

Sistema experto como propuesta de mejora orientado al diagnóstico y reducción de problemas de software y hardware que se presenten en la Cooperativa de Loteros

Expert system as a proposal for improvement oriented to the diagnosis and reduction of software and hardware problems that are presented in the Cooperativa de Loteros

José David Gil Gutiérrez¹
jose.gil03@usc.edu.co

Yana Saint-Priest, M.Sc²
yana.saint-priest00@usc.edu.co

Universidad Santiago de Cali, Facultad de Ingeniería, Programa de Ingeniería en Sistemas (1)
Universidad Santiago de Cali, Facultad de Ingeniería, Programa de Ingeniería en Sistemas (2)

Resumen

Generalmente las pequeñas empresas no cuentan con un departamento de tecnología que atienda los problemas de software o hardware, ocasionando pérdida en la continuidad del negocio; ya que, al no tener soporte técnico interno tardan demasiado en solucionar o mitigar dichas fallas. Este artículo presenta la implementación de un sistema experto para el diagnóstico y reducción de problemas de software y hardware en la empresa Cooperativa de Loteros. El uso de Systemap es intuitivo y las soluciones propuestas se conciben desde soporte técnico en casa hasta un nivel técnico empresarial. Se empleó la metodología Buchanan para construir el sistema, sus fases son: selección del problema, modelo de construcción, formalización e implementación. El sistema propuesto contiene conocimientos, información y experiencias del proveedor que proporciona los dispositivos tecnológicos a la empresa, para hacer eficiente el proceso de diagnóstico y brindar óptimas soluciones. El sistema experto se integra a la manera de operar de pequeñas empresas, facilitando la identificación de problemas y la formulación de estrategias para reducir los problemas que se presentan, además de mitigar los costos asociados con consultorías externas especializadas.

Palabras Clave: sistemas expertos, soporte técnico, Prolog.

Abstract

Generally, small businesses do not have a technology department that addresses software or hardware problems, causing loss of business continuity; since, having no internal technical support, it takes too long to solve or mitigate these failures. This article presents the implementation of an expert system for the diagnosis and reduction of software and hardware problems in the Cooperativa de Loteros. The use of the Systemap is intuitive and the proposed solutions are conceived from technical support at home to a business technical level. The Buchanan methodology was used to build the system, its phases are: problem selection, construction model, formalization and implementation. The proposed system contains knowledge, information and experiences of the provider that provides the technological devices to the company, to make the diagnostic process efficient and provide solutions. The expert system is integrated into the way small businesses operate, facilitating the identification of problems and the formulation of strategies to reduce the problems that arise, in addition to mitigating the costs associated with specialized external consultancies.

Keywords: expert systems, technical support, Prolog.

1. INTRODUCCIÓN

La innovación de las tecnologías de la información y las comunicaciones (TIC), se ha convertido en el motor impulsor de todas las posibilidades que en la actualidad ofrece la informática. La oportunidad de intercambiar información contenida entre miles de bases de datos, la interconexión entre computadores en tiempo real y su interacción en función de un conjunto de órdenes previamente programadas, ha ampliado los horizontes de lo que se conocía desde el punto de vista comercial, como negocio. En este artículo se describe un producto informático, el cual se apoya en técnicas útiles para tratar la información, disminuir la incertidumbre y apoyar la toma de decisiones (Elena & Luque, 2019).

La Cooperativa de loteros es una pequeña empresa comercial dedicada a actividades de juego de azar exclusivamente a la venta de billetes de lotería. Esta cuenta con una base de recursos tecnológicos conformados por 4 equipos de cómputo, 3 impresoras (2 Multifuncional y POS), sistema de cámaras de seguridad (4 cámaras, DVR), 4 lectores laser de diferentes marcas y proveedores además de un software especializado para el manejo de loterías llamado LOTTERY.

Existen problemas relacionados con estos dispositivos que afectan la continuidad del negocio: atasco de papel, desalineación de texto, manchones de tinta, desajuste de rodillo para las impresoras (Pos, multifuncionales), etc. Por estos medios imprimen recibos de caja, vales y reportes, documentos críticos, que son necesarios para llevar control y dar respaldo de las operaciones al cliente.

Para el sistema de seguridad: problemas de grabación, liberar almacenamiento en el DVR, fallas en las cámaras, entre otros, que afectan el monitoreo. En el caso de los lectores, la dificultad al registrar nuevos productos o códigos de barras por cambio en el material de los billetes de lotería o simplemente por mala impresión del proveedor, complica el ingreso de los productos al sistema y retrasa el despacho de estos.

Según Montiel & Riveros (2014), un sistema experto puede ser descrito como un sistema informático que posee un conocimiento específico (conocimiento experto) representado, y funciona mediante la aplicación de un mecanismo de inferencia para obtener sus resultados. Se utilizan para resolver problemas mostrando similitudes a respuestas obtenidos por un experto humano en la solución de ciertas tareas de un área de conocimiento, tales como: medicina, ingeniería, ciencias exactas, naturales, humanas, entre otras. Los sistemas expertos, basados en reglas, son una forma válida de usar el conocimiento e inferir estados. Esto es posible ya que, los hechos y reglas, están directamente relacionadas al manejo y comportamiento del sistema experto.

Estos sistemas expertos se están usando para tareas como la toma de decisiones, por ejemplo Flores & Hadfeg(2017) describen un trabajo donde usa un sistema de expertos basado en reglas para analizar y predecir los gustos de algunos consumidores de un nuevo producto Chilcanan, Navas, & Escobar (2017). También describen un trabajo reciente en la industria de la minería de cobre, el trabajo consiste en crear criterios limpios en la futura explotación de este mineral en zonas no explotadas hasta ahora, usando conocimiento experto y la técnica de neuro-fuzzy para crear modelos y predicción de la explotación.

En una empresa del sector financiero existían diferentes problemas generados por ataques de virus informático que causaban pérdida y modificación de datos en las organizaciones, esto conllevó a plantear un proyecto en el que se desarrolló un sistema experto titulado VIRUS_ITM que permite detección y eliminación de este tipo de programas maliciosos. Adopta conocimiento específico de expertos en virus informático mediante relaciones entre las variables u objetos de entrada y perjuicios ocasionados (Gutierrez.L, 2013).

Estudiantes de la Universidad de Medicina de Teherán implementaron un sistema experto para ayudar a los profesionales a tomar decisiones médicas difíciles. En este caso, se diseñó un software de detección asistida por computadora, que clasifica deformación de próstata en dos clases: cáncer e hiperplasia, para apoyar al urólogo en la toma de decisiones respecto a un procedimiento crucial para la supervivencia del paciente (Ghaderzadeh, Sadoughi, & Ketabat, 2013).

El presente artículo describe el desarrollo tecnológico de un sistema experto que permite ofrecer diagnósticos y soluciones a problemas de software y hardware en la Cooperativa de Loteros. Contiene conocimiento detallado de **TL Usados Store** proveedor de dispositivos, herramientas y tecnologías utilizadas en la empresa.

Para el desarrollo del proyecto se propuso la metodología Buchanan, La primera fase inicia por la obtención de información y conocimiento brindada por el proveedor experto, el cual describe los dispositivos relacionados, su tecnología, funcionamiento y usabilidad con el fin de enfocar las necesidades e inconformidades respecto al uso de estos (Norvig, P., & Russell, 2014).

La segunda fase consiste en conformar la base de conocimiento y presentar el modelo de construcción que contiene información de los posibles problemas de cada dispositivo involucrado, incluyendo su identificador. Se enmarca únicamente en los dispositivos presentes en la Cooperativa de Loteros, sus atributos y características tecnológicas.

En la fase tres se organizaron los hechos y reglas de lenguaje natural a lenguaje formal, sentencias que relacionan los recursos tecnológicos con el funcionamiento de la empresa. Se comprendió la operatividad de la empresa para formar un vocabulario de los términos y palabras esenciales, y luego desarrollar una interpretación de los conceptos a partir de la información brindada por los proveedores.

Por último, en la fase cuatro de implementación se construyó un prototipo inicial que representa una parte del sistema experto, el cual sirve de guía para aplicar los siguientes incrementos. Se usa la base de conocimiento con los recursos tecnológicos organizados previamente, para relacionar las reglas y los hechos. El prototipo interactuó con diferentes tipos de usuarios y un lenguaje común que favoreció el entendimiento del software.

Este prototipo funcional se desarrolló en Prolog, un lenguaje de programación lógica de tipo declarativo común para el desarrollo de sistemas expertos. estricto en sintaxis y estructura; utilizado en el desarrollo de sistemas expertos que especifican un conjunto de hechos, reglas, proposiciones, afirmaciones, observaciones o situaciones que describen el problema y aportan una solución.

Resaltando estas características para el desarrollo de sistemas expertos, prolog proporciona mecanismos matemáticos para mejorar el rendimiento de los programas además de relacionar elementos u objetos para su programación. Las sentencias que utiliza están basadas en la lógica de primer orden que interpreta problemas a partir de inferencias negativas o positivas según sea el caso. prolog es ideal para el desarrollo de sistemas expertos ya que tiene una base de conocimiento independiente modificable para su continua actualización.

El sistema cuenta con interfaces estáticas que permiten mejor usabilidad a los usuarios, ya que siempre ofrece 4 botones principales para hacer uso de su funcionalidad. Para la utilización del software es necesario instalar Prolog en los equipos donde se desee usar el sistema (Roussel, 1995).

El desarrollo de un sistema experto trae beneficios directos a la organización, los tiempos de adaptación, detección y reparación de fallas se reducen, se podría aplicar soporte técnico en casa, siempre y cuando no afecte la garantía o requiera reemplazo de piezas (Barceló, Alonso, de la Cruz, & Cendejas, 2009).

El artículo se estructura de la siguiente forma: la sección 2 de metodología es donde se detalla el desarrollo del sistema; la sección 3 de resultados, diseña y describe las pruebas y retroalimentaciones y; en la sección 4 se listan las conclusiones de esta investigación.

2. METODOLOGÍA

La metodología buchanan se apoya en el ciclo de vida o modelo en cascada utilizado en la ingeniería del software, la cual se caracteriza por ordenar 7 etapas de desarrollo las cuales son: **el análisis de información, modelo de construcción**, diseño del programa, **formalización**, pruebas, **implementación** y mantenimiento. Se puede interpretar que el proceso de desarrollo de un sistema experto es un proceso de revisión constante, que puede implicar la redefinición de conceptos, las representaciones o la actualización del sistema implementado.

Para el desarrollo de este sistema experto se utilizaron 4 etapas principales de la metodología adaptadas al caso estudio de la cooperativa: análisis y adquisición de información, modelo de construcción, formalización, implementación.

Adquisición de información: El proveedor detalló la funcionalidad, usabilidad, fallas y posibles soluciones de los problemas según la lista de dispositivos con los que cuenta la Cooperativa. Apoyado en las experiencias y los manuales de uso recompilado, se procede a conformar la base del conocimiento, la cual contiene la lista de los requerimientos (Mayor & Cauca, 2016).

La documentación o información obtenida debe permitir la construcción del diseño del sistema y servir de referencia para solucionar los problemas que se puedan presentar (Carballo, 2015). A continuación, se muestra la tabla de dispositivos hardware, características y marcas, con los que cuenta la Cooperativa de loteros:

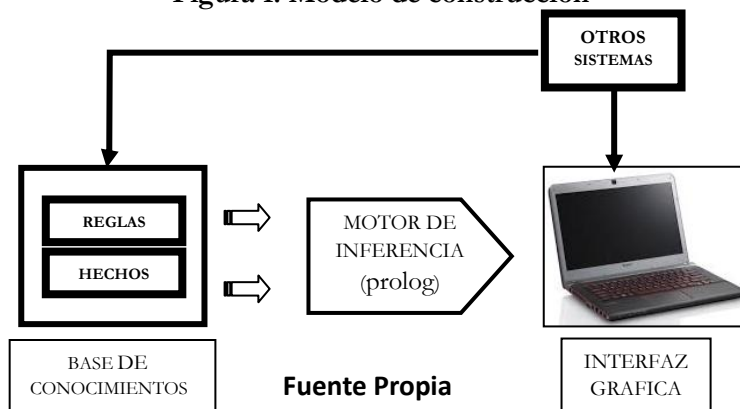
Tabla 1: Recursos de Hardware

COMPUTADORES		SISTEMA DE SEGURIDAD	
HARDWARE	MARCA	HARDWARE	MARCA
-Procesador Intel Core I3 2.94 GHz -S.O Windows 10 Pro -4 GB de RAM -500 GB de disco duro	DELL Optilex 780 x 2	DVR Capacidad de almacenamiento 1 Tb	DS7204 HIKVISION
-Procesador Intel Core 2Dúo2.6 GHz -S.O Windows10 Pro -4 GB de RAM -400 GB de disco duro	ASX (clon)	-3 Cámaras indoor/outdoor -1 Cámara Ni MI	HIKVISION DS-2CD2
-Procesador E1 AMD 1.4 GHz -S.O Windows 10 Pro -4 GB de RAM -500 GB de disco duro	Computadora Portátil TOSHIBA Satélite C-55 ^a	RED	
IMPRESORAS		HARDWARE	MARCA
HARDWARE	MARCA	1 Switch de 8 puertos	TP-LINK
-MAXIFY 2110 -TX525FW -TM-U220D (TIRILLAS, FACTURA DE CAJA, VALES)	CANON EPSON POS EPSON	1 MODEM DE INTERNET MULTIPUERTO (4 PUERTOS)	D-LINK

Fuente Propia

Modelo de construcción: el sistema se compone por un motor de inferencia, el cual tiene capacidad de ingresar nueva información a la base de conocimiento utilizando lógica de primer orden. Para este caso, el motor de inferencia se encuentra inmerso en Prolog, lenguaje basado en programación lógica. Este contiene la base de conocimiento compuesta por un conjunto de hechos y reglas de inferencia que describen detalles y características de los dispositivos tecnológicos de la organización (Fig. 1).

Figura 1: Modelo de construcción



Fuente Propia

La base de conocimiento se constituyó por hechos y reglas basadas en opiniones, información y conocimiento recolectado por medio del proveedor, además de fuentes confiables como artículos universitarios, bibliotecas virtuales universitarias y proyectos académicos. Los hechos son enunciados que describen detalles lógicos sobre el funcionamiento de los diferentes dispositivos, estos determinan la relación de los elementos y así generan las sentencias. Apoyado en la lógica de primer orden, un hecho sería “todos los computadores tienen cables” o “El computador funciona sin tarjeta de sonido”, se involucraron tres objetos: computador, tarjeta de sonido y cables. Para estas sentencias, se asume por lógica que el computador necesita cables internos o externos que transportan energía e información para su funcionamiento, más la tarjeta de sonido no es un dispositivo que impida el funcionamiento integral del computador.

El desarrollador del sistema interpretó la información y experiencias del proveedor, sobre el manejo y seguimiento de los dispositivos tecnológicos entregados a la cooperativa. La lógica con la que se diseñó la base de conocimientos tuvo como principio redactar las reglas sobre los hechos, ya que estas deciden las operaciones que se deben aplicar, para resolver los requerimientos, y los hechos expresan relaciones entre objetos.

Las reglas describieron relaciones de eventos y situaciones relativas al medio tecnológico como, por ejemplo, “El inicio tarda más de 15 minutos entonces un disco duro puede estar presentando problemas”, “El inicio del Windows es demasiado lento, pero aun así arranca, revise la capacidad del disco C” o “Si el computador no tiene memorias RAM entonces el computador no enciende”. Afirman el hecho de que el computador “tiene memoria RAM” y “tiene disco duro”.

Estas reglas contienen información importante como: recomendaciones, diagnósticos y síntomas por dispositivos, orden y secuencia lógica, además de un lenguaje común para mejor entendimiento a los usuarios (Cabello, E, Eugenia M, 2018). Los datos requeridos por el sistema fueron problemas o síntomas de los dispositivos, por ejemplo: dificultad al leer los códigos de barras, incompatibilidad de los controladores, daño en la memoria o pérdida de información entre otros. Se muestran ejemplos en la tabla 2:

Tabla 2: Adquisición y representación

Dispositivo: “Impresora”
<p>Reglas: Para evitar impresiones borrosas o manchadas configure la página antes de imprimir, ajustando una baja calidad para documentos de texto, Si las impresiones muestran líneas torcidas, seleccione propiedades de la impresora, realice el mantenimiento llamado ‘alineación’ o ‘limpieza de cabezales’ y siga las instrucciones, si el cable o puerto están dañados, cámbielos, Si la impresora es inalámbrica, compruebe que la red no tenga problemas. Si persiste podría ser el controlador de la impresora, que esté inactivo.</p>
<p>Hechos: El software de la impresora está ocasionando problemas, La impresión aparece borrosa, Desalineación de textos, colores e imágenes en las impresiones, El computador no reconoce la impresora, impresora muestra mensaje de error o notificación por sonidos, Al sustituir la tinta la impresora sigue sin imprimir, La impresora no enciende, El papel se atasca en la impreza.</p>

Fuente Propia

Formalización: Representa la transformación de los hechos y reglas integrados a la base de conocimiento. de un lenguaje natural a un lenguaje formal, lógica de primer orden. Es decir, la forma en la cual el conocimiento fue organizado y representado para determinar la estructura en la que están relacionados lógicamente los recursos tecnológicos de la empresa. Por ejemplo, una regla que generó un hecho fue “todo dispositivo necesita energía”, “El computador es un dispositivo”, por lo tanto, “el computador necesita energía”. Formalizando esta frase se puede representar “ $\forall X$, dispositivo(X) \rightarrow eléctrico (X)”, expresando que todo dispositivo necesita energía para funcionar.

En esta etapa se realizó integración entre la adquisición del conocimiento formal y la representación lógica. Ya que varias piezas de software y hardware muestran relación, describiendo el conocimiento sobre el entorno tecnológico, eventos entre diagnósticos, problemas, procedimientos y recomendaciones. Se muestran ejemplos en la tabla 3:

Tabla 3: Formalización

Sintaxis de alfabeto formal
Reglas: $\forall i$, impresora (i) \wedge eléctrico (i) \wedge digital (i) \rightarrow dispositivo (i) $\forall p$, pantalla (p) \wedge eléctrico (p) \wedge análogo (p) \rightarrow (p) dispositivo
Hechos: impresora (Canon Maxify) eléctrico (Canon Maxify) pantalla (LG Flatron) digital (LG Flatron) impresora (Epson TX525FW) análoga (Epson TX525FW) enciende (Epson TX525FW)”

Fuente Propia

Implementación: Esta etapa involucra la programación del conocimiento en el computador. Siguiendo el orden del modelo, dentro del motor de inferencia (Prolog) se convirtió esta información de primer orden a la sintaxis del lenguaje de programación.

Se escribieron reglas de producción con el patrón, **“Si entonces <Procedimientos, recomendaciones, diagnósticos >”**. Por ejemplo, **“Si las impresiones muestran líneas torcidas, entonces seleccione propiedades de la impresora, realice el mantenimiento llamado ‘alineación’ o ‘limpieza de cabezales’ y siga las instrucciones”**. Así cada regla **“Si entonces”** estableció un aporte de información a la base de conocimiento.

Tabla 4: Base de conocimiento

<pre> /* Ram */ Id_imagen_preg (“la pantalla no enciende y se escucha 3 pitidos cortos”, “pitido”). Id_imagen_preg (“se escucha un pitido largo y la pantalla no enciende”, “pitido2”). Id_imagen_preg (“el sistema operativo lanza mensajes de error, fallas de protección y se reinicia continuamente”, “soerror”). Id_imagen_preg (“el sistema operativo se detiene de forma inesperada y lanza falla de protección general”, “sodetiene”). Id_imagen_preg (“el sistema operativo no funciona adecuadamente o se apaga por completo”, “pitido”). /* Base de conocimiento */ Conocimiento (‘ram’, ‘la pantalla no enciende y se escucha 3 pitidos cortos’, ‘se escucha un pitido largo y la pantalla no enciende’, ‘el sistema operativo lanza mensajes de error, fallas de protección y se reinicia continuamente’, ‘el sistema operativo se detiene de forma inesperada y lanza falla de protección general’, ‘el sistema operativo no funciona adecuadamente o se apaga por completo’). </pre>
--

Fuente Propia

El motor de inferencias elige que reglas ejecutar (posibles soluciones, recomendaciones y diagnósticos). Las consecuencias son algunos de los eventos a verificar a partir de los hechos, ya que las reglas deciden las operaciones que aplicar, para resolver dichas consecuencias, y los hechos expresan relaciones entre objetos (Tab 4).

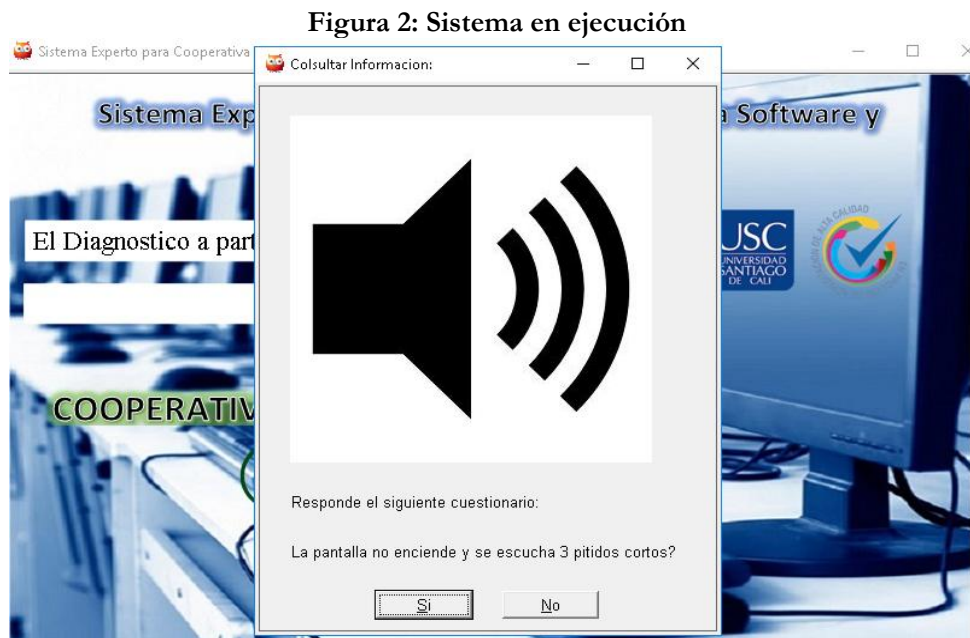
Tabla 5: Estructura del sistema (Prolog)

<pre> /* Disco duro */ resource (discoduro, image, image (“sol_disco.jpg")). resource (solento, image, image (“solento.jpg")). resource (errorre, image, image (“errorre.jpg")). resource (pantazul, image, image (“pantazul.jpg")). resource (errorgrabar, image, image (“errorgrabar.jpg")). resource (scandisc, image, image (“scandisc.jpg")). resource (sonmeta, image, image (“sonmeta.jpg")). </pre>

Fuente Propia

Una vez construida toda la estructura del sistema, se verificó que cada diagnóstico y recomendación presentada fuera correcto según los hechos y las reglas establecidas. La programación comenzó por asignar un identificador principal por dispositivo como ejemplo “discoduro”, todos los hechos se encuentran almacenados en el recuso “sol_disco.jpg” y reglas

están distribuidas por otros recursos dentro del mismo esquema como “solento”, “pantazul.jpg” entre otros (Tab 5).



Fuente Propia

La interfaz gráfica consta de ventanas a modo de cuestionario para que el usuario ingrese al sistema, debe seleccionar “Si” o “No” según la situación. Una vez seleccionado una serie de respuestas, el sistema presenta una serie de eventos que pueden presentar los dispositivos, dependiendo de la selección del usuario y según las reglas y hechos establecidos en la base de conocimientos (BC), el sistema presenta la descripción detallada del daño y las intervenciones que se pueden llevar a cabo para solucionarlas (Fig. 2).

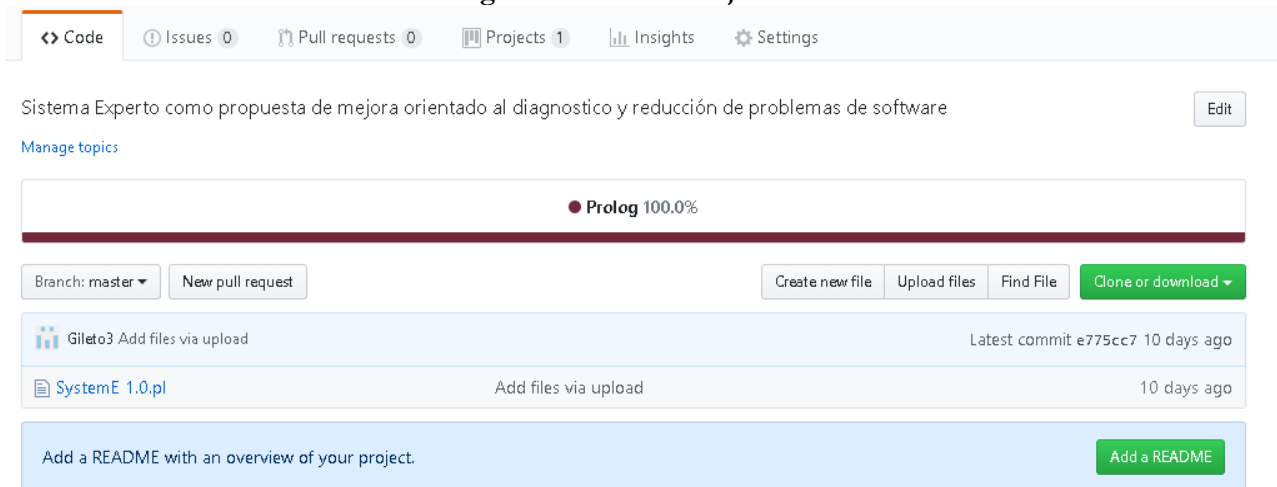
Se realizaron pruebas con los empleados de la Cooperativa con el objetivo de obtener observaciones y sugerencias acerca del prototipo desarrollado para examinar la validez del conocimiento.

3. RESULTADOS Y DISCUSIONES

Se realizaron pruebas y validaciones con testers en línea que son herramientas de apoyo a programadores de software y sirven para identificar o verificar la definición de los requerimientos y riesgos como: errores en ejecución, dificultad en la usabilidad y funcionalidad, edición en línea e instalación de librerías y software en diferentes arquitecturas o sistemas operativos, entre otros.

Se utilizaron las herramientas GitHUB, FireBug, NotePad ++, que pueden alojar, administrar y revisar proyectos de software además de depurar su funcionamiento y operatividad, sin embargo, no arrojaron los resultados esperados. Pese a que las herramientas reconocieron la sintaxis de Prolog como el lenguaje origen del sistema experto, no pudieron depurarlo. Se descubrió que estas herramientas freelance carecen de ciertos archivos externos que no permiten ejecutar esta funcionalidad.

GitHub es una herramienta web que sirve para guardar y ejecutar proyectos de software, código fuente y versiones. Este programa ofreció características como seguimiento de errores, operación de tareas, edición en línea, y otra función de redes sociales como share problems, feed and wikis, que ayudaron al ingeniero de conocimiento a realizar pruebas colectivamente al desarrollo del software (Fig. 3).

Figura 3: GitHub en ejecución

Fuente Propia

Firebug es una extensión o paquete del navegador web Mozilla que ofrece utilidades con las que se puede analizar, editar, monitorear y depurar el código fuente de diferentes lenguajes de programación de manera instantánea y en línea. Aspecto que sirvió al ingeniero de conocimiento en la depuración, edición y supervisión de las líneas de código (Usabilidad, 2015). También es utilizado para identificar librerías software. Se instaló originalmente como una extensión para Mozilla pero, también se pudo agregar a Chrome.

Por último, Notepad ++ que permitió editar el código fuente de Prolog en entorno Windows. Además de resaltar la alineación del código, ediciones sincronizadas entre otras funcionalidades (++Npad, 2019). Notepad ++ es un software gratuito de código abierto utilizado para edición de código escrito en diferentes lenguajes de programación, como Java, Visual, Prolog y Python entre otros, es uno de los editores más conocidos y utilizado por los programadores de software, ya que ofrece diferentes herramientas utilitarias adicionales a sólo interpretar la sintaxis del código, tiene una Interfaz de usuario completamente personalizable que sirvió para simplificar la redacción y orden del código (Tab 6).

Tabla 6: NotePad++ en ejecución

```

botones:-borrado,
    send (@boton, free),
    send(@btntratamiento,free),
    mostrar_diagnostico(Problema),
    send (@texto, selection('El Diagnostico a partir de los datos es:')),
    send (@resp1, selection(Problema)),
    new(@boton, button('INICIAR TEST',
    message(@prolog, botones) )),

    new(@btntratamiento,button('DETALLES',
    message(@prolog, mostrar_tratamiento, Problema))),
    send(@main, display,@boton,point(20,450)),
    send(@main, display,@btntratamiento,point(138,450))

mostrar_tratamiento(X):-new(@tratam, dialog('Solucion')),
    send (@tratam, append, label(nombre, 'Explicacion: ')),
    send(@tratam, display,@lblExp1,point(70,51)),
    send(@tratam, display,@lblExp2,point(50,80)),
    tratamiento(X),

```

Fuente Propia

Finalizada la implementación y los testers se procedió a realizar las primeras pruebas de usabilidad. Se categorizaron los usuarios que emplearon el sistema experto y se observaron cualidades fundamentales: el conocimiento, la frecuencia y experiencia que tenían a la hora de manejar herramientas de software. Utilizando esta estrategia se plantearon 2 tipos de usuarios de prueba: los directivos de la empresa (gerente y tesorero) y los operarios. Los grupos de prueba pudieron adecuarse al sistema propuesto pese a que no se trataba de una aplicación común, de las que manejan en la rutina laboral. A continuación, se presentan los tipos de usuarios según el perfil de la Cooperativa: el novato (operarios), es un usuario que tiene poco conocimiento de las herramientas tecnológicas e informáticas y cuya interacción con el sistema que se construyó es limitada. Dedicaba entre un 5 % y 30 % de sus actividades a solucionar problemas para los cuales el software pueda ofrecer recomendaciones o posibles soluciones. Debido a esto, se encontraron observaciones del sistema en cuanto a la usabilidad ya que tardaron un poco en responder los cuestionarios hasta entender cómo funciona el árbol de respuestas y poder llegar a una recomendación o posible solución.

El usuario intermedio (tesorera, gerente) utiliza con frecuencia el computador para sus actividades rutinarias. Dedicaba entre un 30% y 60% de sus actividades a solucionar problemas computacionales relacionados al mantenimiento u otras situaciones como, por ejemplo, el mantenimiento del antivirus, actualización de controladores o liberación de espacio en el disco duro. A este usuario se le brindó previa instrucción de uso, principalmente en cómo responder los cuestionarios para poder aproximarse a una posible solución. Se observó un mejor desempeño, sin embargo, la sugerencia más destacada era el orden y empalme de las preguntas para llegar a las recomendaciones o posibles soluciones. Vale la pena aclarar, que este usuario no está capacitado para realizar algunas operaciones que sugiere el sistema experto.

Basado en la observación de los usuarios con respecto al uso de las tecnologías en su trabajo se pudo verificar que la interfaz es intuitiva, ya que los usuarios hicieron uso de su función básica. Las pruebas se realizaron con los empleados de la Cooperativa cuyo nivel académico máximo es técnico. Estas pruebas se hicieron durante 4 semanas laborales, fueron repetidas 1 vez por día y requirieron el sistema experto en ejecución; tuvieron una duración entre 10 a 20 minutos de práctica, bajo el marco de 30 reglas y 30 hechos incorporados en la base de conocimiento. Adicionalmente, cuenta con 40 recomendaciones para los dispositivos de la empresa, obtenidas principalmente del proveedor **TL Usados Store** en la ciudad de Cali, que es el vendedor de tecnología encargado de la instalación, mantenimiento, soporte técnico y servicio integral de la base tecnológica en la Cooperativa, además de manuales y documentos técnicos de los dispositivos.

Los primeros días surgieron varias observaciones al interactuar con la herramienta, ya que la manera cómo estaban relacionados los hechos y reglas no era adecuada, impidiendo a los empleados acertar los diagnósticos. Como ventaja de los sistemas expertos, se pudo reorganizar la información (hecho, reglas y recomendaciones) para orientar al empleado de la secuencia de los cuestionarios. Se optó por cambiar la estructura basada en problemas de los dispositivos a una estructura basada sólo en los problemas específicos, es decir, que cada cuestionario y diagnóstico sigan la misma secuencia para facilitar la interpretación al usuario (Ponce,2010). Por ejemplo, los cuestionarios: “La impresión aparece borrosa?” y “La impresión muestra desalineación de textos, colores e imágenes en las impresiones?”, están relacionadas como problemas de impresora y llevan una secuencia lógica. En cuanto a los diagnósticos o recomendaciones fueron “Para evitar impresiones borrosas ajuste la impresora a una baja calidad para documentos de texto e imágenes” y “para evitar desalineación de texto e imágenes realizar limpieza y alineación de cabezales”.

Posterior a las correcciones se reiniciaron las pruebas dinámicas y de aceptación, obteniendo observaciones positivas respecto al cambio de lógica. Sin embargo, al llegar a un posible diagnóstico y recomendaciones resaltaron que las posibles soluciones resultaban un poco complejas de ejecutar para su nivel académico, siempre destacando la gran utilidad y el impacto positivo que causa en la organización a la hora de tomar decisiones respecto a reparaciones, cambio de dispositivos e incluso actualización de aplicaciones y controladores. Se modificó el sistema en cuanto sus recomendaciones cambiando el lenguaje técnico y así facilitar el entendimiento de las sugerencias que genera el sistema.

Este enfoque implicó la reutilización de paquetes de software, mejorar la programación y el tiempo de ejecución del proceso de desarrollo. Para entender el enfoque, se consideró que los sistemas expertos basados en hechos y reglas sobre

el dominio de software y hardware son una prueba de conceptos, ya que la información se compila por medio de experiencias y fuentes confiables (CarrilloVerdu, 1987). El uso del sistema permitió reducir el tiempo invertido para solucionar problemas de software y hardware, permitiendo la continuidad del negocio.

El sistema muestra como salida, una serie de recomendaciones, actividades o estrategias que orientan al empleado sobre la forma cómo puede interceder ante el problema o falla que se esté presentando (Tabla. 7).

Tabla 7: Recomendaciones

Soluciones:
-Confirme los ajustes de relación del código de barras de la impresora y también compruebe que se está imprimiendo en la presión deseada.
-Considere realizar la impresión con un color que tenga buen contraste respecto a la luz roja.
-Colocar una pieza de papel blanco cerca del código de barras, Si la lectura es estable, realice este proceso temporalmente y especule otra solución.
-El cartón es un objeto con muchas condiciones inconsistentes de superficie. Se requieren pruebas suficientes para garantizar la consistencia del lector de código de barras.
-Intente colocar dos piezas de papel blanco en el borde del código de barras. Si se puede leer, entonces se debe agregar una zona inactiva adecuada al código de barras.

Fuente Propia

Las principales ventajas de los sistemas expertos hacen referencia a la disminución de costos, ya que suprimen la necesidad de acudir a expertos humanos en un área determinada de conocimiento, son permanentes y carecen de emociones humanas ya que están integradas a los computadores, maquinas o sistemas artificiales, las cuales son fuentes constantes de error. Por otra parte, sus principales desventajas son la falta de sentido común, ya que no hay nada obvio para ellos y no tienen la capacidad de almacenar la experiencia tal y como lo hacen los humanos (Adarraga & Zaccagnini, 1988).

Los sistemas expertos ofrecen gran capacidad para gestionar grandes cantidades de información, que son uno de los problemas que enfrentan los profesionales en su rutina laboral, factor que puede afectar la continuidad del negocio y la toma de decisiones críticas; pues los empleados pueden desechar datos que no considere relevantes, mientras que un sistema experto, debido a su gran capacidad de procesamiento, analiza toda la información incluyendo desde soluciones artesanales hasta propuestas de nivel técnico profesional, para de esta manera, tomen una decisión más sólida (Autodata-group, 2017).

En esta sección, se han dado criterios de desempeño para probar el método propuesto. Como refiere Ghaderzadeh, Sadoughi, & Ketabat (2013) por lo general, esto implica pruebas de alguna descripción. Un chequeo al grupo de pruebas de software. Las decisiones sobre las recomendaciones se toman sobre la base de los resultados de las pruebas. Estas pruebas son usabilidad, funcionabilidad, testers, valores del diagnóstico entre otros.

Los resultados de usabilidad son buenos debido a que la aceptación de los empleados fue positiva, les agradó las posibilidades que ofrece este sistema y lo tendrán como primera opción ante un acontecimiento o contingencia que permita usarla en el futuro. La validación del sistema fue realizada por el experto y herramientas de apoyo para pruebas, permitiendo explorar su funcionamiento y hacer correcciones en el diseño y contenido de este.

Otro aporte importante es que las descripciones o soluciones son cercanas a lo que soporte empresarial indicaría acerca del estado o síntomas de los componentes, dado que las frases han sido construidas usando conocimiento del dominio y la flexibilidad de una interfaz amigable; que permita comprender mejor al usuario lo que el sistema experto le está presentando (Generation, 2011).

Los avances tecnológicos dieron lugar al desarrollo de herramientas de apoyo, facilitando el proceso de toma de decisiones para el caso estudio Cooperativa, disminuyendo el riesgo de incurrir en errores o valoraciones incorrectas respecto a los elementos de software y hardware (In, 2015). La primera versión del prototipo ha sido probada en

diferentes computadores, tanto computadores de mesa como en portátiles, con diferentes configuraciones, y refleja buena manejabilidad.

Se puede decir que el prototipo funciona correctamente sin necesidad de características técnicas del equipo demasiado grandes. El software funciona con un equipo de cómputo mínimo, con un procesador Intel Celeron 2.6 GHZ, con 2 GB en RAM, un disco duro con un espacio disponible de 40 Gb, una tarjeta de video VGA y un monitor de 15" VGA. Se utilizó como herramienta de desarrollo la suite de trabajo Visual Prolog, en S.O Windows 10.

3. CONCLUSIONES

El diseño de un sistema experto requiere de una relación permanente entre el proveedor experto y el ingeniero de conocimiento, para extraer y representar la información que se plasmará en el software; ya que, por medio del proveedor, el ingeniero de conocimiento interpreta los manuales de uso y documentos técnicos de los dispositivos e integra la base de conocimiento.

La fortaleza de un sistema experto es su base conocimiento conformada por hechos y reglas, siempre y cuando haya sido estructurada y modelada adecuadamente para representar la experiencia y el conocimiento del experto en un dominio específico. Inicialmente el orden de las sentencias se enfocaba en los problemas de los dispositivos y no en los eventos o situaciones que estuvieran ocurriendo, dificultando el entendimiento al usuario, para solucionar el problema se cambió el orden.

Se pudo destacar que con el uso de la metodología Buchanan se aportan elementos que permiten identificar los comportamientos más relevantes en determinada situación, y cuál es la mejor manera de interceder, de forma que sea un aporte a los usuarios en sus funciones laborales; esto resulta interesante para replicar esta metodología en otros dominios con complejidad mayor.

El sistema propuesto en el área de la informática no es una herramienta para reemplazar al soporte técnico en sus labores de realizar mantenimiento preventivo o correctivo, sino que contribuye y apoya la toma de decisiones a la hora de brindar posibles soluciones. Es una herramienta útil que evoluciona constantemente ante los avances de la tecnología, la Ingeniería y otras ciencias, incursionando en nuevos temas y ayudando a empleados o profesionales realizar operaciones que no tienen que ver directamente con su función.

Se pudo comprobar que la utilización del prototipo es intuitiva, ya que personas con entendimiento básico de aplicaciones software y lectura intermedia, son capaces de utilizar e interactuar con el software por medio de una serie de pasos establecidos sin necesidad de la ayuda de la persona que realizó el desarrollo.

El sistema desarrollado permite realizar un diagnóstico de manera rápida y objetiva. Un problema que podía tardar horas en identificarse, mediante la herramienta es posible hacerlo en minutos, las futuras versiones que se deriven de este primer desarrollo deberán aumentar el número de reglas y hechos para incluir más eventos o situaciones en la base de conocimiento. Como línea futura quedará abierto a obtener futuras peticiones, se definen nuevos requerimientos y el prototipo se modifica, retomando el proceso a partir de la fase de implementación, además podrá realizar operaciones más complejas y diferentes características de trabajo.

Esta investigación representa un sistema experto como una herramienta que puede integrarse a la manera de operar de pequeñas empresas con las características de la Cooperativa, es decir en las cuales el conocimiento experto cada vez es más importante para proponer nuevas alternativas orientadas al futuro.

4. REFERENCIAS

Acosta Caicedo, M. Y., Arteaga Cabrera, I. G., González Serrano, C., & López Gutiérrez, D. M. (2011). Sistema Experto para el desarrollo de soluciones de interoperabilidad en Sistemas de Información para Laboratorios Clínicos basado en el Estándar Internacional HL7*. *Revista Virtual Universidad Católica Del Norte*, 32(32), 73–103. Retrieved from <http://revistavirtual.ucn.edu.co/index.php/RevistaUCN/article/view/27/58>

Autodata-group. (2017). AUTODATA. Retrieved November 14, 2017, from <https://www.autodata-group.com/es/>

- Carballo, L. (2015). Revisión de metodologías para el desarrollo de Sistemas Expertos. Retrieved from virtual.udabol.edu.bo/mod/resource/view.php?id=31853
- C. Elena, & Luque, N. (2019). La toma de decisiones en la informática jurídica basado en el uso de los sistemas expertos, 40(1), 131–139
- Chilcanan, D., Navas, P., & Escobar, S. M. (2017). Sistema Experto para la Automatización de Procesos Remotos en Servidores Multiplataforma, Mediante la Conversación Hombre Máquina. In Iberian Conference on Information Systems and Technologies, CISTI. <https://doi.org/10.23919/CISTI.2017.7975913>
- Elena, C., & Luque, N. (2019). La toma de decisiones en la informática jurídica basado en el uso de los sistemas expertos, 40(1), 131–139..
- Espinosa Cabello, Eugenia María, Preciado Álvarez, Francisco, Santana Álvarez, Oscar Alberto, Ramos Salavert, I. (2018). Una aproximación MDE para generar una línea de productos de sistemas expertos. Article, (Revista de la Alta Tecnología y Sociedad. 2018, Vol. 10 Issue 1, p26-42. 17p.).
- Flores, V., & Hadfeg, Y. (2017). Un método para generar explicaciones de resultados de un Sistema Experto, usando Patrones de discurso y Ontología. RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao, (21), 99–114. <https://doi.org/10.17013/risti.21.99-114>
- Fuentes Covarrubias, R., Gerardo, A., & Covarrubias, F. (2013). Desarrollo de un sistema experto para el diagnóstico de fallas automotrices. Retrieved from http://sedici.unlp.edu.ar/bitstream/handle/10915/32398/Documento_completo.pdf?sequence=1
- Garay-garcell, M. (2015). Interfaces Inteligentes en el aprendizaje de la Modelación Intelligent Interfaces in modeling learning, XXXVI(2), 187–201.}
- Generation, A., Software, O. F., From, A., Mda, S., On, B., Method, T. H. E., & Artificial, O. F. (n.d.). Generación Automática De Aplicaciones Software a Partir Del Estandar Mda Basándose Expertos E Inteligencia Artificial Automatic Generation of Software Applications From Standard Mda Standard, 10, 67–72.
- Ghaderzadeh, M., Sadoughi, F., & Ketabat, A. (2013). A Computer-Aided Detection System for Automatic Classification of Prostate Cancer from Benign Hyperplasia of Prostate, 2(2), 1–5.
- Giattatano, J., Riley, G., & Pineda, E. (2001). Sistemas expertos : principios y programación. International Thomson. Retrieved from <https://catalogo.konradlorenz.edu.co/cgi-bin/koha/opac-detail.pl?biblionumber=14448#.WpQwMsty9fs.mendeley>
- Gil-Vera, V. D. (2016). Sistema Experto para la Gestión de Daños en Vías Pavimentadas y en Afirmado. Lámpsakos, (15), 22. <https://doi.org/10.21501/21454086.1710>
- In, I. S., Of, C., & Meteorological, A. (2015). Sistema experto para tomar decisiones de emergencias y seguridad ante meteorología adversa, 90.
- L.Gutierrez, H. T. (2013). Identification of Camputer Viruses Using Expert Systems (p. 9).
- Montiel, L., & Riveros, V. (2014). Los sistemas expertos en el ámbito educativo. Omnia. Retrieved from <http://www.produccioncientifica.luz.edu.ve/index.php/omnia/article/view/5549>}
- Norvig, P., & Russell, S. (2014). Inteligencia artificial. Elsevier Brasil. (Vol. 1). <https://doi.org/M-26913-2004>
- Ponce, P. (2010). Inteligencia artificial con aplicaciones a la ingeniería.

Rolston, D. W., & Pérez Gama, A. (1990). Principios de inteligencia artificial y sistemas expertos. McGraw-Hill. Retrieved from https://catalogo.konradlorenz.edu.co/cgi-bin/koha/opac-detail.pl?biblionumber=6628&query_desc=su:%22INTELIGENCIA+ARTIFICIAL%22+and+su-to:INTELIGENCIA+ARTIFICIAL+and+itype:BK#.WpQvrRYYmY4.mendeley

Roussel, A. C. y P. (1995). PROLOG. Retrieved from <http://www.swi-prolog.org/>

Serrano Climent, S. (1998). Sistemas expertos probabilísticos. Universidad De Valencia, 301.

Usabilidad, D. (2015). Capítulo 4; 57–68.

V. Flores,& Hadfeg, Y. (2017). Un método para generar explicaciones de resultados de un Sistema Experto, usando Patrones de discurso y Ontología. RISTI - Revista Iberica de Sistemas e Tecnologias de Informacao, (21), 99–114. <https://doi.org/10.17013/risti.21.99-114>

Verdu Carrillo, J. D. (1987). R3Z .

++ N. (2019). Notepad ++. Retrieved April 9, 2019, from <https://notepad-plus-plus.org/>