de Conocimientos de NovaDesk, e identificar y diagnosticar problemas.

En el transcurso de la investigación se hace uso de varios métodos y técnicas de investigación siendo de gran importancia para el desarrollo del trabajo. Los métodos que se utilizan son: el **Analítico Sintético** pues permitió el estudio de diferentes fuentes bibliográficas para extraer los elementos más importantes que se relacionan con los Sistemas Expertos. Se realizaron además resúmenes y valoraciones de conceptos relevantes relacionados con este tema y con la gestión de la base de conocimientos. También se utilizó el **Análisis Histórico – Lógico** para determinar los antecedentes históricos relacionados con los Sistemas Expertos en toda su trayectoria y en este caso los relacionados con la gestión de bases de conocimientos.

Para el desarrollo del trabajo se utilizaron además técnicas como **la encuesta** a estudiantes de los primeros años de la carrera de Ingeniería Informática por la sistematicidad en el trabajo con el sistema operativo Nova. Se aplicó para diagnosticar el estado actual del uso del sistema operativo Nova y obtener información acerca de los problemas que enfrentan los usuarios durante el trabajo con el mismo, entre otros aspectos [anexo 1]. Otra técnica utilizada fue **la entrevista** que se les realizó a los expertos en Nova para el desarrollo de la investigación, para la elaboración de las reglas del Sistema Experto y para añadir a la base de conocimientos de NovaDesk más información [anexo 2].

La investigación está desarrollada en 4 capítulos, cuya estructura se describe a continuación:

En el primer capítulo “**Los Sistemas Expertos**” se realiza el estudio del estado del arte donde se hace referencia a los aspectos relacionados con los Sistemas Expertos, como concepto, características, componentes, clasificación, entre otros. Además se abordan conceptos relacionados con la gestión de la base de conocimientos y la Inteligencia Artificial; se mencionan los lenguajes y herramientas orientados al desarrollo de los Sistemas Expertos, destacándose las más adecuadas a emplear.

En el segundo capítulo “**Sistema Experto para la gestión de la Base de Conocimientos de NovaDesk**” se hace referencia a la solución que se propone, planteándose cómo se va a desarrollar esta; se definen cuáles serán las funcionalidades a implementar en el Sistema Experto. Además se describe el proceso ágil basado en las historias de usuario, correspondiente a las nuevas funcionalidades, prototipo de interfaz de usuario, plan de revisión, entre otros artefactos de interés.

En el tercer capítulo “**Diseño y realización de pruebas**” se describen, diseñan, realizan y controlan los casos de pruebas aplicadas al sistema.

En el cuarto capítulo “**Análisis del comportamiento del Sistema Experto**” se evalúa globalmente el comportamiento del Sistema Experto implementado mediante las fases de este proceso: Verificación, Validación y Evaluación.

Capítulo 1: “Los Sistemas Expertos”

1.1 Gestión de la base de conocimientos.

Para la realización del trabajo de diploma es necesario conocer los conceptos relacionados con base de conocimientos y gestión del conocimiento: son dos elementos que guiarán el desarrollo del Sistema Experto, por ser un sistema basado en el conocimiento.

➤ **Base de conocimientos**

El concepto de base de conocimientos tiene muchos significados enmarcados fundamentalmente en el área de las empresas y las organizaciones. Sin embargo resumiendo de la bibliografía consultada acerca del tema: una *Base de Conocimientos (Knowledge Base (KB))* recoge toda la información necesaria para ofrecer una primera línea de soporte de una manera inmediata y eficaz, sin necesidad de recurrir a escalados. Es necesaria para realizar la tarea comercial y de soporte al negocio. [5]

Una base de conocimientos va a englobar información de uno o varios temas relacionados con la organización donde se tenga, es como una fuente de conocimientos a la que se puede recurrir ante cualquier duda acerca de un tema.

➤ **Gestión del conocimiento**

La Gestión del Conocimiento, según *Jeff Angus*, se relaciona con las Tecnologías de Información y el Conocimiento de una manera fundamental: "...la gestión de conocimiento es el concepto bajo el cual la información se convierte en conocimiento activo... Los productos (de la informática) son los que facilitan la gestión de los conocimientos -o al menos, algunas de sus facetas específicas- a partir del uso apropiado de las tecnologías de la información"[6].

Existen, además del anterior, muchos conceptos y definiciones de gestión del conocimiento, a continuación se exponen algunos: “La gestión del conocimiento es el proceso sistemático y explícito de capturar la experiencia individual o colectiva de una organización, independiente del medio en el cual se encuentra, de tal manera de poder distribuirlo o dejarlo accesible a todas aquellas personas a las cuales le es útil. De esta manera las buenas ideas que surjan, donde sea que se encuentren, deberán ser capturadas para adecuadamente transferirlas a quienes tengan problemas análogos.” [7]

"La gestión incluye todos los procesos relacionados con la identificación, puesta en común y creación del conocimiento. Ello requiere tanto sistemas informáticos que permitan la creación y el mantenimiento de repositorios de conocimiento, como una cultura organizativa que favorezca la transmisión del conocimiento individual y el aprendizaje colectivo." [8]

En resumen, la Gestión del Conocimiento es la gestión de los activos intangibles que generan valor para la organización; esta implica manejar eficazmente los recursos tecnológicos de la organización, de manera que propicien una comunicación eficiente entre las partes que la integran, mediante los flujos de información, considerados el *"insumo vital"* para generar, transformar y transferir el conocimiento, es decir, realizar Gestión del Conocimiento.

1.2 Inteligencia Artificial

Actualmente el desarrollo de la computación y de la informática se ha tornado imprescindible para la vida humana, son necesarias en casi todas las esferas en las que el hombre se desempeña; además ha propiciado el surgimiento de nuevas ciencias, una de ellas es la Inteligencia Artificial, a partir de este momento los autores se referirán a esta como IA.

La IA se ha convertido en las últimas décadas en la más popular de todas, lo que ha propiciado un acelerado incremento de la utilidad de los sistemas de cómputo. Este término ha generado mucha polémica desde hace varias décadas, pero la mayoría de los autores coinciden en que es, en esencia, lograr que una máquina tenga inteligencia propia, es decir: “La inteligencia artificial es una de las áreas más fascinantes y con más retos de las ciencias de la computación pues ha tomado a la inteligencia como la característica universalmente aceptada para diferenciar a los humanos de otras criaturas ya sean vivas o inanimadas, para construir programas o computadoras inteligentes.” [9]

Esta es una de las definiciones más completas que ilustra este término, pero no es la única, para otros autores, la IA es el estudio de cómo hacer que los ordenadores hagan cosas que, en estos momentos, hace mejor el hombre. [10] Para otros, la IA es una ciencia que intenta la creación de programas para máquinas que imiten el comportamiento y la comprensión humana, que sea capaz de aprender, reconocer y pensar. [11]

La IA es un área de la investigación donde se combinan las computadoras, la fisiología y filosofía; esta ciencia tiene implícitas varias ramas y dentro de ellas una de las que se pueden destacar son los Sistemas Basados en el Conocimiento (SBC). Un **Sistema Basado en el Conocimiento** es aquel en el que aparece representado el conocimiento de un dominio determinado, de tal forma que dicha representación sea procesable por un programa informático. Un Sistema Basado en el Conocimiento al que se le incorpora conocimiento proveniente de expertos en un dominio se le conoce como Sistema Experto (SE). [12]

Los términos **Sistema Experto**, **Sistema Basado en Conocimiento**, o **Sistema Experto Basado en Conocimiento**, se usan como sinónimos [13]. La mayoría utiliza el término Sistema Experto porque es más corto, en el presente trabajo de diploma se utilizará el término Sistema Experto.

1.3 Sistemas Expertos

Los Sistemas Expertos son una rama de la IA que hace un uso del conocimiento especializado para resolver problemas como un especialista humano. Muchos escritores dieron su definición de Sistemas Expertos. Un Sistema Experto, es básicamente un programa de computadora basado en conocimientos y raciocinio que lleva a cabo tareas que generalmente sólo realiza un experto humano; es decir, es un programa que imita el comportamiento humano en el sentido que utiliza la información que le es proporcionada para poder dar una opinión sobre un tema en especial. Otros autores lo definen como sigue: un Sistema Experto es un programa de computadora interactivo que contiene la experiencia,

conocimiento y habilidad propios de una persona o grupos de personas especialistas en un área particular del conocimiento humano, de manera que permitan resolver problemas específicos de esa área de forma inteligente y satisfactoria. [13]

El objetivo de un Sistema Experto es igualar el comportamiento de los expertos humanos, existen muchas personas que se consideran expertos en la actualidad como por ejemplo: los abogados, economistas, médicos; todos ellos comparten una característica en común: deben tomar decisiones acertadas en ambientes rodeados de riesgos e incertidumbre pero poseen la habilidad superior de hacerlo como resultado de su entrenamiento, experiencia y práctica profesional [14].

Los expertos humanos tienen un corto tiempo de existencia, contando a partir de su capacidad para actuar como un experto, lo que le atribuye la desventaja de que su conocimiento sea un caudal enriquecido temporalmente; a diferencia de los Sistemas Expertos que se alimentan de estos pero perduran mucho más en el tiempo. [12]

1.4 Características de los Sistemas Expertos.

Los Sistemas Expertos poseen un grupo de características que los definen y permiten identificar cuándo son necesarios, es decir cuándo se pueden utilizar, estas características fueron recopiladas de varias bibliografías. Los autores consideran que las siguientes son las más importantes:

- Alto desempeño. El sistema debe tener la capacidad de responder a un nivel de competencia igual o superior al de un especialista en el campo. Esto significa que la calidad del consejo dado por el sistema debe ser muy alta.
- Tiempo de respuesta adecuado. El sistema debe actuar en un tiempo razonable, comparable o mejor al tiempo requerido por un especialista para alcanzar una decisión.
- Confiabilidad. El Sistema Experto debe ser confiable y no propenso a fallos.
- Comprensible. El sistema debe ser capaz de explicar los pasos de su razonamiento. Este rasgo es muy importante debido a que en determinados momentos se puede depender de manera directa de la decisión tomada por el sistema.
- Flexibilidad. Debido a la gran cantidad de conocimiento que puede albergar un Sistema Experto es importante contar con un mecanismo eficiente para añadir, modificar y eliminar el conocimiento.
- Representación explícita del conocimiento.

Existen además razones para usar los Sistemas Expertos en el desarrollo de aplicaciones para la solución de problemas.

1.5 Razones para el uso de los Sistemas Expertos

Los Sistemas Expertos se basan en la simulación del razonamiento humano. El uso de los Sistemas Expertos trae consigo ventajas que permiten la comprensión de los mismos y lo que estos son capaces

CAPÍTULO 1: LOS SISTEMAS EXPERTOS

de brindar a un sistema automatizado. El desarrollo o adquisición de un Sistema Experto es generalmente caro, pero el mantenimiento y el coste marginal de su uso repetido es relativamente bajo. La ganancia en términos monetarios, tiempo y precisión resultantes del uso de los Sistemas Expertos son muy altas y la amortización es muy rápida. Un Sistema Experto, además, mejora la productividad al resolver y decidir los problemas más rápidamente. Esto permite ahorrar tiempo y dinero.

Los valiosos conocimientos de un especialista se guardan y se difunden, de forma que, no se pierden aunque desaparezca el especialista. En los Sistemas Expertos se guarda la esencia de los problemas que se intenta resolver, se programa cómo aplicar los conocimientos para su resolución y ayudan a entender cómo se aplican los conocimientos para resolver un problema; esto es útil porque normalmente el especialista da por ciertos sus conocimientos y no analiza cómo los aplica. Los Sistemas Expertos tienen gran flexibilidad, lo que se traduce en una mejor modularidad y legibilidad del conocimiento. [11]

El proceso de desarrollo de un Sistema Experto también tiene un beneficio colateral, puesto que el conocimiento que se le introduce a partir de los especialistas humanos es claramente detallado y examinado. [12]

1.6 Componentes de un Sistema Experto

Los Sistemas Expertos se definen por su arquitectura, diferenciándose particularmente tres componentes principales: base de hechos, base de conocimientos y motor de inferencia.

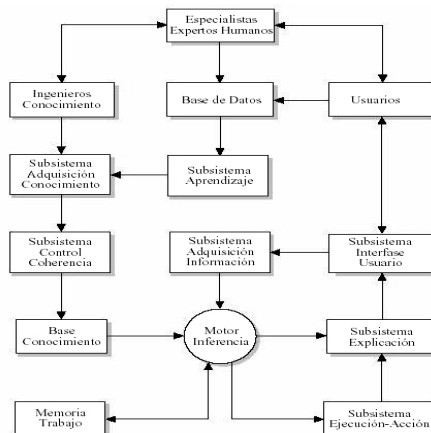


Figura 1.1 Componentes de un Sistema Experto. [15]

La componente humana es la combinación de cada uno de los participantes en el desarrollo del Sistema Experto, de forma que los expertos humanos suministran el conocimiento básico en el tema de interés, y los ingenieros del conocimiento trasladan este conocimiento a un lenguaje, que el Sistema Experto pueda entender. El sistema dispone de la llamada *base de conocimientos* que contiene el conocimiento especializado extraído del experto en el dominio, es decir, contiene conocimiento general sobre el dominio en el que se trabaja. Dentro de la base de conocimientos existen tres tipos de conocimiento para construir un Sistema Experto: reglas, hechos y relaciones entre componentes, afirmaciones y

preguntas; para representar estos tipos de conocimiento en la base de conocimientos, se utilizan tres métodos: reglas, estructuras y lógica.

El Subsistema de Adquisición de Conocimiento controla el flujo del nuevo conocimiento que fluye del experto humano a la base de datos. El sistema determina qué nuevo conocimiento se necesita o si el conocimiento recibido es en realidad nuevo. *El Control de la Coherencia* controla la consistencia de la base de datos y evita que unidades de conocimiento inconsistentes entren en la misma. Sin este subsistema, unidades de conocimientos contradictorios pueden formar parte de la base de conocimiento, dando lugar a un comportamiento insatisfactorio en el sistema. Este comprueba e informa a los expertos de las inconsistencias.

El Subsistema de adquisición de información es utilizado por el motor de inferencia cuando el conocimiento inicial es muy limitado y no se pueden sacar conclusiones, este obtiene el conocimiento necesario y continúa con el proceso de inferencia hasta que se hayan sacado conclusiones. *La interfaz de usuario* es el enlace entre el usuario y el Sistema Experto. Para que un Sistema Experto sea una herramienta efectiva, debe incorporar mecanismos eficientes para mostrar y obtener información de forma fácil y agradable. *El subsistema de ejecución de órdenes* es la componente que permite al Sistema Experto iniciar acciones; estas acciones se basan en las conclusiones sacadas por el motor de inferencia.

El subsistema de explicación le brinda al usuario una explicación de las conclusiones sacadas o de las acciones iniciadas por el Sistema Experto, por ello, es necesario un subsistema que explique el proceso seguido por el motor de inferencia o por el subsistema de ejecución. En muchos casos, la explicación es necesaria debido a los riesgos asociados con la acción a ejecutar, por ejemplo en el campo del diagnóstico médico. *El subsistema de aprendizaje* se conforma de dos tipos: aprendizaje estructural y aprendizaje paramétrico. El aprendizaje estructural se refiere a algunos aspectos relacionados con la estructura del conocimiento (reglas, distribuciones de probabilidad, entre otros). El aprendizaje paramétrico se refiere a estimar los parámetros necesarios para construir la base de conocimiento, por ejemplo la estimación de frecuencias o probabilidades asociadas a síntomas o enfermedades.

La base de hechos o memoria de trabajo, contiene los datos propios del problema que serán tratados con ayuda del sistema. Asimismo, a pesar de ser la memoria de trabajo, la base de hechos puede desempeñar el papel de memoria auxiliar. La memoria de trabajo memoriza todos los resultados intermedios, permitiendo conservar el rastro de los razonamientos llevados a cabo. El último elemento es el *motor de inferencias*, que es el programa que mediante el empleo de los conocimientos obtiene las conclusiones aplicando la lógica clásica a las reglas contenidas en la base de conocimientos. Estas conclusiones pueden estar basadas en conocimientos probabilísticos o deterministas.

1.7 Desarrollo de un Sistema Experto

Existen metodologías para el desarrollo de los Sistemas Expertos, dentro de estas metodologías se

CAPÍTULO 1: LOS SISTEMAS EXPERTOS

encuentran: la de Buchanan, la de Grover, la de Brule, la de Blanqué -García Martínez, la Weiss-Kulikowski, la CommonKADS y la IDEAL. Las metodologías planteadas anteriormente constituyen herramientas utilizadas por el ingeniero del conocimiento que le dan pautas para la realización de un Sistema Experto, además de permitir una correcta documentación y detectar problemas durante el desarrollo del mismo; todas estas metodologías proponen una serie de etapas que guían al ingeniero del conocimiento en el proceso de adquisición y formalización del conocimiento.

Según los autores Luciana Arrúa y Eduardo Meza Fernández [15], de forma general un Sistema Experto cuenta con siete etapas en su desarrollo:

- Planeamiento del problema: Es la etapa más importante en el desarrollo de un Sistema Experto, en la misma se define el problema a resolver.
- Encontrar expertos humanos: En ocasiones las bases de datos pueden jugar el papel del experto humano, del mismo modo son caros y difíciles de encontrar.
- Diseñar un Sistema Experto: Esta etapa incluye el diseño de estructuras para almacenar el conocimiento, el motor de inferencia, el subsistema de explicación, la interfaz de usuario, entre otras.
- Elegir herramienta de desarrollo: Debe decidirse qué herramientas y lenguajes de programación utilizar.
- Construir y probar prototipo: Si el prototipo no pasa las pruebas requeridas, las etapas anteriores (con las modificaciones apropiadas) deben ser repetidas hasta encontrar el prototipo satisfactorio.
- Refinamiento y generación: Etapa en la cual se corrigen los fallos y se incluyen nuevas posibilidades no incorporadas al diseño inicial.
- Mantenimiento y puesta al día: Etapa en la cual el usuario plantea los problemas o defectos del prototipo, corrige errores, actualiza el producto con nuevos avances.

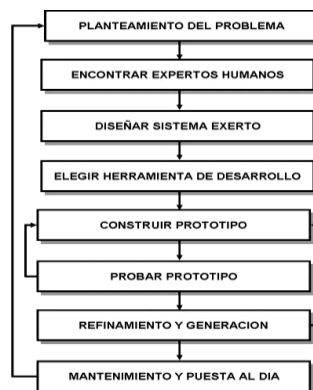


Figura 1.2 Etapas en el desarrollo de un Sistema Experto. [15]

1.8 Tipos de Sistemas Expertos

Los Sistemas Expertos se clasifican según la Naturaleza del Problema en deterministas y estocásticos.

Los Sistemas Expertos deterministas son formulados usando un conjunto de reglas que relacionan varios objetos bien definidos. Se les conoce como: “Sistemas basados en reglas” porque obtienen conclusiones basándose en un conjunto de reglas utilizando un mecanismo de razonamiento lógico, por ejemplo las transacciones bancarias y el control del tráfico. Los Sistemas Expertos estocásticos introducen algunos medios para tratar la incertidumbre en situaciones inciertas; en este aspecto, algunos Sistemas Expertos usan la estructura de los sistemas basados en reglas pero introducen una medida asociada a la incertidumbre de las reglas y a la de sus premisas. Una medida de la incertidumbre es la probabilidad en la que la distribución conjunta de un conjunto de variables se utiliza para describir las relaciones de dependencia entre ellas. A estos sistemas se les conoce como “Sistemas Expertos Probabilísticos” y la estrategia de razonamiento se conoce como “Razonamiento Probabilístico” o “Inferencia Probabilística”.

1.8.1 Sistemas Basados en Probabilidad

En los Sistemas Expertos probabilísticos las relaciones entre las variables se describen mediante su función de probabilidad conjunta. Por ello, para construir la base de conocimiento de un Sistema Experto probabilístico, se necesita definir la función de probabilidad conjunta de las variables. La estructura de los Sistemas Expertos probabilísticos es más abstracta para la lógica humana. La base de conocimientos de estos sistemas está compuesta por un espacio probabilístico y un motor de inferencia, que calcula la probabilidad de los sucesos a través de diversos métodos de cálculo aplicando varias hipótesis de independencia. Generalmente trabajan con una base de datos previamente creada con las probabilidades.

La probabilidad es, hoy en día, la más intuitiva y la más aceptada de las medidas de incertidumbre, dentro de las ventajas de los sistemas basados en probabilidad la más importante es que necesita una intervención, por parte del ingeniero de control, muy reducida, tanto en la etapa de la generación de la base de conocimiento inicial, como en la introducción de nuevos elementos. Dentro de las desventajas de este sistema está presente el alto número de parámetros que manejan, lo que hace que sea difícil su especificación y definición, además el motor de inferencia es más complicado de implementar.

1.8.2 Sistemas Basados en Casos

El razonamiento basado en casos es una rama de la IA que se preocupa por el estudio de los mecanismos mentales necesarios para repetir lo que se ha hecho o vivido con anterioridad, ya sea por uno mismo, o por casos concretos recopilados en la bibliografía o en la sabiduría popular. Un razonador basado en casos resuelve problemas nuevos mediante la adaptación de soluciones previas usadas para resolver problemas similares. [16]

El razonamiento basado en casos, es un modelo de resolución de problemas que utiliza conocimiento específico de experiencias previas. Un problema nuevo es resuelto al encontrar un caso pasado similar

y se reutiliza en la situación del problema nuevo. Un ciclo de vida de un sistema basado en casos está formado esencialmente por cuatro procesos: recuperar el caso o casos pasados más similares; reutilizar la información y conocimiento de este caso o casos recuperados para resolver el nuevo problema; revisar la solución propuesta y guardar la nueva solución una vez ha sido confirmada o validada.

Existen varios tipos de sistemas basados en casos acorde a sus dos componentes principales, y estos son: el interpretativo y el solucionador de problemas.

De los sistemas basados en casos se tienen las siguientes ventajas:

- Adquisición de conocimiento: Razonan desde episodios específicos, lo cual evita el problema de descomponer el conocimiento del dominio y generalizarlo en reglas.
- Flexibilidad en la representación del conocimiento: Pueden explotar múltiples tipos de conocimiento.
- Aumento de la eficiencia al resolver un problema: La reutilización de las soluciones previas al resolver un problema, aumenta la eficiencia.
- Aumento de la calidad de las soluciones: Las soluciones sugeridas por Casos en dominios no entendidos completamente son más precisas que aquellas que surgieron por cadenas de reglas generalizadas.
- La aceptación del usuario: Las soluciones derivadas a partir del Razonamiento Basado en Casos están fundamentadas en Casos reales. Esto le permite al sistema justificar las decisiones al usuario.

Dentro de las desventajas de estos sistemas se encuentran: añadir casos requiere modificar la red, mantener la red en forma óptima es costoso y algunos casos pueden no ser considerados dado el orden de las preguntas. También se pueden señalar otras desventajas, como pueden ser la dificultad para encontrar una estructura apropiada para describir el contenido de un caso y decidir cómo la memoria de Casos debe ser organizada e indexada para un almacenamiento, recuperación y reutilización efectiva y eficiente; además cuando el modelo del conocimiento general del dominio es incompleto y presenta información imprecisa, resulta difícil su integración a la estructura de la base de casos.

1.8.3 Sistemas Basados en Reglas.

Los sistemas basados en reglas son actualmente la elección más popular para el diseño y la construcción de Sistemas Expertos, estos son producto de un proceso de evolución que ha durado 30 años. Los Sistemas Expertos basados en reglas se definen a partir de un conjunto de objetos, que representen las variables del modelo considerado, ligadas mediante un conjunto de reglas, que representarán las relaciones entre las variables.

“Una regla es una afirmación lógica que relaciona dos o más objetos e incluye dos partes, la premisa y la conclusión. Cada una de estas partes consiste en una expresión lógica con una o más afirmaciones objeto-valor conectadas mediante los operadores lógicos “y”, “o”, o “no””. [15]

Las reglas deterministas constituyen la más sencilla de las metodologías utilizadas en Sistemas Expertos. La base de conocimiento contiene el conjunto de reglas que definen el problema y el motor de inferencia saca las conclusiones aplicando la lógica clásica a estas reglas.

Los sistemas basados en reglas tienen ventajas y desventajas cuando se comparan con otras soluciones como el software convencional o los humanos. Entre sus ventajas se tienen:

- Facilidad de expresión. Para muchos problemas es relativamente fácil expresar el conocimiento mediante cláusulas del tipo $p \rightarrow q$ lo que convierte a los sistemas basados en reglas en una alternativa atractiva para el diseño de Sistemas Expertos.
- Separación del control y del conocimiento. Que nos permite intercambiar ambos sin preocuparnos de la funcionalidad del sistema.
- Modularización del conocimiento. Una regla puede ser revisada o ampliada de una manera sencilla.
- Facilidad de expansión. La separación entre control y conocimiento nos facilita la ampliación de este último.
- Utilización relevante del conocimiento. El sistema solo utilizará las reglas que sean importantes para la resolución del problema.
- Explicación de cómo se llegó a la solución. Siguiendo una traza de las reglas que se han ido disparando en el sistema.

Dentro de las desventajas de este tipo de Sistema Experto se encuentran:

- Se requiere un emparejamiento exacto en las reglas.
- Se tienen relaciones entre reglas no directas ($C \rightarrow D$, $B \rightarrow C$, $A \rightarrow B$). Esto puede provocar dificultades a la hora de realizar una traza en el sistema o determinar qué conjunto de reglas se ha utilizado para inferir conocimiento.
- Pueden ser sistemas lentos debido a que el motor de inferencia ha de decidir que reglas aplicar pudiendo llegar a contemplar todo el conjunto de reglas del sistema.

En resumen, un sistema basado en reglas es muy eficaz cuando tiene que analizar una gran cantidad de información, interpretarla y proporcionar una recomendación a partir de la misma, son buenos para predecir resultados futuros a partir del conocimiento que tienen.

1.8.4 Comparación entre los Sistemas basados en Reglas y los Sistemas basados en Probabilidad.

Después del estudio realizado de los Sistemas Expertos, sus tipos y formas de implementarlo se realizó una comparación entre estos. El sistema basado en casos fue descartado en esta comparación porque resulta muy difícil encontrar una estructura apropiada para describir el contenido de un caso, para organizarlo y almacenarlo; además resulta difícil su integración a la estructura de la base de casos y no se adecua a las características del Sistema Experto a implementar.

Los Sistemas Expertos basados en reglas y los basados en probabilidad son los más estudiados, difundidos y utilizados, por esta razón se realizó una tabla comparativa entre estos dos tipos de Sistemas Expertos, teniendo en cuenta la bibliografía consultada [15].

Como resultado de este análisis se decidió utilizar un Sistema Experto basado en reglas. Esta decisión se tomó teniendo en cuenta la tabla comparativa, las características de ambos sistemas, así como sus ventajas y desventajas. El sistema seleccionado es el más sencillo y el que más se ajusta a las particularidades del Sistema Experto que se necesita implementar. Además el conocimiento que se posee del sistema operativo Nova y de sus problemas puede ser relativamente fácil expresarlo mediante cláusulas del tipo $p \rightarrow q$, donde se cuenta con una causa y una consecuencia, es decir mediante reglas.

1.9 Los Sistemas Expertos en el mundo

❖ Primeros Sistemas Expertos:

Los Sistemas Expertos como tales, surgen a mediados de los años sesenta; en esos tiempos, se creía que bastaban unas pocas leyes de razonamiento junto con potentes ordenadores para producir resultados brillantes. Los primeros investigadores que desarrollaron programas basados en leyes de razonamiento fueron Alan Newell y Herbert Simon, quienes desarrollaron el GPS (General Problem Solver). Sin embargo, este programa no podía resolver problemas más “cotidianos” y reales, como, por ejemplo, dar un diagnóstico médico. Entonces algunos investigadores cambiaron el enfoque del problema: se dedicaron a resolver problemas sobre un área específica intentando simular el razonamiento humano; en vez de computarizar la inteligencia general, se centraron en dominios de conocimiento muy concretos. De esta manera nacieron los Sistemas Expertos. En la actualidad continúa el desarrollo de Sistemas Expertos para la solución de problemas de la vida cotidiana, para las empresas, hospitales, entre otras instituciones que se benefician con las ventajas que estos traen.

1.10 Los Sistemas Expertos en Cuba

En Cuba se han desarrollado un grupo de Sistemas Expertos. La principal esfera, en que se pueden encontrar es en la medicina, debido a la capacidad de razonamiento e interpretación que presentan. DITRITS, es un Sistema Experto para el diagnóstico y tratamiento de infecciones de transmisión sexual. Cuyas autoras son: Ing. Maribel García García, Lic. Karina Fernández Sánchez, MSc. Viviana R. Toledo Rivero, Dra. Lisset León Vidal. DITRITS responde a una necesidad social de la provincia de Cienfuegos con alcance nacional. [18]

Otro ejemplo de la utilización de Sistemas Expertos en Cuba es el trabajo: Evaluación de la aptitud de las tierras del municipio San José de las Lajas para las clases generales de uso agrícola ganadero. Desarrollado por Heriberto Vargas Rodríguez y Daniel Ponce de León Lima de la Universidad Agraria de La Habana. A partir de la creación de un Sistema Experto soportado en ALES, el uso del método Delphi

y técnicas de decisión multicriterio. [19] Existen otros ejemplos del desarrollo de Sistemas Expertos en Cuba, pero los planteados anteriormente son lo más relevantes.

1.11 Los Sistemas Expertos en la Universidad de las Ciencias Informáticas (UCI)

La Universidad de las Ciencias Informáticas concebida para que sus estudiantes y profesores centraran su labor en la docencia, investigación y producción es soporte de la informatización del país y la competitividad internacional de la industria cubana del software [20]. En esta se desarrollan producciones de software para la salud y equipos médicos, educación, telecomunicaciones, bioinformática, entre otras; vinculadas con diferentes instituciones, como el Centro Nacional de Genética Médica (CNGM), sede del Programa de la Revolución para el desarrollo de la genética médica en el país. Se han desarrollado Sistemas Expertos para diferentes ramas, siendo la medicina la rama más común y difundida en estos. A continuación se exponen varios ejemplos de Sistemas Expertos desarrollados en la Universidad de las Ciencias Informáticas, la mayoría de estos son sistemas basados en reglas.

Se cuenta con el Sistema Experto para el diagnóstico médico de las enfermedades genéticas con dismorfias (SEGEDIS). [21] La Universidad de las Ciencias Informáticas (UCI) en conjunto con el Centro Nacional de Genética Médica (CNGM) creó alasARBOGEN. El trabajo Implementación de un Sistema para el Cálculo de Riesgo de Enfermedades Genéticas desempeña un papel importante en el desarrollo de la segunda versión de alasARBOGEN. [22]

Otro de los Sistemas Expertos realizados en la UCI es para el “Diagnóstico y Tratamiento de la hipertensión arterial”, perteneciente al proyecto HIPER WEB [23]. Se realizó además una Herramienta inteligente para la captura de Requisitos No Funcionales a partir de las propiedades del sistema [24]. Se creó también un Modelo Basado en Casos para la planificación de proyectos de software en la Universidad de las Ciencias Informáticas [25]. Otro logro en este amplio campo de los Sistemas Expertos fue el desarrollo del sistema SEDIM-SV utilizando la metodología Weiss-Kulikowski. Este es un Sistema Experto de Diagnóstico Médico de Sepsis Vaginales (SEDIM-SV), como medio de ayuda, para la consulta de Ginecología de la Universidad de las Ciencias Informáticas [26]. Se cuenta también con una Herramienta de Generación de Código Mediante Sistema Experto[27].

1.12 Lenguajes utilizados en la construcción de Sistemas Expertos

Realizar un trabajo de diploma de implementación en la ingeniería informática implica el estudio de herramientas, lenguajes y tecnologías necesarias para lograr con éxito las tareas trazadas y así cumplir el objetivo principal del trabajo. Son muchos los lenguajes que en la actualidad se utilizan en la construcción de los Sistemas Expertos, pero dentro de estos se encuentran los que a lo largo de los años se han vuelto más comúnmente utilizados por sus características, facilidades e importancia. A continuación se realiza un resumen de las principales características de los lenguajes más utilizados en

la construcción de Sistemas Expertos.

❖ LISP

LISP, acrónimo de lenguaje de Procesamiento de Listas, fue inventado por John McCarthy y su equipo en la Universidad de Stanford a finales de 1950. Originalmente fue creado como un modelo computacional de procesos matemáticos, reflejando el rigor de las propias matemáticas. [28]

LISP actualmente está diseñado para manejar símbolos matemáticos (variables), por lo que es utilizado perfectamente para la investigación en IA, donde un símbolo puede representar cualquier cosa. LISP tiene dos características principales que lo hacen sobresalir de entre los demás lenguajes para IA: es altamente flexible y es indefinidamente extensible.

❖ CLIPS

CLIPS es un acrónimo de C Language Integrated Production System (Sistema de Producción Integrado en Lenguaje C) es una herramienta para el desarrollo de Sistemas Expertos creada por la Software Technology Branch (STB), NASA/Lyndon B. Johnson Space Center. [29]

Se diseñó para facilitar el desarrollo de software que modele el conocimiento humano:

- Con propósitos específicos: alta portabilidad, bajo coste, y facilidad de integración.
- CLIPS permite integración completa con otros lenguajes de programación como C o Ada.
- Puede ser llamado desde un lenguaje procedural, realizando su función y devolver el control al programa que le llamó.
- También se puede definir código procedural como funciones externas llamadas desde CLIPS.
- Cuando el código externo finaliza su ejecución devuelve el control a CLIPS.

❖ Prolog

Prolog es un lenguaje de programación que se centra alrededor de un conjunto pequeño de mecanismos, incluyendo reconocimiento de patrones, estructuras de datos basadas en árboles y backtracking (retroceso) automático. Este conjunto pequeño constituye una estructura de programación sorprendentemente poderosa y flexible. Prolog es ideal para resolver problemas que involucren objetos – en particular objetos estructurados – y relaciones entre ellos. [30]

Prolog cuenta con las siguientes características, las cuales lo hacen un lenguaje poderoso para la representación del conocimiento:

- Predicados que expresan relaciones entre entidades.
- Un método para definir predicados mediante reglas de aserción y hechos.
- Un método para hacer preguntas para comenzar cómputos.
- Un procedimiento de búsqueda hacia atrás para evaluar metas.
- Estructura de datos que pueden simular registros estilo Pascal o listas estilo LISP.
- Un reconocedor de patrones que construye y analiza las estructuras de datos.
- Un conjunto de predicados pre-construidos para aritmética, entrada y salida y servicios de

sistemas.

Los mecanismos importantes del Prolog son: recursividad, instanciación, verificación, unificación, backtracking e inversión.

De estos tres lenguajes más utilizados se hizo un análisis de cuál se iba a utilizar en el desarrollo del Sistema Experto. Teniendo en cuenta que el Sistema Experto a desarrollar tenía que implementarse en un lenguaje que se pudiera relacionar con PHP, pues con este lenguaje es con el que está implementado NovaDesk, se descartó LIPS por no ser posible esta unión, pues LIPS es compatible con C, es decir C++ y C#. De los dos lenguajes restantes se comenzó a investigar acerca de CLIPS y se encontró una extensión de este lenguaje que era para su unión con PHP llamada PHLIPS [31], sin embargo este proyecto quedó estancado en el año 2005 por no ser estable, por lo que CLIPS fue descartado también. Sólo quedaba entonces Prolog y en investigaciones realizadas se comprobó que este podía unirse con PHP mediante comandos y ficheros, además este lenguaje es el más sencillo de aprender de los tres y posee un motor de inferencia implícito que realiza: recursividad, instanciación, verificación, unificación y backtracking; mecanismos ideales para la construcción de un Sistema Experto.

1.13 Herramientas y shells utilizados en la construcción de Sistemas Expertos

Muchas son las herramientas que existen a nivel mundial que serían de utilidad para el desarrollo, pero al tener en cuenta las particularidades de la investigación es preciso definir cuáles serán aquellas que serán empleadas. Inicialmente cada Sistema Experto que se creaba se construía a partir de un lenguaje de IA tal como LISP. Pero después de que muchos Sistemas Expertos se construyeron así, quedó claro que estos sistemas estaban contruidos como un conjunto de representaciones declarativas (reglas) combinado con un intérprete de estas representaciones; también quedó claro que era posible separar el intérprete del conocimiento específico del dominio y por lo tanto se podían crear sistemas que podían ser usados para construir nuevos Sistemas Expertos agregando simplemente conocimiento correspondiente al dominio del nuevo problema.

Así nacieron los shells los cuales sirven como base para muchos Sistemas Expertos, puesto que proporcionan mucha flexibilidad en representación del conocimiento y razonamiento, así como adquisición de conocimiento. La mayoría de estos shells no son gratuitos pues fueron desarrollados para empresas y estas a su vez los comercializan. Dentro de estos shells uno de los más famosos es EMYCYN y uno de los más utilizados es CLIPS, sin embargo ninguno de estos dos será utilizado en el desarrollo de la tesis pues EMYCYN es sobre LIPS y CLIPS es para el lenguaje del mismo nombre, por esta razón no se utilizará ningún shell sino herramientas para crear Sistemas Expertos.

➤ Herramientas para crear Sistemas Expertos.

En epígrafes anteriores se abordaron los principales componentes de los Sistemas Expertos: base de

conocimientos, base de hechos y motor de inferencia (MI), es necesario retomar este último dado su complejidad. Para obtener conclusiones, los expertos utilizan diferentes tipos de reglas y estrategias de inferencia y control, que son utilizadas por el motor de inferencia para obtener conclusiones simples y compuestas. Implementar un nuevo motor de inferencia sería muy complejo, en cambio existen herramientas que pueden emplearse en la construcción de Sistemas Expertos, cuyas potencialidades proporcionan incluyen un motor de inferencia. Como resultado de una búsqueda en Internet, se hallaron disímiles prototipos como herramientas, unos emplean lenguaje lógico como Prolog, mientras que otros emplean lenguaje de alto nivel como C++, Visual Basic, C#, Java, etc. Como el lenguaje con el que se va a implementar el Sistema Experto es Prolog se escogen las herramientas para programar en este lenguaje y se descartan las restantes.

❖ **SWI Prolog**

Es de software libre bajo licencia pública GNU (GNU Public License) que ofrece un amplio ambiente de trabajo para Prolog. Su desarrollo comenzó en 1987 y ha sido impulsado por las necesidades de las aplicaciones del mundo real. Actualmente SWI-Prolog se utiliza ampliamente en la investigación y la educación, así como en aplicaciones comerciales.

Características más importantes:

- Núcleo bajo licencia LGPL, librerías de Prolog bajo licencia GPL con una declaración adicional que permite su uso en aplicaciones propietarias.
- Compilación rápida.
- Perfil de ejecución para Linux, plataformas Unix en su mayoría y Windows NT/2000/XP/Vista.
- Portable a muchas plataformas.
- Manipulación interna de caracteres UNICODE. Ideal para la Web y aplicaciones internacionales.
- Dispone de bibliotecas rápidas y flexibles para análisis de documentos SGML, XML y documentos escritos en RDF.

❖ **GNU Prolog:**

Es un compilador de Prolog software libre desarrollado por Daniel Díaz disponible para Unix y Windows. Se adapta al estándar ISO para Prolog con muchas extensiones muy útiles en la práctica: variables globales, interfaz a sistemas operativos (SO), sockets, y demás. También incluye programación con restricciones sobre un dominio finito y análisis usando gramáticas de cláusula definida.

Características más importantes:

- Conformado para el estándar ISO para Prolog (números con coma flotante, streams, código dinámico, entre otros).
- Una parte para extensiones: Variable globales, parsing usando gramáticas cláusula definida, interfaz sockets, interfaz a SO, etc.
- Más de 300 predicados integrados en Prolog.

- Depuración para el lenguaje Prolog y depurador a bajo nivel, *WAM*.
- Línea de edición facilitada bajo un intérprete interactivo con completamiento de código.

De las herramientas planteadas anteriormente, por ser la más utilizada, ofrecer un amplio ambiente de trabajo para Prolog, una compilación rápida y con una interfaz gráfica, se utilizará la herramienta SWI-Prolog para el desarrollo del Sistema Experto.

1.14 Otras herramientas y lenguajes a emplear.

El Sistema Experto a implementar será para la aplicación web NovaDesk y por esta razón tendrá particularidades y necesita la utilización de otras herramientas y lenguajes.

Una de estas herramientas serán los servidores web, dentro de estos se utilizará el **Apache**: servidor web hecho por excelencia, de código abierto, para plataformas Unix, Windows, Macintosh y otras. Su nivel de configuración, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. Es flexible, rápido y eficiente. Entre sus características se encuentra que es multiplataforma, puede ser adaptado a diferentes entornos y necesidades.

Como herramienta para el tratamiento de las imágenes se utilizará **GIMP** (GNU Image Manipulation Program), este es un programa de edición de imágenes digitales en forma de mapa de bits, trabaja tanto con dibujos como con fotografías. Es un programa libre y gratuito. Sirve para procesar gráficos y fotografías digitales en la cual podemos afirmar que es una gran herramienta del desarrollo del software libre.

Las Herramientas CASE (Computer Aided Software Engineering, en español Ingeniería de Software Asistida por ordenador), son diversas aplicaciones informáticas destinadas a aumentar la productividad en el desarrollo de software. De estas se utilizará **Visual Paradigm**, que utiliza UML (Unified Modeling Language, en español Lenguaje Unificado de Modelado)”: como lenguaje de modelado. Además posee características gráficas muy cómodas que facilitan la realización de los diagramas de modelado. Posee versiones tanto para Windows como para Linux propiciando ventajas favorables para el modelado y a su vez convirtiéndose en el más adecuado para ser usado.

Los sistemas de gestión de base de datos (SGBD) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. De estos se utilizará **MYSQL**, pues es el que utiliza GLPI, que es el Help desk sobre el que está implementado NovaDesk.

De las aplicaciones destinadas a la administración de SGBD se utilizará **PhpMyAdmin**. Es una herramienta escrita en PHP con la intención de manejar la administración de MySQL a través de una interfaz Web muy intuitiva. Actualmente puede crear y eliminar bases de datos, crear, eliminar y alterar tablas, borrar, editar y añadir campos, ejecutar cualquier sentencia SQL y hacer un backup de la base de datos, administrar claves en campos, administrar privilegios, exportar datos en varios formatos además de está disponible en 55 idiomas.

1.14.1 Lenguajes de programación Web.

Los lenguajes de programación Web han surgido debido a las tendencias y necesidades de las plataformas, con el objetivo de permitir la interacción con los usuarios y la utilización de los sistemas de Base de Datos, se clasifican en lenguajes del lado del cliente y lenguajes del lado del servidor. Entre los lenguajes que trabajan del lado del cliente se pueden citar algunos de ellos como son HTML, JAVASCRIPT, APPLETS DE JAVA, VISUAL BASIC SCRIPT (VBSCRIPT), CSS, XSL. En los lenguajes que trabajan del lado del servidor se pueden mencionar algunos como PERL, ASP, PHP, JSP, JAVA.

Los lenguajes de interés que se utilizarán para el desarrollo del Sistema Experto son los siguientes:

HTML: HTML, siglas de HyperText Markup Language (Lenguaje de Marcas de Hipertexto), es un lenguaje de composición de documentos y especificación de ligas de hipertexto que define la sintaxis y coloca instrucciones especiales que no muestra el navegador, aunque si le indica cómo desplegar el contenido del documento, incluyendo textos, imágenes y otros medios soportados.

Java Script: Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado. Es un lenguaje de programación bastante sencillo y pensado para hacer las cosas con rapidez, a veces con ligereza.

CSS: Las hojas de estilo en cascada CSS (Cascading Style Sheets) son un lenguaje formal usado para definir la presentación de un documento estructurado escrito en HTML o XML. El W3C (World Wide Web Consortium) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores.

PHP: Es un lenguaje de programación interpretado, diseñado originalmente para la creación de páginas Web dinámicas. Es usado principalmente en interpretación del lado del servidor. Es ampliamente usado, está diseñado especialmente para desarrollo Web y puede ser embebido dentro de código HTML. Es un lenguaje multiplataforma, tiene capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destacándose su conectividad con MySQL. Es libre, por lo que se presenta como una alternativa de fácil acceso para todos. En el Sistema Experto a implementar será de gran importancia porque todos los componentes de este excepto el motor de inferencia se desarrollarán utilizando el lenguaje PHP.

1.14.2 Herramientas para el desarrollo Web.

Existe una gran variedad de IDE (Entornos de Desarrollo Integrados) para PHP donde los más habituales son, Zend Studio, Eclipse, NetBeans, Quanta, Bluefish, entre otros.

NetBeans: Es un entorno de desarrollo para todo tipo de tecnologías de java e incluso permite la codificación de programas en C, C++ y otros. Editor de código sensible al contenido. Soporte para java, C, C++, XML y lenguajes HTML. Permite crear el visual de componentes gráficos. [32]

1.14.3 Metodología para el desarrollo del software.

La metodología de ingeniería de software es un enfoque estructurado, cuya finalidad es hacer más eficaz la producción y lograr alta calidad de una forma costeable. A nivel mundial existen diferentes metodologías para el desarrollo de software, como RUP y las Metodologías Ágiles. Para el desarrollo del trabajo de diploma se utilizará la propuesta de la metodología SXP elaborada por la ingeniera Peñalver Romero. SXP es un híbrido de metodologías ágiles que toma las mejores prácticas de las metodologías SCRUM y XP además de regirse por los lineamientos de calidad de la UCI. Es una metodología que se caracteriza por ser iterativa e incremental con pequeñas mejoras unas tras otras, basada en Historias de Usuario (HU), está atenta al cambio y permite que el equipo de programación se mantenga en frecuente interacción con el cliente o usuario. Está especialmente indicada para proyectos de pequeños equipos de trabajo, rápido cambio de requisitos o requisitos imprecisos, muy cambiantes, donde existe un alto riesgo técnico y se orienta a una entrega rápida de resultados y una alta flexibilidad.

Según el análisis realizado en el presente capítulo, se ha logrado el entendimiento de los conceptos más relevantes de los Sistemas Expertos, sus características, componentes, ventajas, desventajas y clasificaciones. El Sistema Experto a implementar será basado en reglas. Con el estudio previo de las herramientas, tecnologías, metodologías y lenguajes propuestos a utilizar en el desarrollo del software, se obtiene finalmente como metodología para el desarrollo, análisis y diseño del software SXP, herramienta CASE Visual Paradigm, lenguajes de programación para la implementación PROLOG, PHP, HTML, CSS y JAVASCRIPT. Como herramientas para el desarrollo de aplicaciones NetBeans y SWI prolog. Para la gestión de las bases de datos MySQL y como sistema para la administración PhpMyAdmin.

Capítulo 2: “Sistema Experto para la gestión de la base de conocimientos de NovaDesk”

2.1 Sistema Experto propuesto.

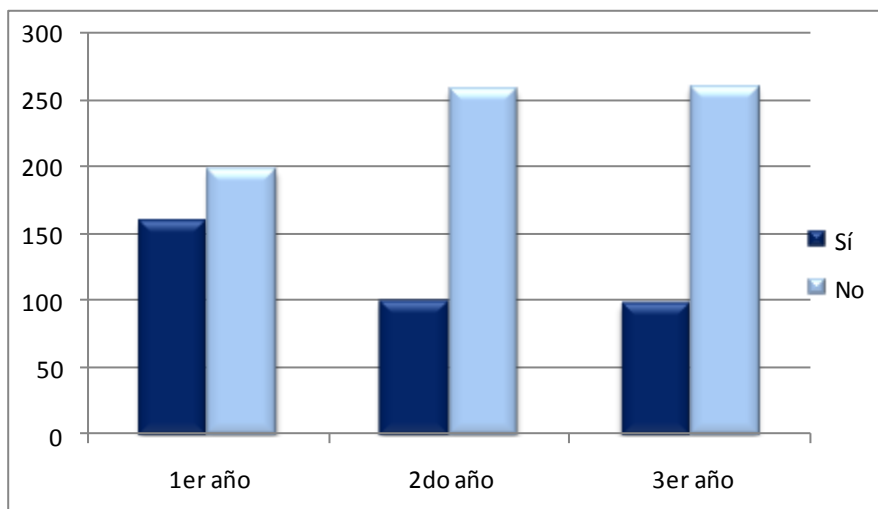
Desarrollar un Sistema Experto es una tarea compleja y cuenta con una serie de pasos lógicos que deben ser seguidos. En el epígrafe 1.7 del capítulo anterior se abordan estos pasos o etapas. A continuación se explica cómo se aplican estos pasos para el desarrollo del Sistema Experto del trabajo de diploma.

Primero se define el problema a resolver con el Sistema Experto: la gestión de la base de conocimientos de NovaDesk. Se identifican los problemas que presenta un usuario en el trabajo con el sistema operativo Nova mediante encuestas a los estudiantes de primero, segundo y tercer año de la facultad 1 de la Universidad de las Ciencias Informáticas, por ser estos los que más trabajan con este sistema operativo. Esta encuesta se aplicó a un total de 846 estudiantes: 257 de primer año, 248 de segundo año y 341 de tercero. Los resultados arrojados por la misma se encuentran recogidos en la Tabla 2.1, así como los aspectos abordados en la misma.

Encuesta	1ro	2do	3ro	Total	%
Les gusta trabajar con Nova	160	100	98	358	42,32
Han encontrado problemas en su trabajo con Nova	155	198	295	648	76,6
Pueden solucionar los problemas que encuentran	155	48	95	298	35,22
Encuestados por año	257	248	341	846	100

Tabla 2.1 Resultados de las encuestas a los estudiantes.

Uno de los aspectos encuestados fue la preferencia de trabajar con Nova cuyo resultado se muestra en la Figura 2.1. Los estudiantes de primer año se han adaptado fácilmente a este sistema operativo, al contrario de los restantes años que poseen experiencia en el trabajo con otros sistemas operativos.



CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Figura 2.1 Resultados de las encuestas a los estudiantes en cuanto a su preferencia de trabajar con Nova.

Otro aspecto analizado fue si los estudiantes encontraban problemas en su trabajo con el sistema operativo Nova. Esto dio como resultado que los estudiantes que llevan más tiempo trabajando con Nova encuentran más problemas pues poseen más conocimientos al respecto, lo anterior se muestra en la Figura 2.2.

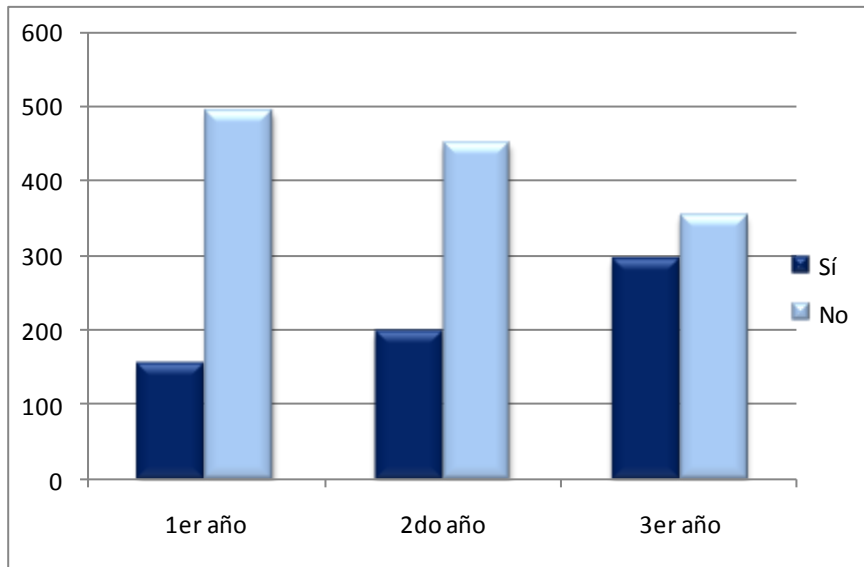


Figura 2.2 Resultados de las encuestas a los estudiantes en cuanto a los problemas encontrados.

Se analizó además el aspecto relacionado a si los estudiantes eran capaces de solucionar los problemas que encontraban, brindando como resultado que los estudiantes no pueden solucionar estos problemas como se muestra en la Figura 2.3. Esto está determinado por la cantidad de problemas encontrados y el nivel de dificultad de los mismos.

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

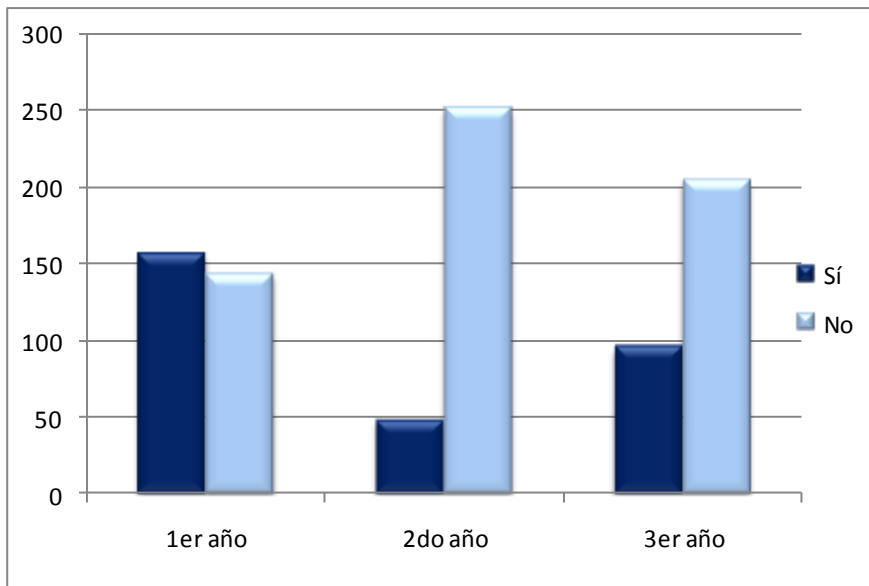


Figura 2.3 Resultados de las encuestas a los estudiantes en cuanto a si pueden solucionar los problemas.

De forma general, a los estudiantes no les gusta trabajar con el sistema operativo Nova porque encuentran muchos problemas en su trabajo con el mismo y no saben cómo solucionarlos, sin embargo, ellos aspiran tener una vía o medio para solucionar estos problemas por sí solos.

Para profundizar en los resultados de la encuesta y obtener toda la información de este sistema operativo: principales problemas y soluciones, se entrevistaron a los expertos humanos. Los expertos fueron especialistas en el sistema operativo Nova, que trabajan en el proyecto Nova de la facultad 1 [Anexo 6]. La información obtenida pasó a formar parte de la base de conocimiento de NovaDesk.

Los problemas encontrados luego de procesadas las encuestas y las entrevistas de los expertos son los siguientes: Montaje de Particiones NTFS, Verificar la integridad de las imágenes ISO, Instalar impresora, Reinstalar Windows sin perder Nova (Restaurar el grub), Conectarse por medio de una red inalámbrica, Conectarse por medio de una red cableada, Montaje de dispositivos en Nova, Unir Nova a un dominio, Cambiar la configuración del teclado, Comprimir y extraer archivos, Configuración e instalación del escáner.

En el capítulo anterior se estudió qué tipo de Sistema Experto se implementaría y por sus características se seleccionó el basado en reglas. Además se eligieron las herramientas y lenguajes a utilizar: NetBeans, SWI prolog, PROLOG, PHP, HTML, CSS y JAVASCRIPT.

De cada problema se buscó las causas que los provocaban y se procedió a elaborar las reglas en lenguaje PROLOG, quedando de la siguiente manera algunas de las que contiene el Sistema Experto:

Regla 1:

problema(montaje_de_particiones_NTFS):- monto_particion.

Regla 2:

problema(montaje_de_particiones_NTFS):- acceso_particion.

Regla 3:

problema(verificacion_de_la_integridad_de_las_imagenes_ISO):- imagen_nova_bien.

Regla 4:

problema(instalar_impresora):- instalar_impresora.

Regla 5:

problema(instalar_impresora):- funciona_impresora.

Regla 6:

problema(restaurar_el_grub):- windows_perder_nova.

Regla 7:

problema(conectarse_por_medio_de_una_red_inalambrica):- red_inalambrica_tarjeta_red.

Regla 8:

problema(conectarse_por_medio_de_una_red_inalambrica):- red_inalambrica_falla_conectar.

Regla 9:

problema(conectarse_por_medio_de_una_red_cableada):- red_cableada_tarjeta_red.

Regla 10:

problema(conectarse_por_medio_de_una_red_cableada):- red_cableada_falla_conectar.

Se diseñaron todas las partes que formarían el Sistema Experto: una base de datos, una base de conocimientos, un motor de inferencia, un subsistema de explicación, la interfaz de usuario y un subsistema de adquisición del conocimiento. Quedando este diseño representado en la Figura 2.4.

La *base de conocimientos* de NovaDesk va a contener la información incorporada a través de los datos de entrada (problemas y soluciones encontradas del sistema operativo Nova).

La *base de datos* va a contener las reglas y preguntas del Sistema Experto que se gestionarán a través del *subsistema de adquisición del conocimiento*. El *motor de inferencia*, haciendo uso de la *base de conocimiento* y la *base de datos* obtendrá la solución del problema que presenta el usuario. Este problema encontrado con su solución se le mostrará al usuario a través de la relación entre el *subsistema de explicación* y el *subsistema de interfaz de usuario*, con el que interactuará el usuario mediante preguntas y respuestas.

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

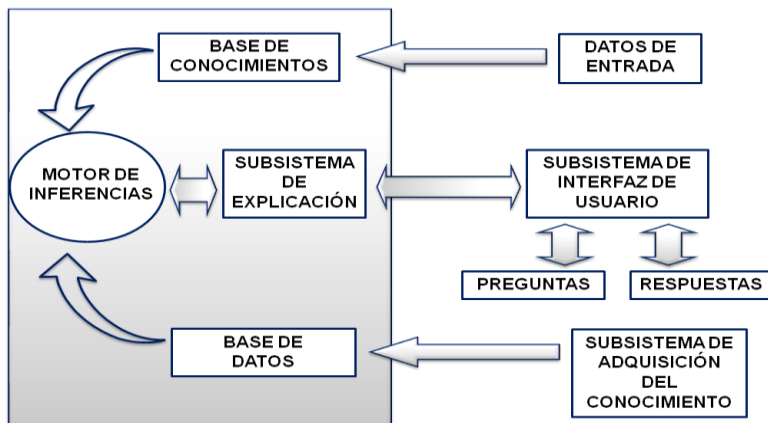


Figura 2.4 Partes del Sistema Experto para la gestión de la base de conocimientos de NovaDesk.

Posteriormente se comenzó el desarrollo del Sistema Experto utilizando la metodología ágil SXP como se aborda en los epígrafes siguientes.

2.2 Proceso de desarrollo del proyecto basado en SXP.

Con el objetivo de llevar a cabo el proceso de desarrollo de forma organizada y efectiva, se toma como base en la metodología ágil SXP, la creación de artefactos (tabla de roles, historias de usuario, modelos auxiliares) que permiten un desarrollo centrado en subtareas, permitiendo mayor rapidez en el cumplimiento de las tareas asignadas.

Rol	Responsabilidad	Nombre
Gerente	Dirige y controla las tareas del equipo. Toma las decisiones finales. Participa en la selección de objetivos y requerimientos. Controla el progreso y da seguimiento a cada iteración. Evalúa si los objetivos son alcanzables con las restricciones de tiempo y recursos presentes.	Yaima Oval Riverón.
Cliente	Participa en las tareas que involucran la lista de reserva del producto.	Integrantes del proyecto SIMAYS
Programadores	Elabora el código de las nuevas funcionalidades a implementar. Escribe las pruebas unitarias. Debe existir una comunicación y coordinación adecuada entre los programadores y el resto del equipo.	Alfredo Pérez Benitez Susana Sánchez Ortiz
Analista	Es el encargado de escribir las historias de usuario y las pruebas funcionales para validar su implementación.	Susana Sánchez Ortiz
Diseñadores	Encargado del diseño del sistema, de los prototipos de interfaces máximos responsables de la realización del diseño de las	Alfredo Pérez Benitez

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

	metáforas y supervisan el proceso de construcción.	
Encargado de pruebas.	Ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.	Alfredo Pérez Benitez Susana Sánchez Ortiz
Arquitecto	Se vincula directamente con el analista y el diseñador debido a que su trabajo tiene que ver con la estructura y el diseño en grande del sistema. Ayuda en el diseño de las metáforas.	Alfredo Pérez Benitez Susana Sánchez Ortiz

Tabla 2.2 Planificación del proyecto por roles

2.3 Modelo de dominio.

Un sistema por pequeño que parezca no lo hace sencillo, por ello se establece una técnica para la especificación de los requisitos más importantes del sistema, que va a dar soporte al negocio, “el modelo del negocio”. En dependencia de la situación o escenario se determina si es necesario un modelo completo del negocio o de lo contrario se procede a definir el modelo conceptual o modelo de dominio.

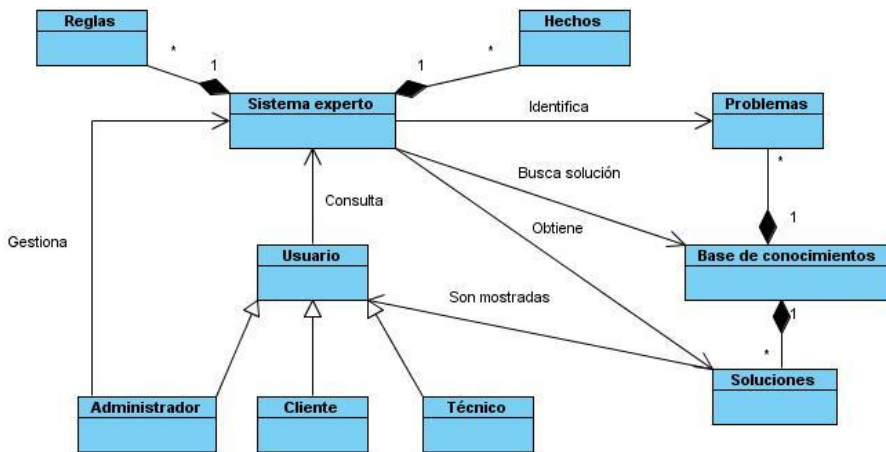


Figura 2.5 Modelo de dominio del Sistema Experto.

2.4 Lista de Reserva del Producto (LRP).

La Lista de Reserva del Producto (LRP) está conformada por una lista priorizada que define el trabajo a realizar en el proyecto. Tiene como objetivo comprobar que el producto resultante sea el definido, el más correcto, de utilidad y competitivo. Solo puede ser modificado entre iteraciones.

Prioridad	Ítem	Descripción	Estimación	Estimado por
Muy Alta				
	1.	Insertar reglas del sistema experto	2 días	ANA

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

	2.	Mostrar reglas del sistema experto	2 días	ANA
	3.	Eliminar reglas del sistema experto	2 días	ANA
	4.	Modificar reglas del sistema experto	2 días	ANA
	5.	Insertar preguntas para identificar problemas	2 días	ANA
	6.	Mostrar preguntas para identificar problemas	2 días	ANA
	7.	Eliminar preguntas para identificar problemas	2 días	ANA
	8.	Modificar preguntas para identificar problemas	2 días	ANA
	9.	Identificar categoría del problema	2 días	ANA
	10.	Insertar pregunta asociada a la categoría de clasificación de los problemas	2 días	ANA
	11.	Mostrar pregunta asociada a la categoría de clasificación de los problemas	2 días	ANA
	12.	Eliminar pregunta asociada a la categoría de clasificación de los problemas	2 días	ANA
	13.	Modificar pregunta asociada a la categoría de clasificación de los problemas	2 días	ANA
Alta				
	14.	Identificar el problema mediante las respuestas del usuario	5 días	ANA
	15.	Buscar solución al problema encontrado	5 días	ANA
	16.	Llenar la base de conocimiento de NovaDesk	10 días	ANA
Media				
	17.	Seleccionar respuestas de cada una de las preguntas	5 días	ANA
	18.	Mostrar respuestas dadas por el usuario	5 días	ANA
	19.	Mostrar solución del problema encontrado	5 días	ANA
Baja				
	20.	Enviar solución por correo	3 días	ANA
	21.	Registrar incidencia si no se identificó el problema	5 días	ANA
RNF (Requisitos No Funcionales)				
	25.	Utilización del lenguaje php, html, css y prolog.		
	26.	Ajustarse al diseño de NovaDesk.		
	27.	Interfaz amigable.		

Tabla 2.3 Lista de Reserva del producto

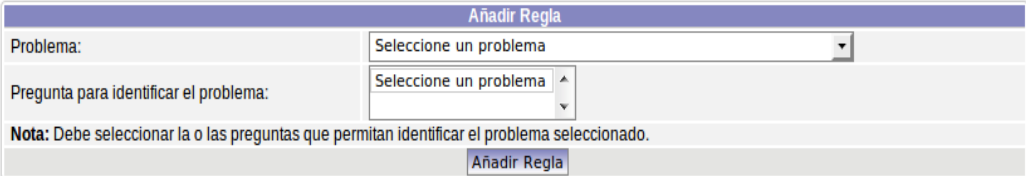
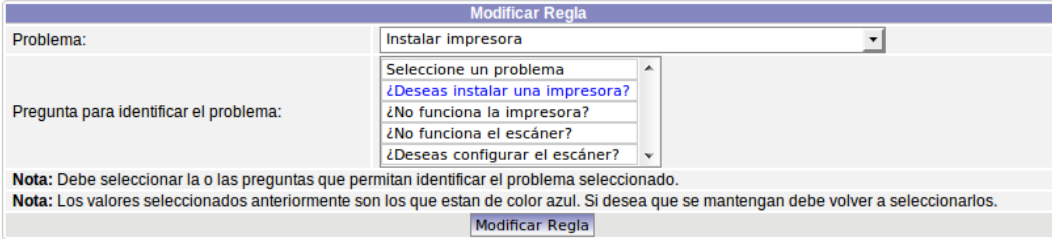
2.5 Historias de usuario.

Cuando se hace uso de la metodología SXP, para especificar los casos de uso como correspondería en

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

RUP, se hace mediante historias de usuarios, descriptoras de las tareas que el sistema debe hacer, cuestión que depende en gran medida de las especificaciones realizadas por el cliente. Se escriben con un lenguaje natural y con palabras concisas para no exceder su tamaño en unas pocas líneas de texto. Van a ser la guía para la construcción posterior de las pruebas de aceptación comprobando de esta manera la correcta implementación de las historias de usuario.

I. Gestionar reglas del Sistema Experto

Historia de Usuario	
Número: HU_1	Nombre Historia de Usuario: Gestionar reglas del Sistema Experto
Modificación de Historia de Usuario Número: ninguna	
Usuario: Susana Sánchez Ortiz	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1
Descripción: El Sistema Experto a implementar es un sistema basado en reglas por lo que necesita que estas sean insertadas, modificadas, mostradas y eliminadas.	
Observaciones: Primero se tienen que definir las reglas con un lenguaje natural y luego deben ser adaptadas al lenguaje en el que se implemente el Sistema Experto, en este caso al lenguaje PROLOG.	
Prototipo de interfaz:	
Insertar Reglas:	
	
Modificar Reglas:	
	
Mostrar y eliminar Reglas:	

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Reglas del Sistema Experto					
	Problema	Categoría	Pregunta para identificar el problema	Reglas del Sistema Experto en lenguaje Prolog	
	Montaje de particiones NTFS	Red y Seguridad	¿No puedes acceder a una partición del disco?	problema(montaje_de_particiones_ntfs):-hecho(acceso_particion).	Editar?
	Montaje de particiones NTFS	Red y Seguridad	¿No puedes montar una partición del disco?	problema(montaje_de_particiones_ntfs):-hecho(monto_particion).	Editar?
	Verificación de la integridad de las imágenes ISO	Instalación	¿Deseas comprobar si la imagen de Nova se descargó bien?	problema(verificacion_de_la_integridad_de_las_imagenes_iso):-hecho(imagen_nova_bien).	Editar?
	Restaurar el grub	Instalación	¿Deseas reinstalar Windows sin perder Nova?	problema(restaurar_el_grub):-hecho(windows_perder_nova).	Editar?

Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 1.1	Número Historia de Usuario: HU_1
Nombre Tarea: Identificar problemas presentes en Nova.	
Tipo de Tarea : Investigación	Puntos Estimados: 1
Fecha Inicio: 8/11/2010	Fecha Fin: 14/11/2010
Programador Responsable: Susana Sánchez Ortiz y Alfredo Pérez Benitez	
Descripción: Identificar mediante encuestas, entrevistas a los expertos, estudio de los foros de nova, entre otros medios, los principales problemas que presenta el sistema operativo Nova. Además identificar las causas de estos problemas y buscar su solución.	

Tarea de Ingeniería	
Número Tarea: 1.2	Número Historia de Usuario: HU_1
Nombre Tarea: Definir las reglas del Sistema Experto.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 15/11/2010	Fecha Fin: 21/11/2010
Programador Responsable: Susana Sánchez Ortiz	
Descripción: Mediante los problemas identificados, se identifican y definen las reglas que va a poseer el Sistema Experto.	

Tarea de Ingeniería	
Número Tarea: 1.3	Número Historia de Usuario: HU_1
Nombre Tarea: Escribir las reglas en lenguaje Prolog	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 15/11/2010	Fecha Fin: 21/11/2010

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Programador Responsable: Susana Sánchez Ortiz
Descripción: Las reglas identificadas deben de ser escritas en lenguaje PROLOG, que es el lenguaje con el que se va a implementar el Sistema Experto.

Tarea de Ingeniería	
Número Tarea: 1.4	Número Historia de Usuario: HU_1
Nombre Tarea: Insertar reglas	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 29/11/2010	Fecha Fin: 4/12/2010
Programador Responsable: Susana Sánchez Ortiz	
Descripción: Implementar la funcionalidad que permita insertar las reglas del Sistema Experto.	

Tarea de Ingeniería	
Número Tarea: 1.5	Número Historia de Usuario: HU_1
Nombre Tarea: Modificar reglas	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 29/11/2010	Fecha Fin: 4/12/2010
Programador Responsable: Susana Sánchez Ortiz	
Descripción: Implementar la funcionalidad que permita modificar las reglas que posee el Sistema Experto.	

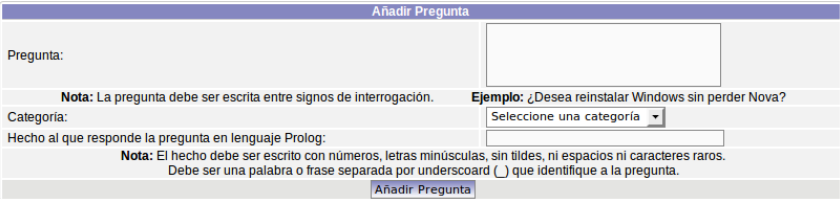
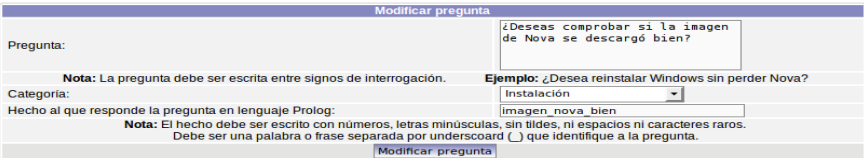

Tarea de Ingeniería	
Número Tarea: 1.6	Número Historia de Usuario: HU_1
Nombre Tarea: Eliminar reglas	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 29/11/2010	Fecha Fin: 4/12/2010
Programador Responsable: Susana Sánchez Ortiz	
Descripción: Implementar la funcionalidad que permita eliminar las reglas que posee el Sistema Experto.	

Tarea de Ingeniería	
Número Tarea: 1.7	Número Historia de Usuario: HU_1
Nombre Tarea: Mostrar reglas	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 29/11/2010	Fecha Fin: 4/12/2010

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Programador Responsable: Susana Sánchez Ortiz
Descripción: Implementar la funcionalidad que permita mostrar las reglas que posee el Sistema Experto.

II. Gestionar preguntas del Sistema Experto

Historia de Usuario	
Número: HU_2	Nombre Historia de Usuario: Gestionar preguntas del Sistema Experto
Modificación de Historia de Usuario Número: ninguna	
Usuario: Alfredo Pérez Benitez	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1
Descripción: Debe permitir insertar, modificar, eliminar y mostrar las preguntas de las que esté compuesto el Sistema Experto. Estas preguntas constituyen la base para que el Sistema Experto pueda determinar el problema al que se está enfrentando el usuario en ese momento.	
Observaciones: Estas preguntas deben ser claras y concisas para que permitan identificar bien el problema. Cada una de estas preguntas va a tener asignada una categoría.	
Prototipo de interfaz:	
Insertar Pregunta:	
	
Modificar Pregunta:	
	
Mostrar y eliminar pregunta:	
	

Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 2.1	Número Historia de Usuario: HU_2
Nombre Tarea: Definir las preguntas que conforman el Sistema Experto	

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Tipo de Tarea : Investigación	Puntos Estimados: 1
Fecha Inicio: 15/11/2010	Fecha Fin: 21/11/2010
Programador Responsable: Susana Sánchez Ortiz y Alfredo Pérez Benitez	
Descripción: Investigar y definir las preguntas que van a conformar el Sistema Experto.	

Tarea de Ingeniería	
Número Tarea: 2.2	Número Historia de Usuario: HU_2
Nombre Tarea: Clasificar y agrupar las preguntas por categoría.	
Tipo de Tarea : Investigación	Puntos Estimados: 1
Fecha Inicio: 15/11/2010	Fecha Fin: 21/11/2010
Programador Responsable: Susana Sánchez Ortiz y Alfredo Pérez Benitez	
Descripción: Clasificar y agrupar las preguntas definidas por las categorías que van a agrupar a los problemas identificados.	

Tarea de Ingeniería	
Número Tarea: 2.3	Número Historia de Usuario: HU_2
Nombre Tarea: Definir los hechos que brindan las preguntas	
Tipo de Tarea : Investigación	Puntos Estimados: 1
Fecha Inicio: 15/11/2010	Fecha Fin: 21/11/2010
Programador Responsable: Susana Sánchez Ortiz y Alfredo Pérez Benitez	
Descripción: Investigar y definir los hechos que va a brindar cada una de las preguntas al Sistema Experto.	

Tarea de Ingeniería	
Número Tarea: 2.4	Número Historia de Usuario: HU_2
Nombre Tarea: Insertar preguntas	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 22/11/2010	Fecha Fin: 28/11/2010
Programador Responsable: Alfredo Pérez Benitez	
Descripción: Implementar la funcionalidad que permita insertar las preguntas en el Sistema Experto.	

Tarea de Ingeniería	
Número Tarea: 2.5	Número Historia de Usuario: HU_2
Nombre Tarea: Modificar preguntas	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Fecha Inicio: 22/11/2010	Fecha Fin: 28/11/2010
Programador Responsable: Alfredo Pérez Benitez	
Descripción: Implementar la funcionalidad que permita modificar las preguntas que posee el Sistema Experto.	

Tarea de Ingeniería	
Número Tarea: 2.6	Número Historia de Usuario: HU_2
Nombre Tarea: Eliminar preguntas	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 22/11/2010	Fecha Fin: 28/11/2010
Programador Responsable: Alfredo Pérez Benitez	
Descripción: Implementar la funcionalidad que permita eliminar las preguntas que posee el Sistema Experto.	

Tarea de Ingeniería	
Número Tarea: 2.7	Número Historia de Usuario: HU_2
Nombre Tarea: Mostrar preguntas	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 22/11/2010	Fecha Fin: 28/11/2010
Programador Responsable: Alfredo Pérez Benitez	
Descripción: Implementar la funcionalidad que permita mostrar las preguntas que posee el Sistema Experto.	

III. Identificar la categoría del problema

Historia de Usuario	
Número: HU_3	Nombre Historia de Usuario: Identificar la categoría del problema
Modificación de Historia de Usuario Número: Ninguna	
Usuario: Susana Sánchez Ortiz	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1
Descripción: Cada problema va a tener una categoría asignada. Cada categoría tiene sus preguntas definidas. Esta historia de usuario consiste en una serie de preguntas que se le plantean al usuario para determinar la categoría del problema que se le presenta en ese momento. Y así se le muestren solo las preguntas relacionadas a su categoría para que el Sistema Experto pueda identificar correctamente el problema.	
Observaciones: El usuario debe dar una respuesta a cada una de las preguntas que se le presenten para determinar cuál es la categoría de su problema.	
Prototipo de interface:	

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Seleccione una pregunta
<input type="radio"/> ¿Presentas algún problema durante la instalación del sistema operativo Nova?
<input type="radio"/> ¿Presentas algún problema relacionado con una aplicación?
<input type="radio"/> ¿Presentas algún problema de conexión, no puedes montar una memoria o tienes problemas de privilegio?
<input type="radio"/> ¿Presentas algún problema relacionado con algún dispositivo o hardware de la máquina?
<input type="button" value="Aceptar"/>
Nota: Si su problema no está enmarcado en ninguna de las preguntas anteriores de clic AQUI para que lo registre como una incidencia.

Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 3.1	Número Historia de Usuario: HU_3
Nombre Tarea: Definir las preguntas que identificarán cada una de las categorías.	
Tipo de Tarea : Investigación	Puntos Estimados: 1
Fecha Inicio: 8/11/2010	Fecha Fin: 14/11/2010
Programador Responsable: Susana Sánchez Ortiz y Alfredo Pérez Benitez	
Descripción: Cada categoría de clasificación de problemas va a tener una pregunta asociada que va a permitir identificarla.	

Tarea de Ingeniería	
Número Tarea: 3.2	Número Historia de Usuario: HU_3
Nombre Tarea: Implementar la funcionalidad que permite identificar la categoría del problema.	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 8/11/2010	Fecha Fin: 14/11/2010
Programador Responsable: Susana Sánchez Ortiz	
Descripción: Implementar la funcionalidad que permite identificar la categoría del problema. Se le presentarán las preguntas que identifican cada una de las categorías al usuario y de acuerdo a las respuestas que de este se identificará la categoría del problema que presenta.	

IV. Gestionar las preguntas para identificar las categorías de clasificación de problemas

Historia de Usuario	
Número: HU_4	Nombre Historia de Usuario: Gestionar las preguntas para identificar las categorías de clasificación de problemas
Modificación de Historia de Usuario Número: ninguna	
Usuario: Alfredo Pérez Benitez	Iteración Asignada: 2
Prioridad en Negocio: Alta	Puntos Estimados: 1
Riesgo en Desarrollo: Medio	Puntos Reales: 1
Descripción: Cada problema tiene asignada una categoría y estas van a tener asociada una o varias preguntas que pueden ser insertadas, modificadas, mostradas y eliminadas de acuerdo aparezcan nuevos problemas con	

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

nuevas clasificaciones y nuevas categorías.

Observaciones: Cada categoría que esté presente en el Sistema Experto va a tener una o varias preguntas asociadas que van a permitir su identificación.

Prototipo de interface:

Insertar Pregunta asociada a la Categoría:

Añadir pregunta asociada a la Categoría	
Categoría:	<input type="text" value="Seleccione una categoría"/>
Categoría:	<input type="text" value="Pregunta asociada a la Categoría:"/>
<p>Nota: La pregunta debe ser escrita entre signos de interrogación. Ejemplo: ¿Presentas algún problema durante la instalación del sistema operativo Nova?</p>	
<input type="button" value="Añadir"/>	

Modificar Pregunta asociada a la Categoría:

Modificar pregunta asociada a la Categoría	
Categoría:	<input type="text" value="Instalación"/>
Categoría:	<input type="text" value="¿Presentas algún problema durante la instalación del sistema operativo Nova?"/>
<p>Nota: La pregunta debe ser escrita entre signos de interrogación. Ejemplo: ¿Presentas algún problema durante la instalación del sistema operativo Nova?</p>	
<input type="button" value="Modificar"/>	

Mostrar y eliminar las preguntas asociadas a cada Categoría:

Categorías de las preguntas		
	Categoría	Pregunta asociada a la Categoría
	Instalación	¿Presentas algún problema durante la instalación del sistema operativo Nova? Editar?
	Red y Seguridad	¿Presentas algún problema de conexión, no puedes montar una memoria o tienes problemas de privilegio? Editar?
	Multimedia	¿Presentas algún problema relacionado con la reproducción o edición de imagen, video o audio? Editar?
	Kernel y Hardware	¿Presentas algún problema relacionado con algún dispositivo o hardware de la máquina? Editar?
	Repositorio	¿Presentas algún problema relacionado con la instalación o actualización de alguna aplicación en el sistema? Editar?
	Juegos	¿Presentas algún problema relacionado con emular o abrir algún juego? Editar?
	Aplicaciones	¿Presentas algún problema relacionado con una aplicación? Editar?

Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 4.1	Número Historia de Usuario: HU_4
Nombre Tarea: Definir las preguntas que van a estar asociadas a las categorías de clasificación de los problemas.	
Tipo de Tarea : Investigación	Puntos Estimados: 1
Fecha Inicio: 8/11/2010	Fecha Fin: 14/11/2010
Programador Responsable: Susana Sánchez Ortiz y Alfredo Pérez Benitez	
Descripción: Identificar y definir las preguntas que van a estar asociadas a las categorías de clasificación de los problemas, según las características que estas presenten.	

Tarea de Ingeniería	
Número Tarea: 4.2	Número Historia de Usuario: HU_4
Nombre Tarea: Insertar pregunta asociada a la categoría	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 15/11/2010	Fecha Fin: 21/11/2010
Programador Responsable: Alfredo Pérez Benitez	

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Descripción: Implementar la funcionalidad que permita insertar las preguntas asociadas a las categorías en las que se van a clasificar los problemas.

Tarea de Ingeniería	
Número Tarea: 4.3	Número Historia de Usuario: HU_4
Nombre Tarea: Eliminar pregunta asociada a la categoría	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 15/11/2010	Fecha Fin: 21/11/2010
Programador Responsable: Alfredo Pérez Benitez	
Descripción: Implementar la funcionalidad que permita eliminar las preguntas asociadas a las categorías en las que se van a clasificar los problemas que hayan sido insertadas.	

Tarea de Ingeniería	
Número Tarea: 4.4	Número Historia de Usuario: HU_4
Nombre Tarea: Modificar pregunta asociada a la categoría	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 15/11/2010	Fecha Fin: 21/11/2010
Programador Responsable: Alfredo Pérez Benitez	
Descripción: Implementar la funcionalidad que permita modificar las preguntas asociadas a las categorías en las que se van a clasificar los problemas que hayan sido insertadas.	

Tarea de Ingeniería	
Número Tarea: 4.5	Número Historia de Usuario: HU_4
Nombre Tarea: Mostrar pregunta asociada a la categoría	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 15/11/2010	Fecha Fin: 21/11/2010
Programador Responsable: Alfredo Pérez Benitez	
Descripción: Implementar la funcionalidad que permita mostrar las preguntas asociadas a las categorías en las que se van a clasificar los problemas que hayan sido insertadas.	

V. Identificar problemas

Historia de Usuario	
Número: HU_5	Nombre Historia de Usuario: Identificar problema
Modificación de Historia de Usuario Número: ninguna	
Usuario: Susana Sánchez Ortiz	Iteración Asignada: 3
Prioridad en Negocio: Alta	Puntos Estimados: 2

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Riesgo en Desarrollo: Alto	Puntos Reales: 1
Descripción: Mediante las respuestas del usuario a las preguntas que le presenta el Sistema Experto y las reglas de la categoría del problema, se identificará el problema que presenta el usuario.	
Observaciones: Para que esto sea posible se tienen que tener bien definidas las reglas que son las que van a permitir identificar el problema con los datos que brindarán las respuestas a las preguntas hechas al usuario.	

Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 5.1	Número Historia de Usuario: HU_5
Nombre Tarea: Procesar respuestas	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 10/01/11	Fecha Fin: 15/01/11
Programador Responsable: Susana Sánchez Ortiz	
Descripción: Las respuestas que brinden los usuarios deben ser procesadas para seleccionar de ahí los datos que permitan identificar el problema.	

Tarea de Ingeniería	
Número Tarea: 5.2	Número Historia de Usuario: HU_5
Nombre Tarea: Identificar problema	
Tipo de Tarea : Desarrollo	Puntos Estimados: 2
Fecha Inicio: 10/01/11	Fecha Fin: 22/01/11
Programador Responsable: Susana Sánchez Ortiz	
Descripción: Implementar la funcionalidad que permita identificar el problema que presenta el usuario utilizando el motor de inferencia de PROLOG.	

VI. Buscar solución del problema

Historia de Usuario	
Número: HU_6	Nombre Historia de Usuario: Buscar solución del problema
Modificación de Historia de Usuario Número: ninguna	
Usuario: Alfredo Pérez Benitez	Iteración Asignada: 3
Prioridad en Negocio: Alta	Puntos Estimados: 1
Riesgo en Desarrollo: Medio	Puntos Reales: 1
Descripción: La presente historia de usuario tiene como objetivo buscar en la base de conocimiento de NovaDesk la solución al problema encontrado.	

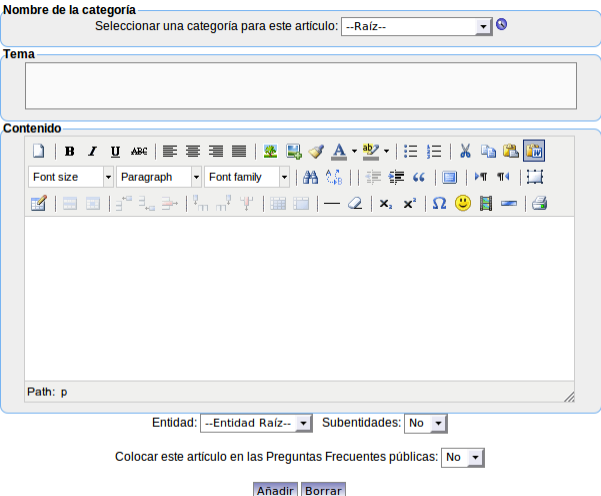
CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Observaciones: Primero se tiene que llenar la base de conocimientos de NovaDesk con información suficiente de cada uno de los problemas que pueda identificar el Sistema Experto.

Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 6.1	Número Historia de Usuario: HU_6
Nombre Tarea: Buscar solución del problema	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 24/01/11	Fecha Fin: 29/01/11
Programador Responsable: Alfredo Pérez Benitez	
<p>Descripción: Buscar el problema identificado en la base de conocimientos de NovaDesk para obtener la solución del mismo.</p>	

VII. Llenar la base de conocimientos

Historia de Usuario	
Número: HU_7	Nombre Historia de Usuario: Llenar la base de conocimientos
Modificación de Historia de Usuario Número: ninguna	
Usuario: Susana Sánchez Ortiz	Iteración Asignada: 3
Prioridad en Negocio: Alta	Puntos Estimados: 1
Riesgo en Desarrollo: Alto	Puntos Reales: 1
<p>Descripción: Llenar la base de conocimientos de NovaDesk desde el Sistema Experto.</p>	
<p>Observaciones: El administrador del Sistema Experto cuando inserte un nuevo problema debe a su vez añadir a la base de conocimientos este problema con su solución.</p>	
<p>Prototipo de interface:</p> 	

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 7.1	Número Historia de Usuario: HU_4
Nombre Tarea: Llenar la base de conocimiento	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 31/01/11	Fecha Fin: 05/02/11
Programador Responsable: Susana Sánchez Ortiz	
<p>Descripción: Insertar un problema y su solución en la base de conocimiento de NovaDesk desde el Sistema Experto.</p>	

VIII. Seleccionar respuestas

Historia de Usuario	
Número: HU_8	Nombre Historia de Usuario: Seleccionar respuestas
Modificación de Historia de Usuario Número: ninguna	
Usuario: Alfredo Pérez Benitez	Iteración Asignada: 4
Prioridad en Negocio: Media	Puntos Estimados: 2
Riesgo en Desarrollo: Medio	Puntos Reales: 2
<p>Descripción: Selección de las preguntas que se le muestran al usuario de acuerdo a la situación que él presente en ese momento al estar trabajando con el sistema operativo Nova.</p>	
<p>Observaciones: La selección de las respuestas por el usuario permite la posteriormente identificar el problema.</p>	
<p>Prototipo de interface:</p> <div style="border: 1px solid gray; padding: 5px; margin: 5px 0;"> <p style="text-align: center; background-color: #d9d9e3; margin: 0;">Seleccione la(s) pregunta(s) para identificar su problema</p> <p><input type="checkbox"/> ¿Deseas comprobar si la imagen de Nova se descargó bien?</p> <p><input type="checkbox"/> ¿Deseas reinstalar Windows sin perder Nova?</p> <p style="text-align: center; margin: 0;"><input type="button" value="Aceptar"/></p> <p><small>Nota: Si su problema no está enmarcado en ninguna de las preguntas anteriores de clic AQUÍ para que lo registre como una incidencia.</small></p> </div>	

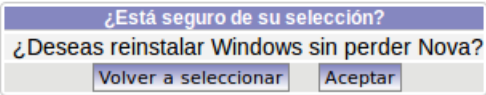
Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 8.1	Número Historia de Usuario: HU_8
Nombre Tarea: Seleccionar respuestas	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 14/02/11	Fecha Fin: 19/02/11
Programador Responsable: Alfredo Pérez Benitez	
<p>Descripción: Implementar la funcionalidad que le permita al usuario seleccionar las respuesta a las preguntas</p>	

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

presentadas para identificar el problema.

IX. Mostrar respuestas

Historia de Usuario	
Número: HU_9	Nombre Historia de Usuario: Mostrar respuestas
Modificación de Historia de Usuario Número: ninguna	
Usuario: Susana Sánchez Ortiz	Iteración Asignada: 4
Prioridad en Negocio: Media	Puntos Estimados: 1
Riesgo en Desarrollo: Medio	Puntos Reales: 1
Descripción: Mostrar las respuestas dadas por el usuario a las preguntas presentadas para que cambie la que desee si se confundió.	
Observaciones: Esto va a permitir que el usuario vea la respuesta que dio y pueda aceptar o cancelar su envío.	
Prototipo de interface:	
	

Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 9.1	Número Historia de Usuario: HU_9
Nombre Tarea: Mostrar respuestas	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 21/02/11	Fecha Fin: 26/02/11
Programador Responsable: Susana Sánchez Ortiz	
Descripción: Mostrar las respuestas que el usuario seleccionó para que pueda aceptar, modificar o cancelar su selección.	

X. Mostrar solución del problema

Historia de Usuario	
Número: HU_10	Nombre Historia de Usuario: Mostrar solución del problema
Modificación de Historia de Usuario Número: ninguna	
Usuario: Alfredo Pérez Benitez	Iteración Asignada: 4
Prioridad en Negocio: Media	Puntos Estimados: 1
Riesgo en Desarrollo: Medio	Puntos Reales: 1

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Descripción: Obtenga la solución al problema. Mediante una búsqueda de la misma en la base de conocimientos de NovaDesk.						
Observaciones: Esta solución va a ser buscada en la base de conocimientos de NovaDesk luego de ser identificado el problema.						
Prototipo de interface: <div style="border: 1px solid gray; padding: 5px; margin-top: 5px;"> <table style="width: 100%; border-collapse: collapse;"> <tr style="background-color: #e6e6fa;"> <th style="text-align: center; padding: 2px;">Solución propuesta a su problema</th> </tr> <tr> <td style="padding: 2px;">Problema: Verificación de la integridad de las imágenes ISO</td> </tr> <tr> <td style="padding: 2px;">Solución:</td> </tr> <tr> <td style="padding: 2px;"> Para comprobar si una imagen .iso de nuestro sistema operativo Nova se descargó correctamente, nos aseguramos de haber descargado el .iso y el .iso.md5, luego teniendo ambos archivos en el mismo directorio, podemos ejecutar lo siguiente en la terminal de comandos, estando ubicados en ese directorio donde se encuentra la imagen: <pre style="font-family: monospace; font-size: 0.9em;">md5sum -c archivo.iso.md5</pre> </td> </tr> <tr> <td style="padding: 2px; text-align: center;"> ¿Desea enviar la solución a su correo electrónico? <input type="button" value="Enviar"/> </td> </tr> <tr> <td style="padding: 2px; text-align: center;"> <input type="button" value="Consultar Sistema Experto"/> </td> </tr> </table> </div>	Solución propuesta a su problema	Problema: Verificación de la integridad de las imágenes ISO	Solución:	Para comprobar si una imagen .iso de nuestro sistema operativo Nova se descargó correctamente, nos aseguramos de haber descargado el .iso y el .iso.md5, luego teniendo ambos archivos en el mismo directorio, podemos ejecutar lo siguiente en la terminal de comandos, estando ubicados en ese directorio donde se encuentra la imagen: <pre style="font-family: monospace; font-size: 0.9em;">md5sum -c archivo.iso.md5</pre>	¿Desea enviar la solución a su correo electrónico? <input type="button" value="Enviar"/>	<input type="button" value="Consultar Sistema Experto"/>
Solución propuesta a su problema						
Problema: Verificación de la integridad de las imágenes ISO						
Solución:						
Para comprobar si una imagen .iso de nuestro sistema operativo Nova se descargó correctamente, nos aseguramos de haber descargado el .iso y el .iso.md5, luego teniendo ambos archivos en el mismo directorio, podemos ejecutar lo siguiente en la terminal de comandos, estando ubicados en ese directorio donde se encuentra la imagen: <pre style="font-family: monospace; font-size: 0.9em;">md5sum -c archivo.iso.md5</pre>						
¿Desea enviar la solución a su correo electrónico? <input type="button" value="Enviar"/>						
<input type="button" value="Consultar Sistema Experto"/>						

Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 10.1	Número Historia de Usuario: HU_10
Nombre Tarea: Mostrar solución del problema	
Tipo de Tarea : Desarrollo	Puntos Estimados: 2
Fecha Inicio: 28/02/11	Fecha Fin: 12/03/11
Programador Responsable: Alfredo Pérez Benitez	
Descripción: Mostrar al usuario la solución del problema identificado.	

XI. *Enviar solución del problema por correo electrónico*

Historia de Usuario	
Número: HU_11	Nombre Historia de Usuario: Enviar solución por correo electrónico
Modificación de Historia de Usuario Número: ninguna	
Usuario: Susana Sánchez Ortiz	Iteración Asignada: 5
Prioridad en Negocio: Baja	Puntos Estimados: 1
Riesgo en Desarrollo: Bajo	Puntos Reales: 1
Descripción: Enviar la solución al problema que presentó el usuario al correo para que la tenga para consultarla posteriormente o cuando se le presente el mismo problema.	
Observaciones: El usuario es el que decide si quiere que le sea enviada la solución a su correo.	

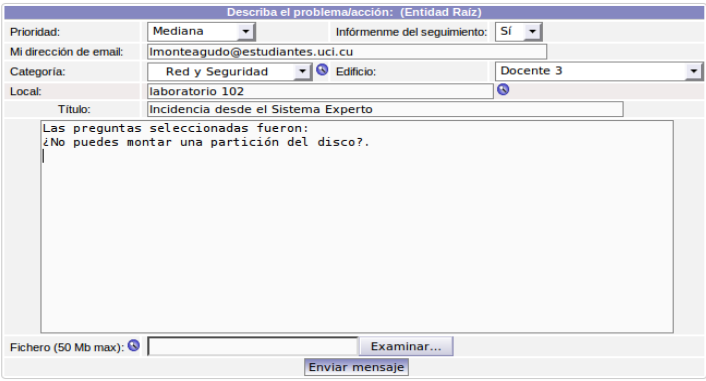
Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 11.1	Número Historia de Usuario: HU_11

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Nombre Tarea: Enviar solución por correo electrónico	
Tipo de Tarea : Desarrollo	Puntos Estimados: 1
Fecha Inicio: 21/03/11	Fecha Fin: 26/03/11
Programador Responsable: Susana Sánchez Ortiz	
Descripción: Enviar la solución del problema identificado por correo electrónico.	

XII. Registrar incidencia

Historia de Usuario	
Número: HU_12	Nombre Historia de Usuario: Registrar incidencia
Modificación de Historia de Usuario Número: ninguna	
Usuario: Alfredo Pérez Benitez	Iteración Asignada: 5
Prioridad en Negocio: Baja	Puntos Estimados: 1
Riesgo en Desarrollo: Bajo	Puntos Reales: 1
Descripción: Si el Sistema Experto no identificó el problema y por consiguiente no pudo darle una solución al usuario, esto sea registrado como una incidencia en el sistema para su posterior solución.	
Observaciones: Esta incidencia se va a mostrar con los datos que proporcionó el usuario al dar respuesta a las preguntas y este va a poder modificarla antes de insertarla, por si desea añadirle algo más.	
Prototipo de interface:	
	

Tareas de ingeniería

Tarea de Ingeniería	
Número Tarea: 12.1	Número Historia de Usuario: HU_12
Nombre Tarea: Registrar incidencia	
Tipo de Tarea: Desarrollo	Puntos Estimados: 1

CAPÍTULO 2: SISTEMA EXPERTO PARA LA GESTIÓN DE LA BASE DE CONOCIMIENTO DE NOVADESK

Fecha Inicio: 28/03/11	Fecha Fin: 02/04/11
Programador Responsable: Alfredo Pérez Benitez	
Descripción: Registrar el problema como una incidencia si este no fue identificado.	

2.6 Plan de release.

Release	Descripción de la iteración	Orden de la HU a implementar	Duración total
Iteración 2	En esta iteración se desarrollarán las historias de usuario que tienen prioridad muy alta.	HU_1, HU_2, HU_3, HU_4	4 semanas
Iteración 3	En esta iteración se desarrollarán las historias de usuario de alta prioridad y se irá integrando con las ya realizadas.	HU_5, HU_6, HU_7	4 semanas
Iteración 4	En esta iteración se desarrollarán las historias de usuario de prioridad media y se integrarán con las historias de usuario ya implementadas.	HU_8, HU_9, HU_10	4 semanas
Iteración 5	En esta iteración se desarrollarán las historias de usuario de baja prioridad y se integrarán con todas las anteriores para conformar el Sistema Experto.	HU_11, HU_12, HU_13, HU_14, HU_15	4 semanas

2.7 Base de datos.

A continuación se presenta el modelo de datos del Sistema Experto. Para la construcción del diagrama entidad relación fue utilizada la herramienta CASE DBDesigner4.0.5.6. Luego se realizaron las transformaciones necesarias para obtener el modelo relacional, y por último se normalizó parcialmente el esquema obtenido hasta la tercera forma normal.

Este diagrama es el correspondiente a las tablas con las que trabaja el Sistema Experto de la base de datos de NovaDesk 2.0. Las tablas son las siguientes: `glsi_entities`, `glsi_kbitems`, `glsi_dropdown_kbcategories`, `glsi_experto_regla`, `glsi_experto_pregunta` y `glsi_experto_categoria`.

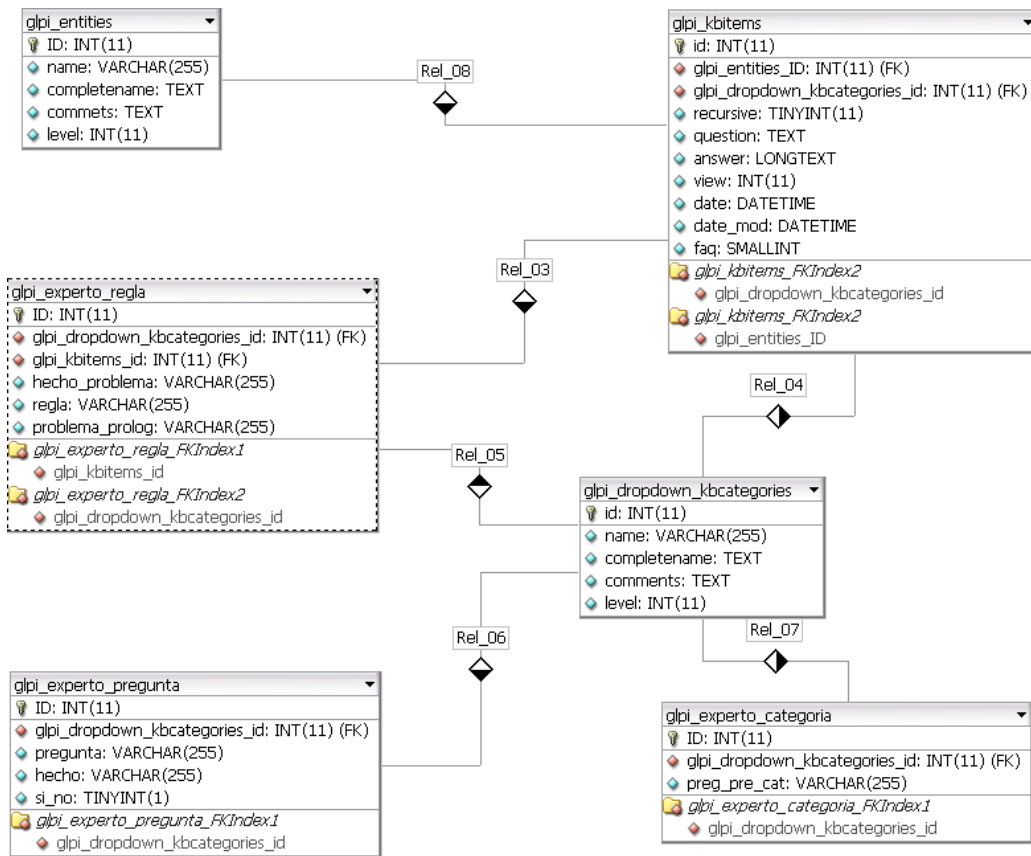


Figura 2.6 Modelo Entidad Relación

2.8 Diseño con metáforas.

Las metáforas conforman el vocabulario para realizar la descripción del problema, conformando el diseño de solución para determinados momentos del proyecto, generando el Modelo de Diseño, el que está integrado por un Diagrama de Paquetes. Este diagrama muestra los elementos físicos del sistema así como las relaciones existentes entre ellos. Muestran las dependencias lógicas entre paquetes de software (componentes, código fuente, librerías, entre otros).

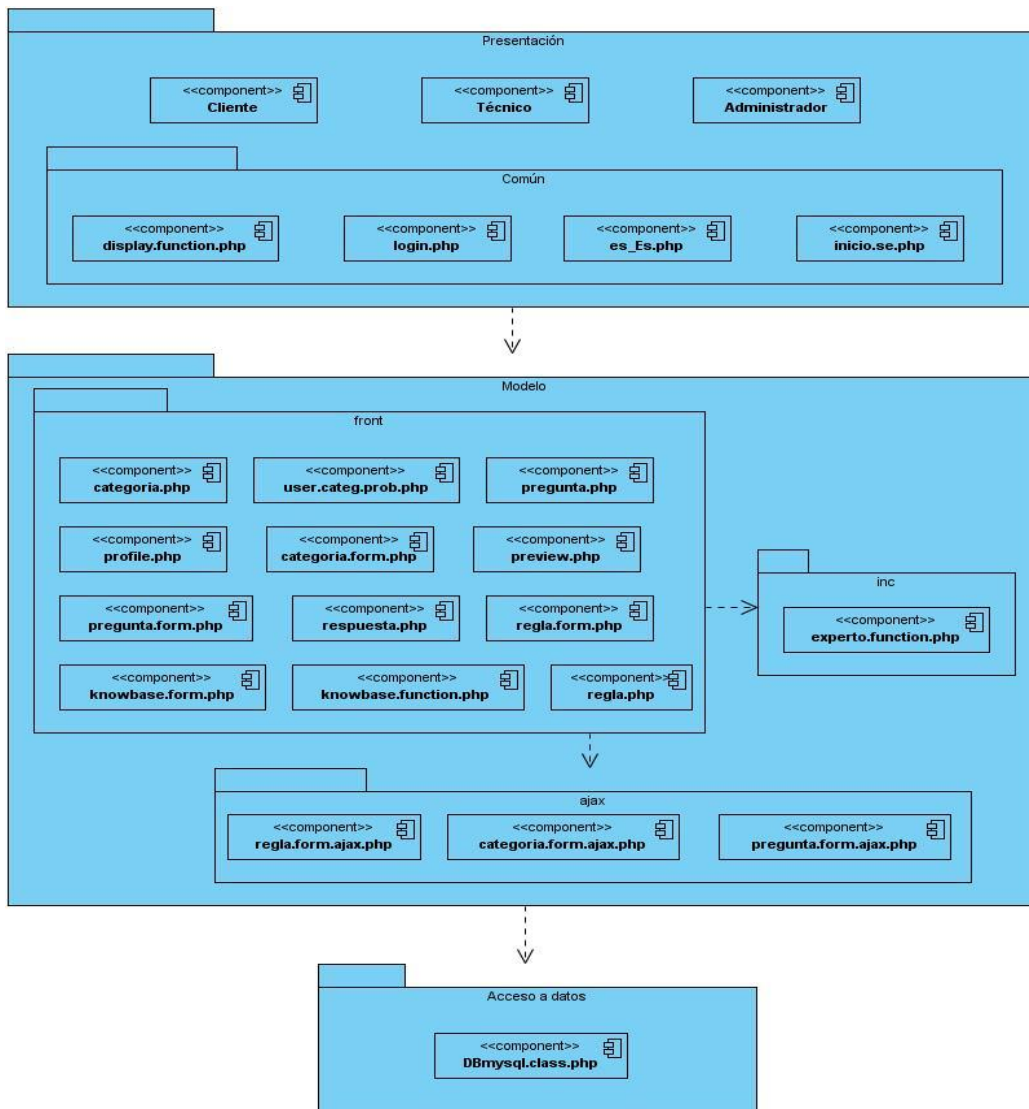


Figura 2.7 Diagrama de paquetes

→ Descripción:

En la **capa de presentación** se muestran los componentes Cliente, Administrador y Técnico los que representan la interfaz de usuario. Tienen relación con el paquete común integrado por `display.function.php` contenedor de las cabeceras y pie de páginas, `login.php` que permite a los usuarios la entrada al sistema, `es_es.php` que permite definir el idioma del mismo e `inicio.se.php` que conforma la interfaz de presentación del Sistema Experto.

La **capa de modelo** integra todas las clases y funcionalidades relacionadas con el negocio que se encuentran divididas en los paquetes principales que integrarán las funcionalidades del Sistema Experto: `front`, `inc` y `ajax`. Esta capa se relaciona con la **capa de datos**, posibilitando el envío de la

información a la base de datos.

2.9 Diagrama de componentes.

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. Muestran las opciones de realización incluyendo código fuente, binario y ejecutable. Los componentes representan todos los tipos de elementos de software que entran en la fabricación de aplicaciones informáticas. Pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, entre otros.

A continuación se presente el diagrama de componentes para el Sistema Experto.

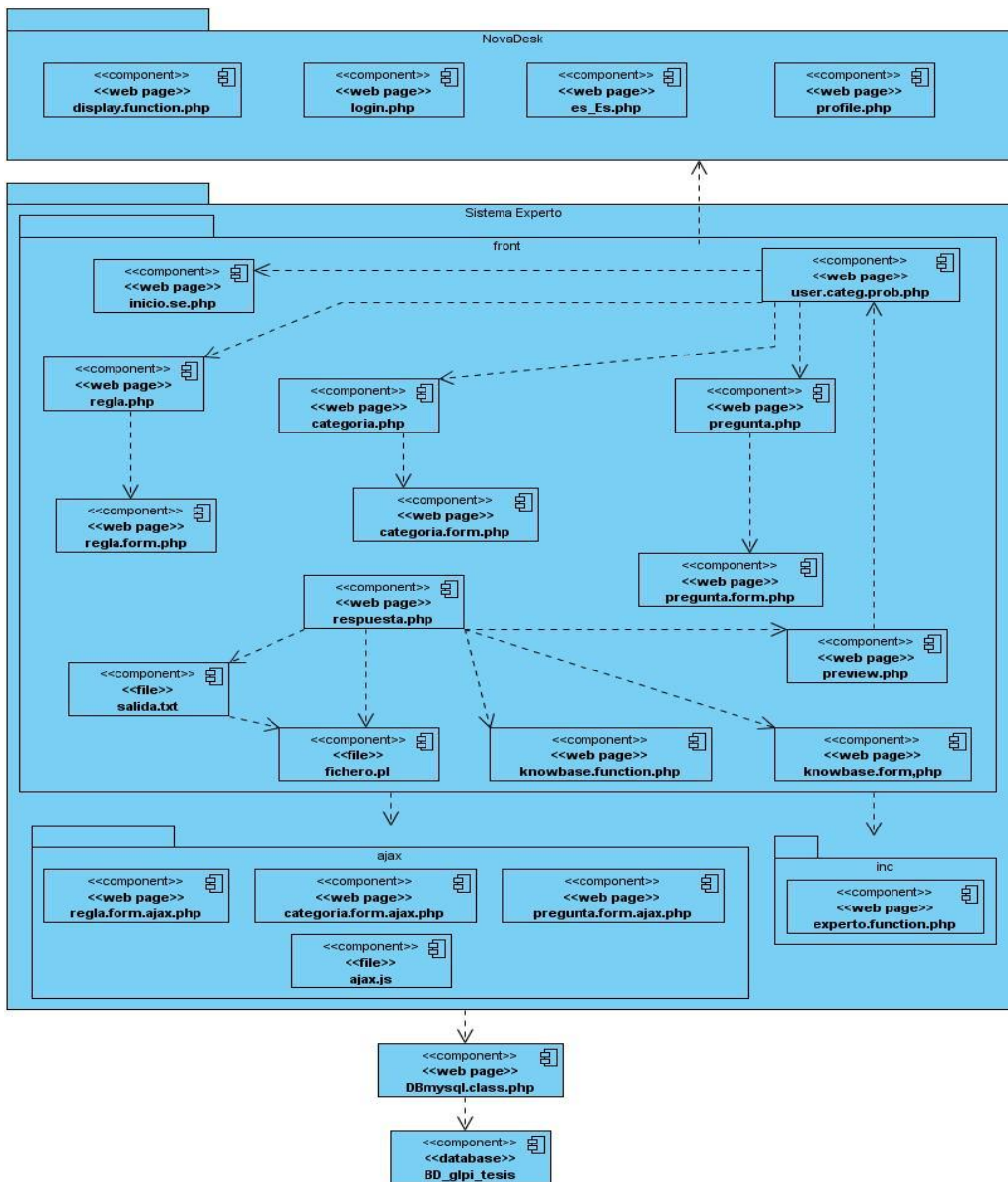


Figura 2.8 Diagrama de componentes


Capítulo 3: “Diseño y realización de pruebas”

En el presente capítulo se muestran las pruebas realizadas a cada una de las historias de usuario correspondientes a cada iteración. Los casos de prueba de aceptación tienen como objetivo validar que el sistema cumpla con el funcionamiento esperado y permitir al cliente determinar su aceptación, desde el punto de vista de su funcionalidad y rendimiento.

3.1 Casos de prueba


Las pruebas de aceptación son definidas por el cliente y preparadas por el equipo de desarrollo, aunque la ejecución y aprobación final corresponden al cliente. La utilización de estas, proporcionan grandes ventajas, permitiendo a los programadores principalmente medir la calidad de su trabajo y garantizar la entrega de un producto con calidad y en correspondencia con las necesidades del cliente. Se definieron casos de prueba para todas las historias de usuario, a continuación se dan a conocer las pruebas que se realizaron a cada una de las historias de usuario con las que cuenta el Sistema Experto para la Gestión de la base de conocimientos de NovaDesk.

❖ Casos de pruebas para la HU Gestionar reglas del Sistema Experto

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-01-1	Nombre Historia de Usuario: Gestionar reglas del Sistema Experto
Nombre de la persona que realiza la prueba: Susana Sánchez Ortiz	
Descripción de la Prueba: Esta prueba consiste en insertar una regla al Sistema Experto.	
Condiciones de Ejecución: En la base de conocimientos debe existir algún problema y el mismo debe poseer preguntas para identificarlo.	
Entrada / Pasos de ejecución: La entrada consiste en seleccionar el problema al que se le va a asociar una regla y las preguntas que permitirán el cumplimiento de la regla, es decir, las preguntas que sea necesario tener para que ocurra ese problema. Una vez dentro del Sistema Experto ejecutar el enlace Reglas SE y aquí dar clic en el botón Añadir Regla o en la imagen  .	
Resultado Esperado: El sistema muestra un mensaje indicando que la regla ha sido añadida correctamente.	
Evaluación de la Prueba: Satisfactoria	

Caso de Prueba de Aceptación

CAPÍTULO 3: DISEÑO Y REALIZACIÓN DE PRUEBAS


Código Caso de Prueba: SEGBCN-01-2	Nombre Historia de Usuario: Gestionar reglas del Sistema Experto
Nombre de la persona que realiza la prueba: Susana Sánchez Ortiz	
Descripción de la Prueba: Esta prueba consiste en eliminar una regla del Sistema Experto.	
Condiciones de Ejecución: En la base de datos debe existir al menos una regla insertada.	
Entrada / Pasos de ejecución: Partiendo del listado de reglas se ejecutará el vínculo  . El sistema muestra el mensaje de confirmación: ¿Está seguro que desea eliminar? Y se da clic en el botón “Aceptar”.	
Resultado Esperado: El sistema elimina automáticamente la regla seleccionada y procede a actualizar el listado de reglas que es mostrado al usuario.	
Evaluación de la Prueba: Satisfactoria	


Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-01-3	Nombre Historia de Usuario: Gestionar reglas del Sistema Experto
Nombre de la persona que realiza la prueba: Susana Sánchez Ortiz	
Descripción de la Prueba: Esta prueba consiste en listar las reglas del Sistema Experto.	
Condiciones de Ejecución: En la base de datos debe existir al menos una regla.	
Entrada / Pasos de ejecución: Una vez dentro del Sistema Experto ejecutar el enlace Reglas SE.	
Resultado Esperado: El sistema muestra el listado con todas las reglas que posee el Sistema Experto.	
Evaluación de la Prueba: Satisfactoria	

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-01-4	Nombre Historia de Usuario: Gestionar reglas del Sistema Experto
Nombre de la persona que realiza la prueba: Susana Sánchez Ortiz	
Descripción de la Prueba: Esta prueba consiste en modificar una regla del Sistema Experto.	
Condiciones de Ejecución: En la base de datos debe existir al menos una regla insertada.	
Entrada / Pasos de ejecución: Partiendo del listado de reglas se ejecutará el vínculo “Editar?”. El sistema muestra la regla seleccionada en el formulario correspondiente. Se realizan los cambios y se selecciona “Modificar Regla”.	
Resultado Esperado: El sistema muestra un mensaje mostrando el éxito de la acción y actualiza el listado de reglas mostrado al usuario.	

Evaluación de la Prueba: Satisfactoria

❖ **Casos de pruebas para la HU Gestionar preguntas del Sistema Experto**

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-02-1	Nombre Historia de Usuario: Gestionar preguntas del Sistema Experto
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	
Descripción de la Prueba: Esta prueba consiste en insertar una pregunta al Sistema Experto para identificar el problema.	
Condiciones de Ejecución: En la base de conocimientos deben estar definidas las categorías de los problemas con al menos una pregunta asociada.	
Entrada / Pasos de ejecución: La entrada consiste en elaborar la pregunta, seleccionar la categoría de esta pregunta y definir el hecho al que va a responder la misma. Una vez dentro del Sistema Experto ejecutar el enlace Pregunta y aquí dar clic en el botón Añadir Pregunta o en la imagen  .	
Resultado Esperado: El sistema muestra un mensaje indicando que la pregunta ha sido añadida correctamente.	
Evaluación de la Prueba: Satisfactoria	

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-02-2	Nombre Historia de Usuario: Gestionar preguntas del Sistema Experto
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	
Descripción de la Prueba: Esta prueba consiste en eliminar una pregunta del Sistema Experto.	
Condiciones de Ejecución: En la base de datos debe existir al menos una pregunta insertada.	
Entrada / Pasos de ejecución: Partiendo del listado de preguntas se ejecutará el vínculo  . El sistema muestra el mensaje de confirmación: ¿Está seguro que desea eliminar? Y se da clic en el botón "Aceptar".	
Resultado Esperado: El sistema elimina automáticamente la pregunta seleccionada y procede a actualizar el listado de preguntas que es mostrado al usuario.	
Evaluación de la Prueba: Satisfactoria	

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-02-3	Nombre Historia de Usuario: Gestionar preguntas del Sistema Experto
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	

CAPÍTULO 3: DISEÑO Y REALIZACIÓN DE PRUEBAS

Descripción de la Prueba: Esta prueba consiste en listar las preguntas del Sistema Experto para identificar el problema.
Condiciones de Ejecución: En la base de datos debe existir al menos una pregunta.
Entrada / Pasos de ejecución: Una vez dentro del Sistema Experto ejecutar el enlace Preguntas.
Resultado Esperado: El sistema muestra el listado con todas las preguntas que posee el Sistema Experto.
Evaluación de la Prueba: Satisfactoria

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-02-4	Nombre Historia de Usuario: Gestionar preguntas del Sistema Experto
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	
Descripción de la Prueba: Esta prueba consiste en modificar una pregunta del Sistema Experto.	
Condiciones de Ejecución: En la base de datos debe existir al menos una pregunta insertada.	
Entrada / Pasos de ejecución: Partiendo del listado de preguntas se ejecutará el vínculo “Editar?”. El sistema muestra la pregunta seleccionada en el formulario correspondiente. Se realizan los cambios y se selecciona “Modificar pregunta”.	
Resultado Esperado: El sistema muestra un mensaje mostrando el éxito de la acción y actualiza el listado de preguntas mostrado al usuario.	
Evaluación de la Prueba: Satisfactoria	


❖ Casos de pruebas para la HU Identificar la categoría del problema


Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-03-1	Nombre Historia de Usuario: Identificar la categoría del problema
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	
Descripción de la Prueba: Esta prueba consiste en identificar la categoría del problema.	
Condiciones de Ejecución: En la base de datos deben existir al menos una categoría con su pregunta asociada.	
Entrada / Pasos de ejecución: La entrada consiste en seleccionar una pregunta para identificar la categoría del problema que presenta el usuario. Si es administrador, una vez dentro del Sistema Experto ejecutar el enlace Consultar SE, aquí dar clic en el vínculo “Consultar Sistema Experto” y seleccionar una de las preguntas que se muestran. Si es un usuario, ir al vínculo “Sistema Experto”, aquí dar clic en el vínculo “Consultar Sistema Experto” y seleccionar una de las preguntas que se muestran.	
Resultado Esperado: El sistema muestra las preguntas para identificar el problema que pertenece a esa categoría.	

CAPÍTULO 3: DISEÑO Y REALIZACIÓN DE PRUEBAS

Evaluación de la Prueba: Satisfactoria

❖ Casos de pruebas para la HU Gestionar categorías de clasificación de problemas

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-04-1	Nombre Historia de Usuario: Gestionar las preguntas para identificar las categorías de clasificación de problemas.
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	
Descripción de la Prueba: Esta prueba consiste en insertar una pregunta para identificar las categorías de clasificación de problemas.	
Condiciones de Ejecución: En la base de conocimientos deben estar definidas las categorías de los problemas.	
Entrada / Pasos de ejecución: La entrada consiste en seleccionar la categoría y elaborar la pregunta que la va a definir. Una vez dentro del Sistema Experto ejecutar el enlace Categoría y aquí dar clic en el botón Añadir Categoría o en la imagen  .	
Resultado Esperado: El sistema muestra un mensaje indicando que la categoría ha sido añadida correctamente.	
Evaluación de la Prueba: Satisfactoria	

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-04-2	Nombre Historia de Usuario: Gestionar las preguntas para identificar las categorías de clasificación de problemas.
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	
Descripción de la Prueba: Esta prueba consiste en eliminar una pregunta para identificar las categorías de clasificación de problemas del Sistema Experto.	
Condiciones de Ejecución: En la base de datos debe existir al menos una pregunta asociada a alguna categoría.	
Entrada / Pasos de ejecución: Partiendo del listado de categorías se ejecutará el vínculo  . El sistema muestra el mensaje de confirmación: ¿Está seguro que desea eliminar? Y se da clic en el botón "Aceptar".	
Resultado Esperado: El sistema elimina automáticamente la categoría seleccionada y procede a actualizar el listado de categorías que es mostrado al usuario.	
Evaluación de la Prueba: Satisfactoria	

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-04-3	Nombre Historia de Usuario: Gestionar las preguntas para identificar las categorías de clasificación de problemas.
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	

CAPÍTULO 3: DISEÑO Y REALIZACIÓN DE PRUEBAS

Descripción de la Prueba: Esta prueba consiste en listar las preguntas para identificar las categorías de clasificación de problemas del Sistema Experto.
Condiciones de Ejecución: En la base de datos debe existir al menos una pregunta asociada a alguna categoría.
Entrada / Pasos de ejecución: Una vez dentro del Sistema Experto ejecutar el enlace Categorías.
Resultado Esperado: El sistema muestra el listado con todas las preguntas y categorías asociadas que posee el Sistema Experto.
Evaluación de la Prueba: Satisfactoria

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-04-4	Nombre Historia de Usuario: Gestionar las preguntas para identificar las categorías de clasificación de problemas.
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	
Descripción de la Prueba: Esta prueba consiste en modificar una pregunta asociada a las categorías de clasificación de problemas.	
Condiciones de Ejecución: En la base de datos debe existir al menos una pregunta asociada a alguna categoría.	
Entrada / Pasos de ejecución: Partiendo del listado de categorías se ejecutará el vinculo "Editar?". El sistema muestra la categoría con su pregunta asociada seleccionada en el formulario correspondiente. Se realizan los cambios y se selecciona "Modificar pregunta".	
Resultado Esperado: El sistema muestra un mensaje mostrando el éxito de la acción y actualiza el listado de las categorías con sus preguntas asociadas mostrado al usuario.	
Evaluación de la Prueba: Satisfactoria	

❖ Casos de pruebas para la HU Identificar problemas

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-05-1	Nombre Historia de Usuario: Identificar problema
Nombre de la persona que realiza la prueba: Susana Sánchez Ortiz	
Descripción de la Prueba: Esta prueba consiste en identificar el problema que presenta el usuario.	
Condiciones de Ejecución: En la base de datos deben existir al menos una categoría con su pregunta asociada, una o varias preguntas para identificar el problema y una o varias reglas asociadas a alguna categoría.	
Entrada / Pasos de ejecución: La entrada consiste en el resultado satisfactorio de la HU_9 y de las reglas asociadas a la categoría del problema presentado.	
Resultado Esperado: El sistema muestra el problema encontrado.	
Evaluación de la Prueba: Satisfactoria	

❖ Casos de pruebas para la HU Buscar solución del problema

CAPÍTULO 3: DISEÑO Y REALIZACIÓN DE PRUEBAS

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-06-1	Nombre Historia de Usuario: Buscar solución del problema
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	
Descripción de la Prueba: Esta prueba consiste en buscar la solución del problema que presenta el usuario.	
Condiciones de Ejecución: En la base de conocimientos debe existir la solución al problema encontrado.	
Entrada / Pasos de ejecución: La entrada consiste en el resultado satisfactorio de la HU_5.	
Resultado Esperado: El sistema muestra la solución del problema encontrado.	
Evaluación de la Prueba: Satisfactoria	

❖ Casos de pruebas para la HU Llenar la base de conocimientos

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-07-1	Nombre Historia de Usuario: Llenar la base de conocimientos
Nombre de la persona que realiza la prueba: Susana Sánchez Ortiz	
Descripción de la Prueba: Esta prueba consiste en Llenar la Base de Conocimientos de NovaDesk desde el Sistema Experto.	
Condiciones de Ejecución: En la base de datos debe existir alguna categoría.	
Entrada / Pasos de ejecución: La entrada consiste en seleccionar la categoría, elaborar el tema del problema a insertar y su contenido. Una vez dentro del Sistema Experto ejecutar el enlace Llenar base de conocimiento.	
Resultado Esperado: El sistema muestra el problema insertado en la base de conocimientos.	
Evaluación de la Prueba: Satisfactoria	

❖ Casos de pruebas para la HU Seleccionar respuestas

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-08-1	Nombre Historia de Usuario: Seleccionar respuestas
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	
Descripción de la Prueba: Esta prueba consiste en permitirle al usuario que seleccione las preguntas que identifican su problema.	
Condiciones de Ejecución: En la base de datos deben existir al menos una categoría con su pregunta asociada y una o varias preguntas.	
Entrada / Pasos de ejecución: La entrada consiste en seleccionar la(s) pregunta(s) relacionada(s) al problema que se presente. Luego de ejecutar satisfactoriamente la HU_3 se seleccionan una o varias preguntas relacionadas al problema que se presenta.	

Resultado Esperado: El sistema muestra las respuestas seleccionadas.
Evaluación de la Prueba: Satisfactoria

❖ **Casos de pruebas para la HU Mostrar respuestas**

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-09-1	Nombre Historia de Usuario: Mostrar respuestas
Nombre de la persona que realiza la prueba: Susana Sánchez Ortiz	
Descripción de la Prueba: Esta prueba consiste mostrarle al usuario las preguntas seleccionadas.	
Condiciones de Ejecución: El usuario debe haber seleccionado alguna pregunta.	
Entrada / Pasos de ejecución: Luego de ejecutar satisfactoriamente la HU_8 se da clic en el botón "Enviar".	
Resultado Esperado: El sistema muestra la solución propuesta al problema presentado.	
Evaluación de la Prueba: Satisfactoria	

❖ **Casos de pruebas para la HU Mostrar solución del problema**

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-10-1	Nombre Historia de Usuario: Mostrar solución del problema
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	
Descripción de la Prueba: Esta prueba consiste en mostrar la solución del problema que presenta el usuario.	
Condiciones de Ejecución: En la base de conocimientos debe existir la solución al problema encontrado.	
Entrada / Pasos de ejecución: La entrada consiste en el resultado satisfactorio de la HU_6.	
Resultado Esperado: El sistema muestra el problema encontrado con su solución.	
Evaluación de la Prueba: Satisfactoria	

❖ **Casos de pruebas para la HU Enviar solución del problema por correo electrónico**

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-11-1	Nombre Historia de Usuario: Enviar solución por correo electrónico
Nombre de la persona que realiza la prueba: Susana Sánchez Ortiz	
Descripción de la Prueba: Esta prueba consiste en enviar la solución encontrada por el Sistema Experto al correo electrónico del usuario.	
Condiciones de Ejecución: Debe de haberse encontrado el problema que presenta el usuario y su solución.	
Entrada / Pasos de ejecución: La entrada consiste en el resultado satisfactorio de la HU_10. Luego de ejecutar satisfactoriamente la HU_10 se da clic en el botón "Enviar".	
Resultado Esperado: El sistema muestra un mensaje indicando que el correo ha sido enviado correctamente.	

Evaluación de la Prueba: Satisfactoria

❖ **Casos de pruebas para la HU Registrar incidencia**

Caso de Prueba de Aceptación	
Código Caso de Prueba: SEGBCN-12-1	Nombre Historia de Usuario: Registrar incidencia
Nombre de la persona que realiza la prueba: Alfredo Pérez Benitez	
Descripción de la Prueba: Esta prueba consiste en que el usuario pueda registrar una incidencia cuando el Sistema Experto no identifique el problema presentado.	
Condiciones de Ejecución: No se ha encontrado el problema que presenta el usuario.	
Entrada / Pasos de ejecución: La entrada consiste en los elementos del problema que haya dado el usuario durante la consulta del Sistema Experto y los restantes que desee añadir del formulario para registrar una incidencia. Luego de la ejecución no satisfactoria de la HU_6, se muestra el formulario para registrar una incidencia y se da clic en el botón "Aceptar".	
Resultado Esperado: El sistema muestra un mensaje indicando que la incidencia ha sido insertada correctamente.	
Evaluación de la Prueba: Satisfactoria	

Capítulo 4: “Análisis del comportamiento del Sistema Experto”

Evaluar globalmente el comportamiento de un Sistema Experto es una tarea compleja, por esta razón está compuesta por diferentes fases o etapas. Estas fases son: verificación, validación y evaluación. Todas estas fases se organizan jerárquicamente como se ilustra en la Figura 4.1.

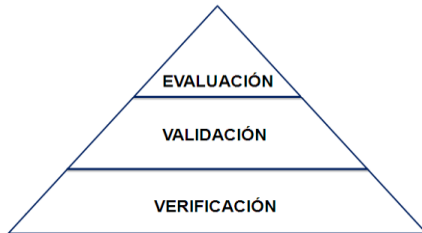


Figura 4.1 Pirámide del análisis del comportamiento de un sistema inteligente.

La verificación y la validación (V&V) son dos de las etapas más importantes en el análisis del comportamiento de un Sistema Experto. La verificación permite comprobar si se ha construido el sistema correctamente, es decir, asegurarse de que el software implementado no contiene errores, y que el producto final satisface los requisitos y las especificaciones de diseño. Por otra parte, el término validación se refiere, a un análisis de la calidad del Sistema Experto en su entorno real de trabajo, lo que permite determinar si el producto desarrollado satisface convenientemente las expectativas inicialmente depositadas. La evaluación se encarga de analizar aspectos que van más allá de la corrección de las soluciones finales del sistema: analiza aspectos como utilidad, robustez, velocidad, eficiencia, posibilidades de ampliación, facilidad de manejo, análisis coste/beneficio, entre otros.

4.1. Verificación del Sistema Experto

La verificación de un Sistema Experto es un proceso que incluye las siguientes tareas: Verificación del cumplimiento de las especificaciones; Verificación de los mecanismos de inferencia y Verificación de la base de conocimientos.

4.1.1 Verificación del cumplimiento de las especificaciones

El análisis del cumplimiento de las especificaciones puede ser llevado a cabo por los desarrolladores, los usuarios, los expertos y/o un grupo de evaluadores independientes.

En este aspecto el Sistema Experto implementado cumple con todos los parámetros que se plantean.

- ❖ El diseño y la implementación han sido llevados a cabo modularmente, porque se elaboró un cronograma por el cual se realizaron todas estas actividades y fueron hechas en un orden de prioridad y complejidad.

- ❖ La interfaz de usuario cumple las especificaciones previstas y está acorde al diseño del sistema NovaDesk, del cual forma parte.
- ❖ Las facilidades de explicación son apropiadas para los potenciales usuarios del sistema. Pues la Base de Conocimientos que posee NovaDesk presenta problemas comunes con sus soluciones explicadas para que los usuarios puedan entenderlas y aplicarlas.
- ❖ El mantenimiento del sistema es posible hasta el grado especificado. Pues este cuenta con un rol de administrador que se encarga de actualizarlo, ampliarlo y gestionarlo.
- ❖ El sistema cumple las especificaciones de seguridad. La base de conocimientos está protegida ante modificaciones realizadas por personal no autorizado. Pues NovaDesk presenta varios roles y niveles de acceso que protegen la Base de Conocimientos y el Sistema Experto.

4.1.2 Verificación de los mecanismos de inferencia

En este aspecto el motor de inferencia utilizado por el Sistema Experto fue el de la herramienta Swi-prolog, por lo que no es necesario que se verifiquen los mecanismos de inferencia del Sistema Experto porque Prolog implícitamente lo realiza como se explica en el epígrafe 1.12 del capítulo 1 que lleva por título: Lenguajes utilizados en la construcción de Sistemas Expertos.

4.1.3 Verificación de la base de conocimientos

La verificación de la base de conocimientos se basa, generalmente, en el concepto de anomalías; una anomalía es un uso poco común del esquema de representación del conocimiento, que puede ser considerado como un error potencial (existen anomalías que no constituyen errores y viceversa).

La verificación de la base de conocimientos no asegura que las respuestas del sistema sean correctas, lo que asegura es que el sistema ha sido diseñado e implementado de forma correcta. Los aspectos que se suelen examinar a la hora de verificar una base de conocimientos son la consistencia y la completitud. En la Tabla 4.1 se muestran una serie de pruebas que se realizan para comprobar que la base de conocimientos es consistente y completa.

Consistencia	Reglas redundantes	$p(x) \wedge q(x) \rightarrow r(x)$ $q(x) \wedge p(x) \rightarrow r(x)$
	Reglas conflictivas	$p(x) \wedge q(x) \rightarrow r(x)$ $p(x) \wedge q(x) \rightarrow \neg r(x)$
	Reglas englobadas en otras	$p(x) \wedge q(x) \rightarrow r(x)$ $p(x) \rightarrow r(x)$
	Reglas circulares	$p(x) \rightarrow q(x)$ $q(x) \rightarrow r(x)$ $r(x) \rightarrow p(x)$
	Condiciones IF innecesarias	$p(x) \wedge q(x) \rightarrow r(x)$

CAPÍTULO 4: ANÁLISIS DEL COMPORTAMIENTO DEL SISTEMA EXPERTO.

		$p(x) \wedge \neg q(x) \rightarrow r(x)$
Complejidad	Valores no referenciados de atributos	Ocurre cuando algunos valores, del conjunto de posibles valores de un atributo, no son cubiertos por la premisa de ninguna regla.
	Valores ilegales de atributos	Una regla referencia valores de atributos que no están incluidos en el conjunto de valores válidos para ese atributo.
	Reglas inalcanzables	$p(x) \rightarrow r(x)$ $p(x)$ no aparece como conclusión de otra regla ni puede obtenerse del exterior (razonamiento progresivo)
	Reglas sin salida	$p(x) \wedge q(x) \rightarrow r(x)$ $r(x)$ no es una conclusión final y no aparece en la premisa de ninguna regla (razonamiento progresivo)

Tabla 4.1 Verificación de la consistencia y la completitud en bases de conocimientos [34].

En la revisión realizada a las reglas del Sistema Experto implementado, en cuanto a la consistencia del mismo, se determinó que no existían reglas redundantes, ni conflictivas, ni englobadas en otras reglas, ni circulares, ni condiciones IF innecesarias. Además en el momento de insertar una regla en el Sistema Experto se realiza una validación que verifica que al sistema no se le inserten reglas que no cumplan con alguno de estos aspectos mencionados anteriormente. También el sistema no posee reglas negativas.

En cuanto a la completitud de las reglas, el Sistema Experto implementado no presenta valores no referenciados, ni valores ilegales de atributos, ni reglas inalcanzables y sin salida. Lo anterior se demuestra pues las reglas del Sistema Experto para ser insertadas dependen de que existan problemas con sus soluciones en la base de conocimientos de NovaDesk, de lo contrario no van a ser insertadas.

4.2. Validación del Sistema Experto

Una vez verificado el Sistema Experto, el proceso debe continuar con la validación del mismo. Validar un Sistema Experto supone analizar si los resultados del sistema son correctos y si se cumplen las necesidades y los requisitos del usuario.

La validación puede verse desde dos ópticas diferentes: por un lado una *validación orientada a los resultados*, cuyo objetivo es comparar el rendimiento del sistema con un rendimiento esperado (proporcionado por una referencia estándar o por expertos humanos), y comprobar que el sistema alcanza un nivel que se considera aceptable. Por otro lado se tiene la *validación orientada al uso*, que se centra en cuestiones que hacen referencia a la relación hombre-máquina, más allá de la corrección de los resultados obtenidos por el sistema.

Normalmente la validación orientada a los resultados constituye un prerrequisito para la realización de una validación orientada al uso. Así, si un sistema no presenta un rendimiento aceptable, los aspectos concernientes a la validación orientada al uso son irrelevantes. Por este motivo muchos autores incluyen la validación orientada al uso como una de las primeras fases de la evaluación, pasando el término “validación” a referirse únicamente a la validación orientada a los resultados. [35]

4.2.1 Principales características del proceso de validación

Una cuestión importante a determinar en todo tipo de validación es quién va a llevarla a cabo como se muestra en la Figura 4.2. En la validación siempre es necesaria la presencia de una persona que tenga un conocimiento amplio del sistema. También es necesario contar con expertos humanos. El Sistema Experto implementado se validará por los desarrolladores del mismo y los expertos en el Sistema Operativo Nova.

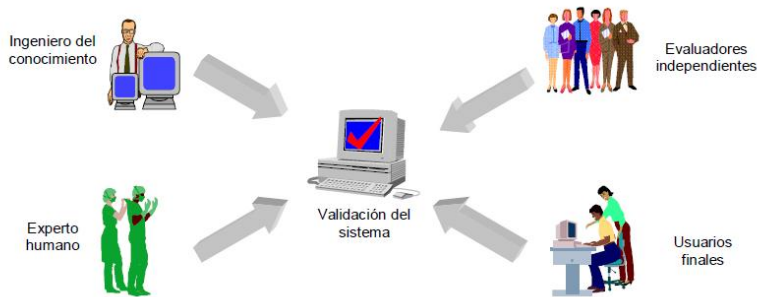


Figura 4.2. Personal involucrado en la validación de un sistema inteligente [36].

Existen diferentes criterios para validar un Sistema Experto, por lo que se pueden diferenciar dos tipos de validación atendiendo al tipo de criterio establecido: *validación contra el experto* y *validación contra el problema*.

La *validación contra el problema* consiste en descubrir si el sistema resuelve realmente el problema que le han planteado. Este tipo de validación podría traer consigo la desventaja de exigirle más al Sistema Experto de lo que se le exigiría a un experto humano y además puede no ser posible obtener una solución real. Por estas razones se recomienda la validación contra los expertos. La *validación contra el experto* consiste, básicamente, en utilizar las opiniones y los diagnósticos de expertos humanos como criterio de validación. Este tipo de validación es la más comúnmente empleada en los sistemas inteligentes, porque lo que se pretende es construir un modelo del conocimiento del experto humano, por lo que resulta lógico utilizar a los expertos como criterio de validación.

Existen tres posibles tipos de validación contra los expertos: *validación contra un experto*, *validación contra un grupo de expertos* y *validación contra un consenso de expertos*. La *validación contra un único experto* no es la más recomendable de todas. El inconveniente de utilizar un único experto es que la

objetividad del estudio es cuestionable. Lo más recomendable a la hora de realizar la validación es contar con las opiniones de una serie de expertos humanos. Esto conlleva una serie de ventajas: no se posee una única opinión, que puede ser errónea; permite comparar el grado de consistencia existente entre los expertos del dominio.

La otra opción comúnmente empleada en la validación con expertos, es conseguir unir las opiniones de varios expertos en una única opinión. Este consenso tiene la ventaja de procurar ser lo más objetivo posible, y si el acuerdo del sistema inteligente con el consenso es amplio, la confianza en el sistema aumentará considerablemente. El inconveniente de esta técnica es que, en cierta manera, estamos volviendo a la técnica de validación con un único experto, es decir, todo aquello que cae fuera del consenso es considerado erróneo.

Por lo planteado anteriormente la validación del Sistema Experto se realizará contra un grupo de expertos. Un aspecto a tener en cuenta es el momento de realizar la validación, esta se puede realizar desde etapas tempranas del desarrollo del Sistema Experto, durante su desarrollo o al concluir el mismo. El Sistema Experto implementado se validará luego de concluir su implementación y antes de que se comience a utilizar por los usuarios finales.

La validación de un Sistema Experto se realiza utilizando métodos que pueden dividirse en dos grupos principales: *métodos cualitativos* y *métodos cuantitativos*. Los *métodos cualitativos* emplean técnicas subjetivas de comparación del rendimiento mientras que los *métodos cuantitativos* se basan en la utilización de medidas estadísticas. Dentro de los *métodos cualitativos* de validación se pueden destacar los siguientes: la validación de superficie, la prueba de Turing, los tests de campo, la validación de subsistemas, el análisis de sensibilidad y los grupos de control. A diferencia de los *métodos cualitativos*, los *métodos cuantitativos* de validación están basados en análisis estadísticos, que tratan de comparar las conclusiones del sistema inteligente con las producidas por los expertos del dominio. Existen varias técnicas estadísticas que pueden ser utilizadas en el proceso de validación, en este trabajo se expondrán las más usadas que se dividen en tres grupos: *medidas de pares*, *medidas de grupo* y *ratios de acuerdo*.

Para la validación del Sistema Experto implementado se utilizará el *método cuantitativo* pues este está basado en análisis estadísticos como se explicaba anteriormente y se considera fundamental para analizar e interpretar la información de forma acertada y confiable. A continuación se explican las diferentes partes en las que se subdividen los *métodos cuantitativos*.

Las *medidas de pares* son útiles cuando el número de expertos es reducido. Consisten en comparar las interpretaciones del sistema con las interpretaciones de un único experto humano. Por esta razón, si la validación involucra un grupo amplio de expertos la información que proporcionan las medidas de pares puede resultar difícil de interpretar.

Las *medidas de grupo*, tienen como objetivo analizar conjuntamente las interpretaciones de los expertos

y tratar de buscar estructuras de representación que permitan una interpretación más sencilla dentro del contexto de la validación. Los *ratios de acuerdo* se encargan de medir el acuerdo existente entre un experto y una referencia estándar. Dicha referencia puede ser un consenso existente entre los expertos, lo que implicaría una validación contra el experto, o la solución real al problema planteado, lo que se convertiría en una validación contra el problema.

De estos tres métodos las *medidas de pares* no pueden ser aplicadas a la validación del Sistema Experto implementado pues se cuenta con más de un experto para realizar la validación y la información que brindaría sería difícil de interpretar. De los otros dos métodos *los ratios de acuerdo* analizan los resultados obtenidos en las distintas categorías en las que se divide la interpretación. Además, en dependencia a la referencia que se seleccione se pueden convertir en una validación contra el problema y esta no es la que se va a realizar; por lo que el método cuantitativo a utilizar es el de las *medidas de grupos*.

❖ **Medidas de grupo**

Las medidas de pares sirven de base a las medidas de grupo, cuyo objetivo es dar una medida que indique el grado de consenso entre los expertos humanos y comprobar si el Sistema Experto se comporta como un experto humano más, pudiendo ser incorporado al grupo de expertos sin empeorar el grado de consenso. El procedimiento para obtener las medidas de grupo es el siguiente, como se muestra en la Figura 4.3: (1) se obtienen las medidas de pares para cada uno de los posibles pares de expertos de la validación, (2) se agrupan los resultados de cada medida de pares en una tabla resumen y, (3) se obtiene la medida de grupo a partir de los datos incluidos en las tablas resumen.

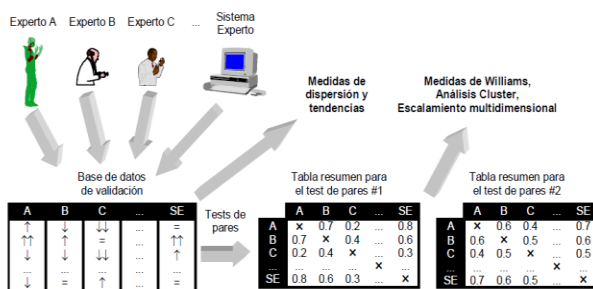


Figura 4.3 Proceso de realización del método medidas de grupo [37].

Los pasos para realizar la validación del Sistema Experto implementado se describen a continuación:

1. *Creación de una base de datos de validación.* Los expertos del dominio, así como los desarrolladores del Sistema Experto realizan una clasificación de la dificultad de los desafíos.
2. *Creación de las tablas de contingencia para los posibles pares de expertos.* Para cada emparejamiento posible de expertos se crea una tabla de contingencia que muestra los acuerdos y desacuerdos en la clasificación entre los pares de expertos.

3. *Cálculos de las medidas de pares.* A partir de las tablas de contingencia se calculan las medidas de pares, que indican el grado de acuerdo entre dos expertos, dando lugar a la tabla de pares que recopila las medidas de pares de todos los expertos.
4. *Cálculo de la medida de grupo a partir de las tablas de pares.* A partir de los datos de las tablas de pares se calcula la medida de grupo, que resume la información contenida en las tablas de pares y da una medida del grado de consenso dentro de un grupo.
5. Con el objetivo final de probar si el Sistema Experto puede “pasar” como un experto humano, se determina si el Sistema Experto está de acuerdo con los expertos del grupo de referencia en el mismo grado que los expertos están de acuerdo entre ellos mismos.

En la Figura 4.4 se muestra un esquema de tabla de contingencia. Cada celda tiene un valor n_{ij} que corresponde al número de veces que el experto A y el experto B han clasificado un caso como clase j y clase i , respectivamente.

		Experto A				
		Clase 1	Clase 2	...	Clase m	Total
Experto B	Clase 1	n_{11}	n_{12}	...	n_{1m}	$n_{1.} = \sum_{j=1}^m n_{1j}$
	Clase 2	n_{21}	n_{22}	...	n_{2m}	$n_{2.} = \sum_{j=1}^m n_{2j}$

	Clase m	n_{m1}	n_{m2}	...	n_{mm}	$n_{m.} = \sum_{j=1}^m n_{mj}$
	Total	$n_{.1} = \sum_{i=1}^m n_{i1}$	$n_{.2} = \sum_{i=1}^m n_{i2}$...	$n_{.m} = \sum_{i=1}^m n_{im}$	N

Figura 4.4. Tabla de contingencia

Las medidas de pares se dividen en dos grupos: *medidas de acuerdo* y *medidas de asociación*.

Las *medidas de acuerdo* dan un índice que cuantifica las coincidencias entre las interpretaciones de dos expertos. Las *medidas de asociación* miden el grado de asociación lineal existente entre el sistema y el experto humano. Estas medidas sólo son aplicables en el caso de que las categorías semánticas de la interpretación sigan una escala ordinal, debido a que presuponen la existencia de un orden.

De estos dos grupos la validación del Sistema Experto implementado se realizará utilizando las *medidas de acuerdo*, pues se quiere conocer el grado de acuerdo que existe entre el Sistema Experto y los expertos humanos y para esto existen medidas para determinarlo entre las que se pueden destacar: el índice de acuerdo, el índice de acuerdo dentro de uno, kappa, y kappa ponderada como se mencionaban anteriormente. Cada una posee sus características específicas, sus ventajas y sus desventajas y luego de analizarlas todas se utilizará: el índice de acuerdo por ser una medida muy popular pues la sencillez de su interpretación ha hecho que su uso se extienda en distintos campos y aplicaciones.

El índice de acuerdo es el cociente entre el número de observaciones de acuerdo y el número de observaciones totales, como se muestra en la ecuación (Figura 4.5) en la que N es el número total de

casos, k el número total de categorías semánticas y n_{ij} el número total de casos en la celda ij de la tabla de contingencia.

$$\text{Índice de acuerdo} = \frac{\sum_{i=j}^k n_{ij}}{N} = \sum_{i=j}^k P_{ij}$$

Figura 4.5 Ecuación para calcular el índice de acuerdo.

Una vez se tienen las medidas de pares, se procede a calcular la medida de grupo con el objetivo de conocer el nivel de acuerdo que existe entre los expertos del grupo. Dentro de las medidas de grupo se encuentran: (a) el índice de Williams, (b) el análisis cluster, (c) el escalamiento multidimensional y (d) las medidas de dispersión y tendencia. Cada una posee sus características específicas, sus ventajas y sus desventajas y luego de analizarlas todas se utilizará el índice de Williams.

El **índice de Williams** se representa en la Figura 4.6. En esta expresión P_0 representa el acuerdo existente entre un experto aislado en relación a un grupo de expertos de referencia, mientras que P_n representa el acuerdo existente dentro de dicho grupo de referencia. P_0 se define en la Figura 4.7 y P_n en la Figura 4.8 en las que n representa el número de expertos de referencia y $P(a, b)$ una medida de pares que relaciona las interpretaciones de los expertos a y b .

$$I_0 = \frac{P_0}{P_n}$$

Figura 4.6: Ecuación del índice de Williams

$$P_0 = \frac{\sum_{a=1}^n P_{(0,a)}}{n}$$

Figura 4.7: Ecuación para calcular el acuerdo existente entre un experto aislado en relación a un grupo de expertos de referencia.

$$P_n = 2 \frac{\sum_{a=1}^{n-1} \sum_{b=a+1}^n P_{(a,b)}}{n(n-1)}$$

Figura 4.8: Ecuación para calcular el acuerdo existente dentro un grupo de referencia.

La interpretación del índice de Williams (I_n) es la siguiente:

- Si I_n es menor que uno indica que el acuerdo entre el experto aislado y el grupo de expertos es menor que el acuerdo entre los propios miembros del grupo.
- Si I_n es igual a uno indica que el experto aislado coincide con el grupo al mismo nivel que los miembros del grupo coinciden entre sí.

- Si *In* es mayor que uno podríamos decir que el experto aislado coincide con el consenso del grupo de expertos.

4.2.2 Validación del Sistema Experto implementado

De acuerdo a lo explicado en el sub-epígrafe anterior, se procede a validar el Sistema Experto implementado. Primero se crea la base de datos de validación, para esto se determinan los desafíos que va a contener la misma y los niveles de dificultad que se van a utilizar.

Los desafíos presentados a los expertos humanos y al Sistema Experto consisten en situaciones que se puedan presentar los usuarios en su trabajo con el sistema operativo Nova y cómo las clasificarían de acuerdo a su dificultad para identificarlas y para solucionarlas. Se utilizarán tres niveles de dificultad: fácil, asequible y difícil. Los desafíos se encuentran contenidos en la Tabla 4.2:

Número	Causas	Problema a identificar
1	No se puede acceder a particiones del disco o no se puede montar una partición del disco.	Montaje de particiones NTFS y FAT 32
2	No se ve la impresora o no funciona.	Instalar impresora
3	Se desea instalar Windows sin perder Nova.	Restaurar el grub
4	No se ve la red inalámbrica y falla al tratar de conectarse.	Conectarse por medio de una red inalámbrica
5	No se ve la red cableada y falla al tratar de conectarse.	Conectarse por medio de una red cableada
6	No se pueden montar los dispositivos externos o no reconoce las memorias.	Montaje de dispositivos
7	No se encuentran las letras en el teclado o no se corresponden las letras en el teclado.	Cambiar configuración del teclado
8	No se puede acceder a los archivos.	Comprimir y extraer archivos
9	No se puede entrar por el dominio.	Unir PC al dominio
10	Se desea comprobar si la imagen de Nova se descargó bien.	Verificación de la integridad de las imágenes ISO

Tabla 4.2 Desafíos para la validación del Sistema Experto

Utilizando la información de los desafíos y las clasificaciones realizadas por los desarrolladores del Sistema Experto y el grupo de expertos de acuerdo a los niveles propuestos se elabora la base de datos

CAPÍTULO 4: ANÁLISIS DEL COMPORTAMIENTO DEL SISTEMA EXPERTO.

de validación, quedando como se muestra en la Tabla 4.3:

Identificador del desafío	Sistema Experto	Experto 1	Experto 2	Experto 3
1	Asequible	Asequible	Asequible	Fácil
2	Fácil	Asequible	Fácil	Fácil
3	Difícil	Difícil	Difícil	Difícil
4	Asequible	Asequible	Fácil	Asequible
5	Asequible	Fácil	Fácil	Fácil
6	Asequible	Difícil	Difícil	Asequible
7	Fácil	Fácil	Fácil	Fácil
8	Fácil	Fácil	Fácil	Fácil
9	Asequible	Asequible	Fácil	Asequible
10	Fácil	Fácil	Fácil	Fácil

Tabla 4.3 Base de datos de validación

A partir de la base de datos de validación se crean las tablas de contingencia que representan los acuerdos y desacuerdos entre los expertos.

E2\E1	F	A	D	Total
F	4	3		7
A		1		1
D			2	2
Total	4	4	2	10

E3\E1	F	A	D	Total
F	4	2		6
A		2	1	3
D			1	1
Total	4	4	2	10

E3\E2	F	A	D	Total
F	6			6
A	1	1	1	3
D			1	1
Total	7	1	2	10

SE\E1	F	A	D	Total
F	4			4
A		4	1	5
D			1	1
Total	4	4	2	10

CAPÍTULO 4: ANÁLISIS DEL COMPORTAMIENTO DEL SISTEMA EXPERTO.

SE\E2	F	A	D	Total
F	4			4
A	3	1	1	5
D			1	1
Total	7	1	2	10

SE\E3	F	A	D	Total
F	4			4
A	2	3		5
D			1	1
Total	6	3	1	10

Tabla 4.4 Tablas de contingencia de los pares de expertos.

A continuación se calculan las medidas de pares para cada par de expertos, utilizando el índice de acuerdo. Los resultados se muestran en la Tabla 4.5.

	Experto 1	Experto 2	Experto 3	Sistema Experto
Experto 1	-	0,7	0,7	0,9
Experto 2	0,7	-	0,8	0,6
Experto 3	0,7	0,8	-	0,8
Sistema Experto	0,9	0,6	0,8	-

Tabla 4.5 Tablas de pares con los valores del índice de acuerdo.

En principio, estos parecen buenos resultados pero se va a comprobar si al introducir al Sistema Experto dentro del grupo de expertos disminuye el nivel de consenso en el grupo. Para ello se calcula por un lado la medida de grupo de los tres expertos, obteniéndose un valor P_n igual a 0,733, y por otro lado la medida del acuerdo del Sistema Experto con los expertos humanos, P_0 , que toma un valor igual a 0,767. Luego se calcula el índice de Williams (I_n) obteniéndose 1,046 como resultado, lo que significa que introducir al Sistema Experto en el grupo de expertos no supondría una disminución del consenso, es decir, el comportamiento del Sistema Experto se asimila al comportamiento de un experto humano. Se confirma de esta manera que el comportamiento del Sistema Experto es el adecuado, y puede realizar el diagnóstico de los problemas que presenten los usuarios en su trabajo con el Sistema Operativo Nova del mismo modo que lo realizaría un experto humano.

4.3 Evaluación del Sistema Experto

La evaluación de un Sistema Experto incluye diferentes aspectos que van más allá de la corrección de las soluciones finales, debido a que en esta etapa se analiza la utilidad, robustez, posibilidades de ampliación, facilidad de manejo y beneficio del Sistema Experto.

El Sistema Experto implementado es de gran utilidad, pues es de gran importancia en el proceso de migración al software libre que se lleva a cabo en Cuba, su objetivo es el diagnóstico de los problemas que presenten los usuarios en su trabajo con el Sistema Operativo Nova y esto contribuye de forma

positiva con el proceso de migración.

La robustez de un Sistema Experto consiste en la capacidad de establecer la relevancia de determinada información en orden a obtener sus objetivos, el Sistema Experto basado en reglas tiene que ocuparse de situaciones no ideales. Estas situaciones no ideales podían ser errores en la entrada, falta de conocimiento o incertidumbres en la interpretación, en esos casos el sistema tiene que corregir el problema o, si no, proveer la mejor respuesta posible con el conocimiento disponible. El Sistema Experto implementado cumple con esta característica pues si la información proporcionada por el usuario no es suficiente para identificar un problema, él es capaz de brindar otra vía de solución.

El Sistema Experto implementado posee posibilidades de ampliación pues permite que las reglas, las preguntas para identificar problemas y las preguntas asociadas a las categorías de los problemas sean gestionadas, es decir, que sean insertadas, modificadas, mostradas y eliminadas. De esta manera cuando surjan nuevos problemas en el uso del Sistema Operativo Nova se puede ampliar la base de conocimientos de NovaDesk, así como adicionar las preguntas y las reglas necesarias que permitan identificarlos.

Cuenta además con una amplia facilidad de manejo pues está implementado de manera que un usuario inexperto pueda interactuar con él, mediante descripciones de cómo usar el sistema, fácil acceso a las ventanas de consulta del sistema y sencilla selección de las respuestas. A su vez le permite al usuario identificar y solucionar el problema que presenta. De forma general facilita la relación e interacción del usuario con el Sistema Operativo Nova.

El Sistema Experto desarrollado es de gran beneficio para la aplicación Web NovaDesk pues brinda una primera línea de soporte en esta aplicación permitiendo una mejora en la atención a los usuarios y otra vía para que estos identifiquen y solucionen los problemas que presenten en su trabajo con Nova.

Conclusiones generales

- Se realizó un estudio de los Sistemas Expertos, sus características, componentes, clasificaciones, así como lenguajes y herramientas para su implementación.
- Se desarrolló un Sistema Experto que posibilita la gestión de la base de conocimientos de la aplicación web Novadesk permitiendo lograr una mejor atención a los usuarios al brindar una respuesta inmediata al problema que presenten.
- Los problemas que no puedan ser resueltos serán reportados como incidencias, posteriormente resueltos e incorporados a la base de conocimientos de NovaDesk y al Sistema Experto con sus posibles soluciones.
- Se analizó el comportamiento del Sistema Experto implementado mediante la verificación, validación y evaluación del mismo.
- Se determinó que el comportamiento del Sistema Experto es similar al comportamiento de un experto humano y debido a esto, puede realizar el diagnóstico de los problemas que presenten los usuarios en su trabajo con el Sistema Operativo Nova del mismo modo que lo realizaría un experto humano.
- El Sistema Experto de NovaDesk aporta grandes beneficios a los usuarios del sistema operativo Nova pues permite identificar y diagnosticar los problemas más comunes que presentan los mismos en su trabajo con este Sistema Operativo.

Recomendaciones

Los autores de este trabajo recomiendan:

- Propiciar la utilización del Sistema Experto de la aplicación web NovaDesk para brindar soporte a Nova.
- Mantener el seguimiento del Sistema Experto por parte de los expertos humanos.
- Ampliar la Base de Conocimientos de NovaDesk con nuevos problemas y soluciones.
- Incrementar las reglas del Sistema Experto.
- Incorporarle autoaprendizaje al Sistema Experto.
- Realizar una validación orientada al uso.

Referencias bibliográficas

- [1] Lic. María Vidal Ledo. Revista Cubana de Informática Médica (RCIM). ALFABETIZACIÓN DIGITAL E INFORMACIÓN DE LA SOCIEDAD. UN RETO PARA EL PRESENTE Diciembre del 2008. [Cited 1 April 2010]. Available from World Wide Web:
<http://www.cecam.sld.cu/pages/rcim/revista_9/articulos_htm/alfabetizdigital.htm>.
- [2] Emilio Lago Henríquez, y Yusdel Juan Concepción Peña. Aplicación Web para brindar soporte a Nova: NovaDesk. June 2009, 110.
- [3] Yaima Oval Riverón. Servicio de Soporte Técnico utilizando la tecnología Service-Desk. 2007, 111.
- [4] Leanet Tamayo Oro, y Yanet Pavón Bernal. Herramienta para el Soporte de los Servicios NovaDesk 2.0. 2010, 103.
- [5] Curso_ITIL. Extraído el 21 de marzo del 2007 de:
http://itil.osiatis.es/Curso_ITIL/Gestion_Servicios_TI/fundamentos_de_la_gestion_TI/que_es_ITIL/soporte_al_servicio.php
- [6] LEIBMANN M. Un camino hacia las soluciones de gestión del conocimiento. Washington: Microsoft. 1999.
- [7] M. Nüssbaum, "Gestión del Conocimiento", Clase 1
- [8] Serradell López, Enric y Juan Pérez, Ángel A. "La gestión del conocimiento en la nueva economía"
- [9] Criado Briz, José Mario. Introducción a los Sistemas Expertos. [En línea] 2010. [Citado el: 13 de Noviembre de 2010.] http://www.ingenieroseninformatica.org/recursos/tutoriales/sist_exp/index.php.
- [10] Hurtado Vega, José de Jesús. Inteligencia Artificial. [En línea] 2010. [Citado el: 16 de Noviembre de 2010.] <http://www.itlp.edu.mx/publica/boletines/actual/inteligencia.html>.
- [11] Pajares Santos, M. Inteligencia Artificial e Ingeniería del Conocimiento. Ra-Ma, 2005. [En línea] 2008.
- [12] Giarratano, Riley. Sistema Experto, principios y programación. ISBN 7-111-10844-2.
- [13] Rodríguez, Yanet. Evaluación de un Sistema Basado en Casos Interpretativo. Tesis de Maestría. 1998.
- [14] Gutiérrez, José Manuel y S. Hadi, Ali. Libro Sistema Experto y Redes Probabilísticas.
- [15] Arrúa, Luciana y Meza Fernández, Eduardo. Inteligencia Artificial: Sistema Expertos y Redes Neuronales, 2003
- [16] Riesbeck, Christopher K, y otros. AICOM. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. s.l.: Inside Case-Based Reasoning Lawrence Erlbaum Associates, Publishers, 1989. Vol. 7, 1, March 1994.
- [17] Russell, S. J. y Norvig, P. Artificial Intelligence: A Modern Approach. s.l.: 2nd Ed, Prentice Hall,
- [18] García García, Maribel y otros. DITRITS, SISTEMA EXPERTO PARA EL DIAGNÓSTICO Y TRATAMIENTO DE INFECCIONES DE TRANSMISIÓN SEXUAL.

- [19] Vargas Rodríguez, Heriberto y Ponce de León Lima, Daniel. Evaluación de la aptitud de las tierras del municipio San José de las Lajas para las clases generales de uso agrícola ganadero. I. Aptitud física, Revista Ciencias Técnicas Agropecuarias, Vol. 17, No. 4, 2008
- [20] Oficina Nacional de Estadísticas de Cuba. Panorama económico y social. Cuba: s.n., 2008.
- [21] Gutiérrez Rodríguez, Marianela y Bedoya Rusenko, Jorge. Sistema Experto para el diagnóstico médico de las enfermedades genéticas con dismorfias (SEGEDIS). 2010, 78
- [22] Revilla Mugarra, Daniel Ernesto y Sánchez Fernández, Enrique. IMPLEMENTACIÓN DE UN SISTEMA PARA EL CÁLCULO DE RIESGO DE ENFERMEDADES GENÉTICAS. 2010, 66.
- [23] Izquierdo, Raykenler y Lazo, René. "Diagnóstico y Tratamiento de la hipertensión arterial", 2006-2007
- [24] Prado Lemus, Edison y González Yero, Leandro José. Herramienta inteligente para la captura de Requisitos No Funcionales. Ciudad de la Habana, 2009, 87
- [25] González Rodríguez, Yoney y Rodríguez Bombalé, Roberto. Modelo Basado en Casos para la planificación de proyectos de software. Junio 2009, 105
- [26] Mena Jorge, Ernesto Ferrer y García Suárez-Arango, Adrián. Desarrollo del sistema SEDIM-SV utilizando la metodología Weiss-Kulikowski. Junio 2008, 94
- [27] Campoalegre Vera, Lázaro. Herramienta de Generación de Código Mediante Sistema Experto. Junio 2008, 91
- [28] Alonso Jiménez, José Antonio, Martín Mateos, Francisco Jesús, y Ruiz Reyna, José. Tema 1: Introducción a Lisp. Curso -2011 2010.
- [29] Luis Daniel Hernández Molinero. TUTORIAL DE CLIPS VERSION 1.01, December 22, 2006
- [30] Tutorial de Prolog. [cited 7 Diciembre 2010]. Available from World Wide Web: <<http://proton.ucting.udg.mx/tutorial/prolog/index.htm>>.
- [31] CLIPS - PHP Extension. [Cited 7 Diciembre 2010]. Available from World Wide Web: <<http://phlips.sourceforge.net/>>.
- [32] Universidad de Carlos III de Madrid. Desarrollo de aplicaciones interactivas en java. July 2002. [Cited 22 April 2010]. Available from World Wide Web: <<http://www.scribd.com/doc/967380/Tutorial-Netbeans>>.
- [33] Rafael Andrés Leño, y Luis Carlos Beltrán. La Arquitectura De Netbeans. 2008. [Cited 22 April 2010]. Available from World Wide Web: <<http://www.slideshare.net/ralphkui/la-arquitectura-de-netbeans-v2>>.
- [34] Castro Souto, Laura M. y otros. Inteligencia Artificial, curso 2001-2002.
- [35] Moret Bonillo, Vicente. Validación y Usabilidad de Sistemas Informáticos, mayo de 2005, página 39.
- [36] Moret Bonillo, Vicente. Validación y Usabilidad de Sistemas Informáticos, mayo de 2005, página 40.
- [37] Moret Bonillo, Vicente. Validación y Usabilidad de Sistemas Informáticos, mayo de 2005, página 56.

Bibliografía consultada

- Abel Alfonso Fírvida Donéstevez. Concepción del sistema Nova. Nova para soluciones de escritorio. 2009.
- Abel Meneses Abad, Viviana Ávarez Ferriol, Yenisleidys Cariaga Cristo, and Yoandy Pérez Villazón. Primer Concurso Universitario Software Libre. 2006. [Cited 2 November 2009].
- Castro Souto, Laura M. y otros. Inteligencia Artificial, curso 2001-2002.
- Claudio Hormazábal. Gestión Libre del Parque Informático, GLPI. March 2009. [Cited 14 May 2010]. Available from World Wide Web: <<http://softwarelibrevenezuela.blogspot.es/i9-03/>>.
- cubasi.cu. Cuba. Noticias de Cuba. Informatización vs. bloqueo. October 2006. [Cited 27 January 2010]. Available from World Wide Web: <<http://www.cubasi.cu/desktopdefault.aspx?spk=160&clk=134453&lk=1&ck=69740&spka=35>>.
- Daniel Carrión Reinoso et al. BREVE HISTORIA DEL SWL. 2006. [Cited 22 January 2010]. Available from World Wide Web: <<http://www.infor.uva.es/~bperezle/disenio/TeoriaUNIX0607>>.
- Elena Verdú Pérez. Contribución a la aplicación de técnicas de Inteligencia Artificial para el diseño efectivo de sistemas adaptativos de aprendizaje competitivo. 2010, 140.
- Emilio Lago Henríquez, and Yusdel Juan Concepción Peña. Aplicación Web para brindar soporte a Nova: NovaDesk. June 2009, 110.
- infoALEPH Team. Gestión libre del parque informático (glpi) « infoALEPH. June 2008. [Cited 29 March 2010]. Available from World Wide Web: <<http://infoaleph.wordpress.com/2008/07/24/gestion-libre-del-parque-informatico-glpi/>>.
- Juventud Rebelde. Cuba celebra el software libre - Cuba - Juventud Rebelde - Diario de la juventud cubana. de Abril del 2009. [Cited 27 January 2010]. Available from World Wide Web: <<http://www.juventudrebelde.cu/cuba/2009-04-26/cuba-celebra-el-software-libre/>>.
- Leanet Tamayo Oro, y Yanet Pavón Bernal. Herramienta para el Soporte de los Servicios NovaDesk 2.0. 2010, 103.
- Lic. María Vidal Ledo. Revista Cubana de Informática Médica (RCIM). *ALFABETIZACIÓN DIGITAL E INFORMACIÓN DE LA SOCIEDAD. UN RESTO PARA EL PRESENTE* Diciembre del 2008. [Cited 1 April 2010]. Available from World Wide Web: <http://www.cecam.sld.cu/pages/rcim/revista_9/articulos_htm/alfabetizdigital.htm>.
- MINISTERIO DE CIENCIA Y TECNOLOGÍA. PLAN NACIONAL DE MIGRACIÓN A SOFTWARE LIBRE DE LA ADMINISTRACIÓN PÚBLICA NACIONAL. March 2005. Available from World Wide Web: < www.mct.gov.ve>.
- Moret Bonillo, Vicente. Validación y Usabilidad de Sistemas Informáticos, mayo de 2005.
- Nova y la apuesta cubana por el software libre. 2009. [Cited 27 January 2010]. Available from world wide web:

<http://www.cadenagramonte.cu/index.php?option=com_content&view=article&id=1392:nova-y-la-apuesta-cubana-por-el-software-libre&catid=41:tecnologia&Itemid=8>.

- Rafael Andrés Leaña, and Luis Carlos Beltrán. La Arquitectura De Netbeans. 2008. [Cited 22 April 2010]. Available from World Wide Web: <<http://www.slideshare.net/ralphkui/la-arquitectura-de-netbeans-v2>>.
- Ramón Paumier Samón, Yoandy Pérez Villazón, and Abel Meneses Abad. GUÍA CUBANA DE MIGRACIÓN A SOFTWARE LIBRE. 2008.
- Resulta Servicios de Marketing S.A.U. Resulta - Glosario. 2004. [Cited 25 March 2010]. Available from World Wide Web: <<http://www.resulta.es/index.php?page=glosario>>.
- Sayda Coello González, Rolando Alfredo Hernández León. *EL PARADIGMA CUANTITATIVO DE LA INVESTIGACIÓN CIENTÍFICA*. [Ciudad de la Habana]: EDUNIV, 2002.
- Sergio Talens-Oliag. Breve historia de Linux y el movimiento del Software Libre. 2004. [cited 22 January 2010].
- SofSO Team. GLOSARIO en línea | sofocorp.com portal. 2009. [Cited 25 March 2010]. Available from World Wide Web: <<http://www.sofocorp.com/glosario>>.
- Steven Pemberton. Preguntas frecuentes sobre XHTML. July 2004. [Cited 21 April 2010]. Available from World Wide Web: <<http://www.w3c.es/Traducciones/es/Markup/2004/xhtml-faq.htm#need>>.
- Tutorial de Prolog. [cited 7 Diciembre 2010]. Available from World Wide Web: <<http://proton.ucting.udg.mx/tutorial/prolog/index.htm>>.
- Yaima Oval Riverón. Servicio de Soporte Técnico utilizando la tecnología Service-Desk. 2007, 111.

Anexos

Anexo 1 Encuesta a los estudiantes

Encuesta para los estudiantes acerca del sistema Operativo Nova:

Estimado estudiante: Con la presente encuesta usted está colaborando a una investigación llevada a cabo para conocer el nivel de conocimiento, preparación y aceptación que posee del Sistema Operativo Nova, así como los problemas que usted ha encontrado en su trabajo con este sistema, por lo que se requiere de su máxima sinceridad.

Gracias por su colaboración.

1. ¿Te gusta estudiar usando Nova?

Sí no

¿Por qué? _____

2. ¿Te has encontrado con problemas durante tu trabajo con Nova?

Sí no

¿Cuáles? _____

3. ¿Puedes solucionar los problemas con los que te has encontrado?

Sí no

¿Por qué? _____

Anexo 2 Preguntas de la entrevista al experto

- ¿Cuáles son las categorías de clasificación de los problemas que se pueden presentar durante el trabajo con Nova?
- ¿Cuáles son los problemas más comunes a los que se enfrenta un usuario cuando utiliza el Sistema Operativo Nova?
- ¿En qué categorías clasificaría cada uno de estos problemas?
- ¿Cómo se identifican cada uno de estos problemas?
- ¿Qué solución se les daría a cada uno de estos problemas?

Glosario de términos

Activos: Es el conjunto de bienes tangibles o intangibles que posee una empresa.

Costes o **costo:** Montante económico que representa la fabricación de cualquier componente o producto, o la prestación de cualquier servicio.

Determinista: Perteneciente o relativo al determinismo.

Determinismo: Doctrina filosófica que preconiza la tesis que entre todos los acontecimientos hay una relación ineludible de causa a efecto, especialmente en cuanto a herencia y ambiente como condicionantes de las posibilidades del ser humano, el que de ninguna manera puede sustraerse a esta relación preestablecida. Se le opone la doctrina del libre albedrío.

HTTP: (Hypertext Transfer Protocol, en español protocolo de transferencia de hipertexto).

Ítem: Palabra latina adoptada y difundida por el inglés. Significa cada uno de los artículos que constituyen un escrito. Como sustantivo se utiliza para designar los mismos artículos. Los angloparlantes usan intensamente la palabra para designar también a un artículo en el sentido de objeto unitario.

Prototipo: Ejemplar original o primer molde en que se fabrica una figura u otra cosa. Producto terminado al que se llega en la fase de investigación y desarrollo, pero que no es, todavía, el definitivo que se comercializa.

Solución: Es la propuesta de la solución definitiva al problema, es analizada con respecto a su: posible impacto, viabilidad y conveniencia.

Torres de Hanoi: El problema de las torres de Hanoi consiste en mover un conjunto de discos de un palo a otro palo utilizando un palo auxiliar situado en medio. Las reglas para mover los discos son las siguientes: no se puede mover más de un disco en cada ocasión; no puede quedar un disco de menor radio debajo de un disco de mayor radio; mover el conjunto de discos se puede ver como: mover $N-1$ discos del palo izquierdo al palo del medio, mover el disco sobrante del palo izquierdo al palo derecho y mover $N-1$ discos del palo del medio al palo derecho.

Releases: Nueva versión de una aplicación informática.

Usuarios: las personas que utilizan el servicio.