

REPUBLICA BOLIVARIANA DE VENEZUELA
MINISTERIO DEL PODER POPULAR PARA LA EDUCACION UNIVERSITARIA,
CIENCIA Y TECNOLOGIA
UNIVERSIDAD “VALLE DEL MOMBOY”
CARVAJAL ESTADO TRUJILLO



**SISTEMA EXPERTO PARA EL DIAGNÓSTICO Y FALLA DE LA
RED DE SERVIDORES DE LA UNIVERSIDAD VALLE DEL
MOMBOY**

AUTOR:
JOSÉ LUIS MATHEUS SIVOLI. CI: 25.631.792

TUTOR ACADÉMICO:
DR. IVÁN PEREZ

Carvajal, Enero de 2019

REPUBLICA BOLIVARIANA DE VENEZUELA
MINISTERIO DEL PODER POPULAR PARA LA EDUCACION UNIVERSITARIA,
CIENCIA Y TECNOLOGIA
UNIVERSIDAD “VALLE DEL MOMBOY”
CARVAJAL ESTADO TRUJILLO



SISTEMA EXPERTO PARA EL DIAGNÓSTICO Y FALLA DE LA RED DE SERVIDORES DE LA UNIVERSIDAD VALLE DEL MOMBOY

(Trabajo Especial de Grado para optar por el Título de Ingeniero en Computación)

AUTOR:
JOSÉ LUIS MATHEUS SIVOLI. CI: 25.631.792

TUTOR ACADÉMICO:
DR. IVÁN PEREZ

Carvajal, Enero de 2019



REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD VALLE DEL MOMBOY
FACULTAD DE INGENIERÍA
CARVAJAL – EDO TRUJILLO

CARTA DE ACEPTACIÓN DE TUTOR

Ciudadano (a)

Director(a) CIDIFI

Presente.-

Me dirijo a usted en la oportunidad de informarle que yo **Iván Pérez**, portadora de la Cédula de Identidad N° **V-4.884.756**, de profesión Ingeniero en Computación, he aceptado la responsabilidad de servir de tutor al Bachiller: **Matheus José**, C.I. **V-25.631.792** en su Trabajo Especial de Grado titulado: **SISTEMA EXPERTO PARA EL DIAGNÓSTICO Y FALLA DE LA RED DE SERVIDORES DE LA UNIVERSIDAD VALLE DEL MOMBOY.**

Dando fe de lo expuesto a los 7 días del mes de Febrero de 2019

Bachiller

Tutor



REPÚBLICA BOLIVARIANA DE VENEZUELA
UNIVERSIDAD VALLE DEL MOMBOY
FACULTAD DE INGENIERÍA
CARVAJAL – EDO TRUJILLO

CARTA DE APROBACIÓN DEL TUTOR

Ciudadano(a)

Director(a) CIDIFI

Presente.-

Me dirijo a usted en la oportunidad de informarle que el Trabajo Especial de Grado que presentan los bachilleres: **Matheus José**, C.I V-25.631.792, titulado: **SISTEMA EXPERTO PARA EL DIAGNÓSTICO Y FALLA DE LA RED DE SERVIDORES DE LA UNIVERSIDAD VALLE DEL MOMBOY**. Lo considero listo para ser presentado y defendido ante el jurado evaluador que la universidad convenga en nombrar para tal fin. En tal sentido, solicité formalmente señalar la fecha, hora y sitio para efectuar la defensa correspondiente.

Agradeciendo su atención.

Fecha: ____/____/____

DEDICATORIA

A mis padres, que han estado apoyándome durante mi carrera académica, dando siempre su mayor esfuerzo por brindarme un soporte fundamental en mi vida.

A mi hermana, quien ha estado en los buenos y malos momentos de mi vida, además de enseñarme importantes lecciones de vida, por su apoyo, comprensión, paciencia, y su atención en todo momento.

A mi novia, por su apoyo, consejos, preocupación y su más cálido amor, que me ha motivado mucho a darle continuidad a mis proyectos, he aprendido muchísimo a tu lado.

A mi mascota Gary, que es un pilar fundamental en la familia, por su amor y su ternura que me han hecho sentir conmovido.

AGRADECIMIENTOS

A mis padres por preocuparse, darme su apoyo y dedicación a lo largo de mi vida, enseñándome importantes lecciones y formas de ver el mundo.

Un profundo agradecimiento a mi Jefe Andreas Buechel, que sin su apoyo a lo largo de mi carrera nada de esto sería posible, una gran persona que sin duda tengo mucho que agradecerle.

A mis profesores de la Universidad Valle Del Momboy que me formaron académicamente, impartiendo su conocimiento del cual estoy muy agradecido.

-José L. Matheus S.

RESUMEN

El presente trabajo especial de grado tuvo como finalidad la creación de un sistema experto que permita responder de manera oportuna a los problemas presentados con la red de servidores de la Universidad Valle Del Momboy, dirigido principalmente a el centro de cómputo de la sede Mirabel ubicado en la ciudad de Valera, con la finalidad de presentar una solución a dichos problemas que afectan distintos servicios ofrecidos por la Universidad. Se apoya en los sistemas expertos basado en reglas y encadenamiento hacia delante, de inteligencia artificial que son capaces de procesar la información de manera dinámica a través de la base de conocimiento. Se enmarca dentro de las investigaciones del tipo Proyecto factible y su diseño documental de campo. Para la ejecución de este proyecto se llevó a cabo mediante una encuesta al personal del centro de cómputo y estudiantes de la facultad, analizando los datos recolectados con el fin de elaborar un diseño del prototipo; que daría como resultado un sistema experto para el diagnóstico de fallas en la red de servidores de la Universidad Valle Del Momboy.

Palabras Clave: Sistema Experto, Inteligencia Artificial.

INDICE

Contenido

INTRODUCCIÓN	1
CAPÍTULO I	3
Planteamiento del problema.....	3
Objetivos general.....	6
Objetivo Específico.....	6
Justificación.....	7
Delimitación.....	8
CAPÍTULO II	9
MARCO TEÓRICO	9
Antecedentes del sistema experto.....	9
Bases Teóricas.....	13
Conceptos básicos para el conocimiento de un sistema experto y sus atributos.....	13
Conceptos básicos para la implementación de un sistema experto para el diagnóstico de falla en redes de servidores.....	19
Conceptos básicos para conocer estabilidad en el sistema y alternativas viables.....	22

Monitorización general.....	25
Tipos de monitorización.....	25
Objetivos que se deben monitorizar.....	26
Ancho de banda de una red Ethernet.....	27
Metodologías del desarrollo de software.....	29
Componentes de la metodología.....	29
Ciclo de vida de la metodología.....	31
Modelo de ciclo de vida de software.....	32
Análisis.....	33
Diseño.....	34
Codificación.....	35
Pruebas.....	35
Implementación.....	36
Solución de problemas.....	37
Formular el problema.....	37
Algoritmo.....	38
Algoritmos genéticos.....	40

Vocabulario en Algoritmos Genéticos.....	41
Componentes de un Algoritmo Genético.....	41
Algoritmos evolutivos.....	43
Estructura general de los algoritmos evolutivos.....	43
Programación Genética.....	45
DEAP (software) como herramienta para la programación genética.....	46
Optimización por enjambre de partículas.....	46
Programación genética haciendo uso de la herramienta DEAP.....	47
Desarrollo de los sistemas expertos.....	50
Clasificación de herramientas de desarrollo según etapa de proyecto en que se usan.....	50
Tipos de representación del conocimiento.....	51
Ejemplos de representación.....	51
Lenguajes y generadores de sistemas expertos.....	52
Integración y migración de los sistemas expertos.....	53
Tendencias futuras de los sistemas expertos.....	53
Campos actuales de investigación y de aplicación.....	54
Aplicaciones de los sistemas expertos.....	54

Simulación.....	55
Ejecutivo del planteamiento de las tres fases.....	57
Corriendo como Sistema Experto.....	57
CAPÍTULO III.....	58
MARCO METODOLÓGICO.....	58
Tipo y Diseño de la Investigación.....	59
Diseño de la investigación.....	59
Población.....	60
Muestra.....	60
Técnica.....	62
Instrumento.....	63
Validación del instrumento.....	64
Confiabilidad.....	65
Técnicas de procesamiento y análisis de datos.....	65
Metodología seleccionada para el desarrollo del sistema experto.....	66
CAPÍTULO IV.....	68
ANÁLISIS DE LOS RESULTADOS.....	68
ANÁLISIS DE LOS RESULTADOS DEL INSTRUMENTO.....	69
CAPÍTULO V.....	79
DESARROLLO DEL SOFTWARE.....	79
Primera fase: Análisis.....	79
Segunda fase: Diseño.....	80

Tercera fase: Codificación.....	81
Cuarta fase: Pruebas.....	81
Quinta fase: Implementación.....	81
CAPÍTULO VI.....	83
CONCLUSIONES Y RECOMENDACIONES.....	83
BIBLIOGRAFÍA.....	85
ANEXOS.....	86
Anexo 1. Instrumento.....	87
TABLA DE FIGURAS.....	89
TABLA DE EJEMPLOS.....	90

INTRODUCCIÓN

Desarrollar un SE (Sistema Experto) que permita el diagnóstico de las fallas y problemas que se puedan presentar en una red de servidores para corregir o prevenir estos inconvenientes es la principal finalidad de este proyecto, su propósito general es darle un óptimo desempeño a los servidores de la Universidad Valle Del Momboy que a su vez se mantiene en correcto funcionamiento todos los servicios y procesos de red alojados en los servidores, con la aplicación de un sistema experto basado en algoritmos evolutivos e inteligencia artificial, que cumpla con el soporte y ayuda que pueda ofrecerle al usuario en sustitución de una persona experta.

Los sistemas expertos son una herramienta poderosa en el apoyo o guía de los usuarios en los procesos que tienen una secuencia de pasos definida, pero que puede ser configurable. La presente investigación aborda el tema de sistemas expertos, debido a que son una herramienta cuya utilidad ya está comprobada, y que sin embargo, muchas personas desconocen y otras más no las aceptan todavía. Se desean presentar los aspectos generales de un sistema experto, sus orígenes, componentes, construcción y utilización primordialmente. Es un sistema basado en el conocimiento que, mediante el buen diseño de su base de información y un adecuado motor de inferencias para manipular dichos datos proporciona una manera de determinar resoluciones finales dados ciertos criterios.

El trabajo se estructuró en seis capítulos, cuyos contenidos se describen a continuación:

En el Capítulo I llamado “El problema”, se expone el objeto de estudio, su planteamiento y formulación, junto con los objetivos del trabajo investigativo, seguido de la justificación e importancia y su delimitación

En el capítulo II lleva como título “Marco Teórico” y aquí se desarrollan los antecedentes del estudio, las bases teóricas que se componen de teorías en algoritmos, desarrollo de software, inteligencia artificial, y distintos conceptos que desprende estos temas que complementan la ejecución del proyecto.

En el Capítulo III que lleva por nombre “Marco Metodológico” expone los aspectos técnicos de la investigación como lo son el método y diseño de investigación que se llevó a cabo; la población y muestra de estudio; las técnicas e instrumentos de recolección de datos y la metodología que se utilizó para desarrollar el proyecto.

En el Capítulo IV, que lleva por nombre “Análisis de los resultados”, presenta los datos recolectados a través del instrumento aplicado para la recolección de información que posteriormente sería usada para el desarrollo del proyecto.

En el Capítulo V, denominado “Desarrollo de Software”, desprende cinco fases que se aplican en el desarrollo y creación del sistema experto.

Por último el Capítulo VI, se exponen las conclusiones y recomendaciones que atiende los objetivos puestos.

CAPÍTULO I: EL PROBLEMA

Planteamiento del problema

Los servidores son una herramienta fundamental para las tareas más robustas con relación al proceso de datos, un servidor es una aplicación en ejecución capaz de atender las peticiones de un cliente, y devolverle una respuesta en concordancia, muchas empresas e instituciones utilizan computadoras de escritorio simulando servidores, existen en diversas formas, que a su vez podemos encontrarlos como servidores de bases de datos Mysql/SQL, servidores de internet IIS/Apache, servidores de dominio, servidores firewall, entre otros. En tal sentido, como estos cumplen un papel importante en sus fases de ejecución tenemos que resaltar que una enorme cantidad de dispositivos en la empresa o institución dependen de ellos, en la mayoría de los casos una misma computadora puede proveer múltiples servicios y tener varios servidores en funcionamiento, existen otros que el contar con uno o varios servidores dentro de una misma organización hace que la monitorización de los mismos sea un asunto importante.

Asimismo, se debe tomar en cuenta que los servidores en una organización son de vital importancia, ya que proveen servicios esenciales dentro de una red, con el objetivo de reforzar la productividad que en efecto mantiene al personal de la organización productivo y reduce los costos de tecnología con una fuente centralizada para el almacenamiento de archivos, acceso a impresoras, bases de datos y una conexión compartida a internet, un servidor ejecuta aplicaciones como software contable lo que permite el acceso simultaneo de personas que utilicen el software y tengan acceso al mismo archivo simultáneamente, de igual manera protege los sistemas y sus respectivos datos que en consecuencia mantiene copias de los datos de manera automática,

colabora y comparte los datos con mayor efectividad con aplicaciones de software de acceso remoto.

Esto permite reunir a los usuarios de manera virtual lo que garantiza tiempos ágiles de respuesta y facilidad de traslado y almacenamiento de información, en cuanto a espacio de almacenamiento permite manejar aplicaciones de software con mayor facilidad, ya que optimiza el acceso de los usuarios y la conectividad, maximiza el desempeño y reduce los costos de las licencias de software cuando hay aplicaciones con grandes requisitos de espacio almacenadas de un servidor en contra parte de un sistema de escritorio, en cuanto a seguridad los servidores permiten mantener las oficinas protegidas de intrusos o virus a través de excelentes herramientas antivirus que puedan examinar archivos recibidos antes de que lleguen a las estaciones de trabajo.

En consecuencia, cuando un servidor falla muchos procesos se detienen y esto es considerado un gran problema dentro de una institución, empresa u organización, por tanto es conveniente mencionar que aunque el 80% de los problemas que presentan los servidores son debido a software, el otro 20% restante no es menos importante. Es importante resaltar, que un servidor está basado en la misma arquitectura que el resto de las computadoras, aquí es donde nos damos cuenta que el problema es parte de un contexto en el que se conjugan otros problemas relativos en la red de servidores de la Universidad Valle del Momboy, pero primero vamos a definir como está estructurado la red de servidores en la Universidad, en la sede Mirabel:

Se cuenta con una conexión inalámbrica de 8 mb, aquí es donde se ubica el centro de procesamiento de datos, que también se le puede llamar centro de cómputo donde podemos encontrar todas las redes de comunicaciones, en Mirabel existe una sala para mantener en él una gran cantidad de equipamiento informático y electrónico, con el fin de garantizar la continuidad

de los servicios en la Universidad se decide mudar los equipos a esta sede principal en el año 2015, que contaba con las condiciones mínimas para soportar estos equipos, No obstante en el año 2014 la Universidad registró un fallo que correspondía a un problema de vulnerabilidad de acceso, para ese entonces no se contaba con ningún instrumento para validar los usuarios que ingresaban al sistema, igualmente cualquier usuario mal intencionado podía borrar o robar datos del sistema, la seguridad en los servidores es tan importante como la seguridad de la red, si bien es cierto que un servidor usualmente contiene una gran cantidad de información vital de la organización, esto en consecuencia permite que un servidor se vea comprometido, ya que todos sus contenidos pueden estar disponibles para que un intruso los pueda manipular o robar a su gusto.

Para lograr posibles soluciones con respecto a este problema se debió crear una zona protegida para disminuir los ataques informáticos y a su vez se debía actualizar los servidores, sin embargo se puede enumerar muchos más problemas que han ocurrido con respecto a fallas eléctricas, que en consecuencia deja inútil la operatividad de equipos fundamentales para el procesamiento de datos, si bien se sabe que estos problemas pueden ser tratados a tiempo en la Universidad no se cuenta con personas que observen constantemente el funcionamiento del servidor así como han ocurrido otros casos de quemarse el servidor por falta de mantenimiento, es cierto que han ocurrido muchas fallas de energía eléctrica, causando que todos los servicios y páginas web de la universidad quedaran fuera de operatividad por un tiempo, se toma la decisión en el año 2015 de comprar una planta eléctrica que permitiera el funcionamiento de los servidores las 24 horas del día, 7 días a la semana.

En un sistema automatizado se debe discriminar la información recopilada de estos, para que solamente se dé un aviso de los eventos que necesiten atención, la omisión de una propuesta

de monitorización conlleva a la conjugación de problemas secuenciales, aquí es donde se plantea la siguiente interrogante: ¿Qué elementos son necesarios para el desarrollo de un sistema experto de diagnóstico y fallas de la red de servidores que se encuentran en la Universidad Valle Del Momboy, Valera, estado Trujillo?

Objetivos de la Investigación:

Objetivo General

- Desarrollar un sistema experto de diagnóstico y fallas de la red de servidores que se encuentran en la Universidad Valle Del Momboy, Valera, estado Trujillo.

Objetivos Específicos

- Identificar las fallas procedentes de la red de servidores de la Universidad Valle del Momboy, Valera, estado Trujillo, mediante encuestas dirigidas a los estudiantes y personal administrativo.
- Generar una propuesta que permita una eficiente toma de decisiones para actuar de manera anticipada a los problemas.
- Analizar el algoritmo y la metodología correcta a ser planteada.
- Desarrollar el software de inteligencia artificial acorde a la metodología y los parámetros elegidos.

Justificación

A lo largo de la investigación se han tomado aspectos importantes en cuanto al origen y el desarrollo del problema, entre estas podemos definir las etapas más importantes que nos llevan a sus cambios sustanciales, sobre la base de las ideas expuestas podemos definir los aspectos teóricos que motivaron el interés de la investigación el cual tiene por objeto de estudio los servidores y su desempeño en una organización, dentro de la Universidad Valle Del Momboy existen numerosos dispositivos y equipos que requieren de un servicio continuo, este proceso engloba lo que es el sistema académico, campus virtual, pagina web de la universidad y todo esto alojado a la red de servidores correspondiente.

En tal sentido, si se detiene el procesamiento de datos dentro de la red de servidores o es interrumpido por algún factor externo, afectaría lo que es el cronograma académico en la Universidad, aunado a la situación es fundamental resaltar que es la misma Universidad quien se va a beneficiar del desarrollo de un sistema experto de diagnóstico y fallas, de esta manera se puede tener un continuo desempeño óptimo en los servicios con la obtención de una propuesta que sea favorable y capaz de optimizar los procesos académicos, para tal efecto, una aportación resaltante de este trabajo es la correcta interpretación de los datos, que se logra obtener con el proceso de monitorización, ayudando así a los administradores de los servicios a tomar decisiones oportunas agilizando de esta manera la solución a los problemas que se presenten, tomando en cuenta que la justificación tiene una orientación institucional se debe enfatizar la importancia del estudio en cuanto a la aportación de información valiosa y relevante para la toma de decisiones en beneficio a la organización.

Con esto podemos decir que no solo se puede actuar de forma reactiva, el poder analizar oportunamente el estado de los equipos con los que se cuenta, los servicios que se tienen y el estado de la red nos permite actuar de forma preventiva al percatarse de algún posible fallo en el sistema, evitando así de manera considerable la interrupción de los servicios ya sean de red o de los servidores. El esquema de monitorización que se va a presentar en este trabajo tiene como finalidad informar del correcto comportamiento de la red los servidores y servicios de la Universidad Valle Del Momboy.

Delimitación

Geográfica: Con respecto a nuestra delimitación espacial podemos definir que el lugar para desarrollar el proyecto y los distintos espacios a tomar en cuenta para considerar su respectiva solución será la Universidad Valle el Momboy, ubicada en la ciudad de Valera Estado Trujillo. Llevando a cabo un profundo análisis de los servidores y el esquema actual de sus procesos, la manera en que podamos poner en marcha el proyecto se basa en los conocimientos adquiridos de la misma Universidad, analizando los procesos de los servidores y los problemas que se presentan tomando en cuenta que son el principal objeto de estudio para posteriormente realizar un diseño conceptual y práctico de un sistema experto que diagnostique estas fallas.

Temática: La investigación está adscrita a la Línea de Investigación: Modelos matemáticos, lógica difusa y cibernética. Partiendo también de algoritmos genéticos que ayudan a interpretar de manera eficiente los problemas y sus paso a paso dentro de una organización.

Temporal: Este proyecto de investigación se realizará en un periodo de tiempo estimado desde septiembre 2018 hasta enero de 2019.

CAPÍTULO II

MARCO TEÓRICO

Antecedentes del Sistema Experto

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 computadoras 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 ProblemSolver). Este sistema permitía resolver problemas como el de la torre de hanoi y otros similares, a través de la criptoaritmética, a pesar a ello, este programa no permitía resolver problemas más “cotidianos” y reales como por ejemplo dar un diagnóstico de alguna falla en la red de servidores de una organización. Es por ello que algunos investigadores cambiaron el enfoque del problema, a partir de ese momento se dedicaban a resolver problemas sobre un área específica intentando simular el razonamiento humano.

Ahora con este avance en vez de dedicarse a computarizar la inteligencia general, se centraron en dominios de conocimiento muy concretos, de esta manera nacieron los sistemas expertos. El primer SE que se aplicó a problemas más reales fue desarrollado en 1965 con el fin de identificar estructuras químicas: el programa se llamó DENDRAL. Lo que este sistema experto hacía, al igual que lo hacían los expertos de entonces, era tomar unas hipótesis relevantes como soluciones posibles, y someterlas a prueba comparándolas con los datos. El nombre DENDRAL significa árbol en griego. Debido a esto, el programa fue bautizado así porque su principal tarea era buscar en un árbol de posibilidades la estructura del compuesto. El próximo

sistema experto que causó gran impacto fue el Mycin en 1972, pues su aplicación era detectar trastornos en la sangre y recetar los medicamentos requeridos. Tuvo un rotundo éxito, eso produjo que se llegara a utilizar en algunos hospitales.

Tomando como referencia el proyecto de grado “Desarrollo de un prototipo de sistema experto para el apoyo en la toma de decisiones del proceso de selección de personal” (Pérez, E 2011) tiene como objetivo el desarrollo de un prototipo de sistema experto que contenga el conjunto de procesos de razonamiento, y conocimientos requeridos por un experto en selección de personal. Con el fin de orientar el desarrollo de los elementos de software se proporciona como bosquejo o técnica el modelado UML (Unified Modeling Language) dando como resultado un software propuesto y orientado a apoyar la toma de decisiones en el proceso de selección de personal en una organización.

Por otra parte el proyecto de tesis mencionado anteriormente, guarda aspectos similares al sistema experto de diagnóstico y fallas de la red de servidores de la Universidad Valle Del Momboy, sobre todo por el planteamiento de sistema experto capaz de agilizar y emular el proceso que una persona en una organización llevaría a cabo. Es un aporte significativo para comparar y analizar el razonamiento del sistema, mediante un conjunto de operaciones matemáticas.

En el trabajo especial de grado “Monitorización de servicios de red y servidores” (Delgadillo J, García L, 2010) tiene como objetivo crear un sistema para la monitorización del desempeño de la red y los servidores utilizados en la Unidad de servicios de Cómputo Académico de la Facultad de Ingeniería de la UNAM. En representación de la técnica utilizada para el esquema de monitorización pretende informar del correcto comportamiento de la red, los

servidores y servicios de una organización, describiendo los modelos de referencia OSI (“Open System Interconnection”) y TCP/IP (“Transmission Control Protocol/ Internet Protocol”) siendo estos los más resaltantes en el cumplimiento de la tarea de monitorización de red. Como resultado se obtiene la correcta interpretación de los datos que se logran obtener con dicho proceso de monitorización, a través de un sistema que permite contabilizar y llevar un registro de los datos relevantes de servidores y switches.

En efecto existe un gran aporte con respecto a este proyecto siendo la principal característica de este una considerable fuente de información para los modelos sistemáticos de monitorización para un diagnóstico de fallas en la red de servidores, apoyándose de los fundamentos básicos de los sistemas expertos.

“Algoritmos evolutivos y algoritmos genéticos” (Mateos, A) los algoritmos genéticos constituyen una técnica poderosa de búsqueda y optimización con un comportamiento altamente paralelo, inspirado en el principio darwiniano de selección natural y reproducción genética. Se basa de los modelos de computación bio-inspirados que permite emplear analogías con sistemas naturales o sociales para diseñar métodos heurísticos no determinísticos de búsqueda de aprendizaje, de imitación de comportamiento, entre otros. Existen distintos modelos de computación bio-inspirados como pueden ser: Redes neuronales, algoritmos inmunológicos, algoritmos basados en inteligencia de enjambres (swarm intelligence) y dentro de ellos, algoritmos basados en colonias de hormigas.

Es un trabajo fundamental para el desarrollo del trabajo especial de grado, basado en algoritmos genéticos, redes neuronales y lógica difusa. En tal sentido, la búsqueda de problemas menores como por ejemplo si se detiene el procesamiento de datos dentro de la red, para ir en

búsqueda de problemas más grandes como la interrupción de un cronograma de datos específico permite modelar en base a los algoritmos genéticos propuestas pertinentes y acertadas de un sistema experto.

Para 1973, se creó Tiersias, cuya función era la de servir de intérprete o interfaz entre los especialistas que manejaban el Mycin cuando introducían nuevos conocimientos. Tiersias entraba en acción cuando Mycin cometía un error en un diagnóstico, por la falta de información o por alguna falla en el árbol de desarrollo de teorías, corrigiendo la regla que generaba el resultado o bien destruyéndola. Para 1980 se instauró en Digital Equipment Corporation (DEC) el primer Sistema Experto: el XCON. Para esto se tuvieron que dedicar dos años al desarrollo de este SE. Y valió la pena para la compañía, pues aun cuando en su primer intento al implantarse en 1979 consiguió solo el 20% del 95% de la resolución de las configuraciones de todas las computadoras que salieron de DEC, volvió al laboratorio de desarrollo otro año más, y a su regreso le resultó en un ahorro de 40 millones de dólares a dicha compañía.

Para los años comprendidos entre 1980 y 1985 se crearon diversos sistemas expertos, tales como el DELTA de la General Electric Company, el cual se encargaba de reparar locomotoras diésel y eléctricas, o como “Aldo en Disco”, que reparaba calderas hidrostáticas giratorias usadas para la eliminación de bacterias. En ese momento surgen empresas dedicadas a desarrollar sistemas expertos, las cuales estimaron en conjunto una inversión total de más de 300 millones de dólares. Los productos más importantes que creaban estas nuevas compañías eran las “maquinas LISP”, las cuales consistían en unas computadoras que ejecutaban programas LISP con la misma rapidez que en una computadora central. Desgraciadamente para DEC, el SE que antes fuera un gran ahorro dejó de ser rentable en 1987, debido a que al actualizarlo llegaban a gastar más de 2 millones de dólares al año, para darle mantenimiento por lo que su beneficio quedó en duda y por

lo tanto quedó discontinuado; algo similar ocurrió con DELTA. (Samper Márquez, Juan José. **Sistemas Expertos, el conocimiento al poder**) (Roos, Rita. **Historia de los sistemas expertos**).

Bases Teóricas

Conceptos básicos para el conocimiento de un sistema experto y sus atributos

Inteligencia artificial: Es la ciencia e ingenio de hacer maquinas inteligentes, especialmente programas de cómputo inteligentes (McCarthy J, 1956)

Sistema experto: En inteligencia artificial un sistema experto (SE) es un sistema computacional que emula la capacidad de tomar decisiones de un humano experto.

Base de conocimientos: Es la parte del sistema experto que contiene el conocimiento sobre el dominio. Hay que obtener el conocimiento del experto y codificarlo en la base de conocimientos. Una forma clásica de representar el conocimiento en un sistema experto son las reglas. Una regla es una estructura condicional que relaciona lógicamente la información contenida en la parte del antecedente con otra información contenida en la parte del consecuente (Samper J, 2003)

Base de hechos (Memoria de trabajo): Contiene los hechos sobre un problema que se han descubierto durante una consulta. Durante una consulta con el sistema experto, el usuario introduce la información del problema actual en la base de hechos. El sistema empareja esta información con el conocimiento disponible en la base de conocimientos para deducir nuevos hechos.

Motor de inferencia: El sistema experto modela el proceso de razonamiento humano con un módulo conocido como el motor de inferencia. Dicho motor de inferencia trabaja con la información contenida en la base de conocimientos y la base de hechos para deducir nuevos

hechos. Contrasta los hechos particulares de la base de hechos con el conocimiento contenido en la base de conocimientos para obtener conclusiones acerca del problema (Marcel C, 2002).

Subsistema de explicación: Una característica de los sistemas expertos es su habilidad para explicar su razonamiento. Usando el módulo del subsistema de explicación, un sistema experto puede proporcionar una explicación al usuario de por qué está haciendo una pregunta y como ha llegado a una conclusión. Este módulo proporciona beneficios tanto al diseñador del sistema como al usuario. El diseñador puede usarlo para detectar errores y el usuario se beneficia de la transparencia del sistema.

Interfaz de usuario: La interacción entre un sistema experto y un usuario se realiza en lenguaje natural. También es altamente interactiva y sigue el patrón de la conversación entre seres humanos. Para conducir este proceso de manera aceptable para el usuario es especialmente importante el diseño del interfaz de usuario. Un requerimiento básico del interfaz es la habilidad de hacer preguntas. Para obtener información fiable del usuario hay que poner especial cuidado en el diseño de las cuestiones. Esto puede requerir diseñar el interfaz usando menús o gráficos.

Principales aplicaciones de los sistemas expertos: Según Félix Justo (2004), las principales aplicaciones de los sistemas expertos se dan en las gestiones empresariales debido a que casi todas las empresas disponen de un ordenador que realiza las funciones básicas de tratamiento de la información: contabilidad general, decisiones financieras, gestión de la tesorería, planificación, entre otros. Este trabajo implica manejar grandes volúmenes de información y realizar operaciones numéricas para después tomar decisiones, esto crea un terreno ideal para la implantación de los sistemas expertos anexo a esto se aplican en la contabilidad en apartados como: Auditoria (que es el campo en el que más aplicaciones de sistemas expertos se está

realizando, fiscalidad, planificación, análisis financiero y la contabilidad financiera, los sistemas expertos se aplican en una gran diversidad de campos y/o áreas, pero las principales podrían enlistarse como el campo militar, informática, telecomunicaciones, química, derecho, aeronáutica, geología, arqueología, agricultura, electrónica, transporte, educación, medicina, industria, finanzas y gestión.

Ventajas de los sistemas expertos: Criado José, (2002) establece las ventajas de los sistemas expertos como la proporción y la capacidad de trabajar con grandes cantidades de información, que son uno de los grandes problemas que enfrenta el analista humano que puede afectar negativamente a la toma de decisiones pues el analista humano puede depurar datos que no considere relevantes mientras un sistema experto debido a su gran velocidad de proceso analiza toda la información incluyendo las no útiles para de esta manera aportar una decisión más sólida.

Limitaciones de los sistemas expertos: tomando en cuenta la reprogramación de los SE como una de las limitaciones más acentuadas, otro factor fundamental dentro de las limitaciones es el elevado costo en tiempo y dinero, además que estos programas son poco flexibles a cambios y de difícil acceso a información no estructurada. Debido a la escasez de expertos humanos en determinadas áreas los SE pueden almacenar su conocimiento para cuando sea necesario poder aplicarlo. Así mismo los SE pueden ser utilizados por personas no especializadas para resolver problemas: si una persona utiliza con frecuencia un SE aprenderá de él. Por otra parte la inteligencia artificial no ha podido desarrollar sistemas que sean capaces de resolver problemas de manera general, de aplicar el sentido común para resolver situaciones complejas ni de controlar situaciones ambiguas.

Monitorización: Es un caso particular de la interpretación que trata sobre la comparación continua de los valores, señales o datos de entrada y unos valores que actúan como criterios de normalidad o estándares. En el campo del mantenimiento predictivo los sistemas expertos se utilizan fundamentalmente como herramientas de diagnóstico, se trata de que el programa pueda determinar en cada momento el estado de funcionamiento de sistemas complejos, anticipándose a los posibles incidentes que pudieran acontecer. En este sentido, usando un modelo computacional del razonamiento de un experto humano, proporciona los mismos resultados que alcanzaría dicho experto.

Diseño de un sistema experto: Es el proceso en el que se especifica una descripción de un artefacto que satisface varias características desde un número de fuentes de conocimiento, sin embargo el diseño puede concebirse de varias formas entre ellas se define el diseño en ingeniería como el uso de principios científicos, información técnica e imaginación en la definición de una estructura mecánica, máquina o sistema que ejecute funciones específicas con el máximo de economía y eficiencia, por otro lado en el diseño industrial se busca rectificar las omisiones de la ingeniería como un intento consciente de traer forma y orden visual a la ingeniería de hardware donde la tecnología no provee estas características. Podemos decir que los sistemas expertos en diseño ven este proceso como un problema de búsqueda de una solución óptima o adecuada, las soluciones alternas pueden ser conocidas de antemano o se pueden generar automáticamente probándose distintos diseños para verificar cuáles de ellos cumplen los requerimientos solicitados por el usuario, puede darse el caso que la prueba se termina cuando se encuentra la primera solución pero existen casos donde el principal objetivo es encontrar la solución óptima.

Tipos de sistemas expertos: Según Criado Briz, José Mario, (2002). Existen diferentes puntos de vista de los cuales se podrá clasificar a los sistemas expertos, entre ellos se encuentran los siguientes:

- **Por la forma de almacenar conocimiento:** Se podrían diferenciar sistemas basados en reglas y sistemas basados en probabilidad, así en el primer caso el conocimiento se almacena en forma de hechos y reglas, mientras que el segundo, la base de conocimientos está constituida por hechos y sus dependencias probabilísticas.
- **Por la naturaleza de la tarea a realizar:** En este modo se tiene cuatro probabilidades de las cuales se menciona el diagnóstico o clasificación tomando como ejemplo un sistema de diagnóstico médico, luego tenemos la monitorización que trata el análisis del comportamiento de un sistema buscando posibles fallos, el diseño que busca la construcción de la solución a un problema que en principio es desconocida y por último tenemos la predicción que estudia el comportamiento de un sistema.
- **Por la interacción del usuario:** Este modo interpone el apoyo del sistema que aconseja al usuario y mantiene la capacidad de una última decisión tomando como ejemplo el diagnóstico médico, el sistema actúa directamente sin intervención humana mediante el control, que a su vez su misión es analizar y criticar decisiones tomadas por el usuario.
- **Por la certeza de la información:** Cuando se conoce perfecta o completa se tienen todos los datos y reglas necesarias para la decisión, pero se puede dar el caso en que sea imperfecta o incompleta (falta de información para tomar decisiones) datos inciertos (o no confirmados), conocimiento incierto (reglas no siempre válidas), terminología ambigua (dobles sentidos, entre otros).

Seguridad: Es el estado de bienestar de la información y las estructuras en las cuales la posibilidad que puedan realizarse con éxito y sin detenerse, el robo, alteración y parada del flujo de información se mantienen en niveles bajos o tolerantes. “garantizar que los recursos informáticos de una compañía estén disponibles para cumplir sus propósitos, es decir, que no estén dañados o alterados por circunstancias o factores externos, es una definición útil para conocer a lo que implica el concepto de seguridad informática” (Internet, <http://www.formulaenlosnegocios.com.mx/seguridad-informatica-%C2%BFque-por-que-y-para-que/>).

Seguridad informática: Es un área de la informática que se enfoca en la protección de las estructuras computacionales, que nos permite tener confianza de que la información que estemos utilizando se encuentre en un lugar seguro y confiable.

Gestión de seguridad en redes: “La gestión se puede definir como el conjunto de actividades que controlan o vigilan el uso de los recursos. Se debe proporcionar la posibilidad de supervisar el estado, medir el rendimiento, reconocer actividades anormales y recuperar el servicio” (Torres Andagana, G. C 2010). Las funciones de red suelen agruparse en dos categorías:

- **Supervisión de red:** La supervisión de red se puede considerar como una función de lectura y se encarga de observar y analizar el estado y el comportamiento de la configuración y los componentes de la red.
- **Control de red:** Se le puede considerar como una función de escritura y se encarga de alterar los parámetros de los distintos componentes de la configuración de la red y hacer que lleven a cabo acciones que se determinen.

Vulnerabilidad de la información en los servidores: Según Rodrigo Ferrer, (2011), una vulnerabilidad se puede considerar como un estado de un sistema o conjunto de sistemas que puede permitir a un atacante acceder a información confidencial, modificar información o permitir a un atacante negar un servicio.

Conceptos básicos para la implementación de un sistema experto para el diagnóstico de falla en redes de servidores

Red de datos: Se conoce como red de datos a la infraestructura cuyo diseño posibilita la transmisión de información a través del intercambio de datos. Cada una de estas redes ha sido diseñada específicamente para satisfacer sus objetivos, posee una arquitectura determinada para facilitar el intercambio de los contenidos. No podemos pasar por alto que una red de datos se pone también en funcionamiento con otros dos objetivos primordiales: Compartir tanto el software como el hardware y otorgarle soporte y centralización a la administración pertinente. (Pérez J, Merino M, 2014)

Elementos fundamentales en redes de datos (Pérez J, Merino M, 2014):

Pach Panel's: Son los sistemas que se encargan de organizar todo el cableado necesario.

Hubs: Proceden a ejecutar lo que sería la amplificación de las señales que toman protagonismo en ese intercambio de información. Los cables conocidos como Pach Cord o el conocido como cableado de tipo horizontal son otras de las propuestas que también cobran protagonismo en una red de datos.

Tipos de redes de datos (Pérez J, Merino M, 2014): De acuerdo a su alcance una red de datos puede ser considerada como una red de área personal (Personal Area Network o PAN), red de área local (LAN), red de área metropolitana (MAN) o una red de área amplia (WAN):

- **Red PAN:** Es aquella red que interconecta computadoras situadas cerca de una persona.
- **Red LAN:** Favorece el intercambio de datos en una zona pequeña (como una oficina o un edificio).
- **Red MAN:** Brinda una cobertura en un área geográfica extensa.
- **Red WAN:** Permite compartir datos en una superficie de gran extensión.

Servidor: Es una computadora que forma parte de una red y que provee servicios a distintos ordenadores, dichos ordenadores se definen como **clientes**. Los servidores suelen utilizarse para almacenar archivos digitales, en este sentido, los clientes se conectan a través de la red con el servidor y acceden a dicha información. En ocasiones, un ordenador puede cumplir con las funciones de servidor y de cliente de manera simultánea. (Pérez J, Gardey A, 2010)

Tipos de servidores (Pérez J, Gardey A, 2010): Existe una variedad de tipos de servidores, cada uno con funciones dedicadas, y distribución de almacenamiento a los clientes, dichos servidores son:

- **Servidores de correo:** Guardan, reciben y envían correos electrónicos.
- **Servidores web:** Almacenan los documentos que son accesibles a través de internet.
- **Servidores de archivos:** Almacenan los documentos y los distribuyen a los clientes de la red.

Clasificación de los servidores (Pérez J, Gardey A, 2010): Se pueden distinguir entre dos grandes grupos, los dedicados enteramente al servicio de los clientes de la red; y los no dedicados, que pueden a su vez funcionar como estaciones de trabajo local, podemos hablar de servidores tales como:

- **De impresiones:** Controla una o varias impresoras, con la posibilidad de recibir trabajos de varios clientes de la red. Cuando hay un mínimo de dos documentos pendientes, éstos son organizados en una cola, para finalmente ser atendidos uno a uno; por otro lado, es posible modificar la prioridad de cada uno con el propósito de alterar el orden.
- **De fax:** Similar al servidor de correo, pero relacionado con este sistema de envío de información, más antiguo y considerablemente más limitado.
- **De telefonía:** Ofrece una serie de funciones propias de las comunicaciones telefónicas, tales como la contestación automática, a través de sistemas interactivos con múltiples opciones, incluso con reconocimiento de voz, que son capaces de resolver las necesidades del cliente o de derivarlos a un operador que pueda asistirlos.
- **Proxy:** Se encarga de realizar algunas funciones que maximizan el rendimiento de operaciones que tengan lugar muy a menudo, además de ofrecer servicios de protección de datos, tales como Firewall (pared de fuego). A su vez administra los permisos de acceso que tienen los ordenadores de una red a ciertos sitios web.
- **De acceso remoto:** Lleva a cabo el control de las líneas que comunican un equipo a otro para acceder a él de forma remota, respondiendo a los llamados telefónicos entrantes, reconociendo automáticamente las peticiones de conexión, y realizando las

verificaciones de identidad pertinentes para proceder, finalmente, al registro de nuevos usuarios en la red.

- **De uso:** Resuelve la porción lógica de las tareas que los usuarios realizan en otros nodos, quienes le envían una serie de instrucciones que, una vez ejecutadas, les son devueltas de forma silenciosa, como si el procesamiento hubiera tenido lugar en forma local. Podemos ver este tipo de sistemas de trabajo comúnmente en bancos, clínicas privadas y otras instituciones que manejan grandes cantidades de información y que operan de cara al público.

Cliente servidor: Refiere a un modelo de comunicación que vincula a varios dispositivos informáticos a través de una red. El cliente, en este marco, realiza peticiones de servicios al servidor, que se encarga de satisfacer dichos requerimientos. Con esta arquitectura las tareas se distribuyen entre los servidores (que proveen los servicios) y los clientes (que demandan dichos servicios). Dicho de otro modo: el cliente le pide un recurso al servidor, que brinda una respuesta. Entre las disposiciones más comunes del modelo cliente servidor se encuentran los sistemas multicapa, según los cuales el servidor ofrece la ejecución de varios programas para que varios ordenadores puedan solicitarlos según sus necesidades, de manera que el nivel de distribución aumenta (Pérez J, Gardey A, 2016).

Conceptos básicos para conocer estabilidad en el sistema y alternativas viables

Desempeño óptimo de un sistema: Ciertamente, para que un sistema se pueda considerar óptimo, debe cumplir con ciertas especificaciones del usuario. Si tenemos el caso de que el sistema sea un servidor, se puede decir que el sistema es óptimo cuando el mismo está organizado correctamente o bien administrado, este es seguro y con respuestas rápidas a los clientes. Por esta

razón se tienen que medir de alguna forma, parámetros clave del sistema, para que no alcancen niveles que generen un mal comportamiento.

Funcionamiento óptimo para servidores Windows (<https://support.microsoft.com/es-ve>):

Entre los parámetros fundamentales que se deben tener en cuenta para que los servidores Windows estén en un estado óptimo y funcional podemos encontrar:

- Hardware de 64 bits, memoria suficiente y tarjetas de red rápidas.
- Usar el procesador de un servidor manteniendo una carga del 60 por ciento aproximadamente durante las horas de máxima actividad. Este porcentaje admite períodos de carga muy elevada. Si el uso del procesador está por encima del 75 por ciento de manera continuada, el rendimiento del procesador se considera un cuello de botella.
- Usar tarjetas de red de gigabit para todas las funciones de servidor.
- Para los servidores cliente-web y los servidores de aplicaciones, utilizar tarjetas NIC duales en los entornos de producción: una para los usuarios y otra para la comunicación de SQL Server.
- Otorgar al menos 2 GB de RAM por cada procesador para los servidores cliente-web y los servidores de aplicaciones.
- Otorgar al menos 4 GB de RAM por cada procesador para los servidores de base de datos.
- No se debe sobrecargar con demasiada información los servidores de base de datos.
- Se debe limitar las bases de datos de contenido a 100 GB.
- Memoria usada: menos del 70 por ciento.

- No almacenar más de 50 bases de datos en una única instancia física de SQL Server.
- Espacio libre en disco: más del 25 por ciento.

Cloud Computing: La computación en nube, a menudo llamada “la nube”, es la entrega de servicios de computación bajo demanda (desde aplicaciones a centros de datos) a través de internet en una base de pago por uso. (Fuente: <https://www.ibm.com/cloud-computing/mx-es/learn-more/what-is-cloud-computing/>)

Cloud Computing como alternativa: Esta opción permite gestionar y almacenar los archivos de una empresa, institución u organismo, es una alternativa que ofrece una serie de beneficios. Para quien está empezando esta tecnología no requiere una gran inversión y, teóricamente, será necesario un equipo de TI menor para gestionar el servidor, esto es una opción que encaja con las necesidades y disponibilidades de pequeños y medianos negocios. Es fundamental tomar en cuenta que el proveedor elegido por la empresa sea estable y confiable, los fundamentos de dicho proveedor deben ser sólidos pues una crisis o incluso el quiebre causarán serios problemas a la empresa (Fuente: <https://blogmexico.comstor.com/como-elegir-un-servidor-para-la-empresa>).

Virtualización: Es la creación a través de software de una versión virtual de algún recurso tecnológico, como puede ser una plataforma de hardware, un sistema operativo, entre otros. (González J, Aragón D, 2014)

Virtualización como alternativa: De acuerdo a un relevante trabajo de investigación proporcionado desde el portal: <https://dialnet.unirioja.es/servlet/articulo?codigo=5157975> se da como resultado que las tecnologías de virtualización han brindado importantes beneficios como reducción de costos, automatización de sistemas, escalabilidad, ahorro de espacio físico,

reducción de consumo energético, entre otros; los servidores se crean con el objetivo de procesar cargas de uso que varían de niveles promedio a muy altos, por ello la mayor parte del tiempo están subutilizados. Al operar diversas máquinas virtuales en un solo servidor físico, una empresa puede utilizar estos recursos de forma más eficiente, pues elevará entre un 60 por ciento y un 80 por ciento el nivel de utilización de los servidores.

Monitorización general

¿Qué es?

Por definición, la monitorización es la acción y efecto de monitorizar, que según el Diccionario de la Lengua Española en su vigésima segunda edición, significa observar, mediante aparatos especiales, el curso de uno o varios parámetros fisiológicos o de otra naturaleza, para detectar posibles anomalías. Por ende, la monitorización de red, es la acción que nos permite verificar sistemáticamente el desempeño y la disponibilidad de los dispositivos críticos dentro de la red, a través de la identificación y detección de posibles problemas.

Tipos de monitorización

Hay diversas formas de monitorización. Las dos más comunes son la monitorización pasiva y la activa. Se puede visualizar ventajas y desventajas en ambas:

- **Monitorización activa:** Tiene la capacidad de inyectar paquetes de prueba dentro de la red o enviar paquetes a servidores con determinadas aplicaciones, siguiéndolos para medir los tiempos de respuesta. Por lo que, genera tráfico extra lo suficiente para recabar datos precisos. Este tipo de monitorización permite el control explícito en la generación de paquetes para realizar las mediciones, como el control en la naturaleza

de generar tráfico, las técnicas de muestreo, frecuencia, tamaño de los paquetes entre otras.

- **Monitorización pasiva:** Solo está al tanto de lo que pasa sin modificar ningún parámetro u objeto. Usa dispositivos para ver el tráfico que está circulando a través de la red, estos dispositivos pueden ser sniffers o en su defecto sistemas incluidos en los switches y ruteadores. Una de las características de la monitorización pasiva es que no incrementa el tráfico en la red para poder realizar las lecturas. Se puede capturar el tráfico teniendo un puerto espejo (mirror) en un dispositivo de red o un dispositivo intermedio que esté capturando el tráfico.
- **Alarmas:** Es una señal de cualquier tipo, que advierte de la proximidad de un peligro. Con consideradas como eventos fuera de lo común y por lo tanto, necesitan atención inmediata para mitigar la posible falla detectada. Las alarmas son activadas cuando un parámetro ha alcanzado cierto nivel o se tiene un comportamiento fuera de lo normal.

Objetivos que se deben monitorizar

Hay diversos individuos que se refieren a la salud de la red cuando hablan del rendimiento de la red y su capacidad para dar servicio. Existen funciones críticas dentro de los dispositivos de red que necesitan ser monitorizados constantemente y las alarmas que se lleguen a presentar deben ser atendidas tan pronto como sea posible cuando un evento ocurra. Algunos de los parámetros más comunes son la utilización de ancho de banda, el consumo de CPU, consumo de memoria, el estado físico de las conexiones, el tipo de tráfico que manejan los ruteadores, switches, hubs, firewalls y otros servicios de web, correo, base de datos, entre otros. También se debe estar pendientes de las bitácoras de conexión al sistema y configuración de los sistemas, ya que aquí se presenta información valiosa cuando un evento sucede. Estos parámetros deben ser

monitorizados muy de cerca para detectar posibles cambios y tendencias que afecten al usuario final.

Ancho de banda de una red Ethernet

Se puede encontrar en variadas bibliografías que la tecnología Ethernet se satura al llegar al 37% de su utilización. Sin embargo, esto no es del todo cierto. De acuerdo con Charles Spurgeon, en su libro *Ethernet The Definitive Guide*, el 37% fue reportado por primera vez por Bob Metcalfe y David Boggs en 1976. En un documento escrito por ellos, describen el desarrollo y operación de la primera Ethernet, la cual operaba a 3 Mbps. En su modelo asumían una transmisión constante de 250 computadoras. El sistema alcanzó una saturación de cerca del 36.8 por ciento en la utilización del canal. Los autores advirtieron que éste era un modelo simplificado y que no tenía ninguna relación con el funcionamiento normal de las redes. Este mito del desempeño de Ethernet persistió por muchos años, debido quizás a que nadie entendía que fue una prueba de la peor carga de tráfico de red que pudieran imaginar.

Las nuevas pruebas realizadas por estos mismos autores muestran lo contrario. Recientemente realizaron pruebas con 24 computadoras, mandando un flujo constante de datos a 10 Mbps en un canal Ethernet usando diferentes tamaños de frames. Para frames enviados entre pocas computadoras, la utilización del canal fue mayor a 9.5 Mbps y para frames más grandes resultó cercano al 10 Mbps, es decir casi del 100 por ciento. No se registra en ningún caso una saturación en un punto arbitrario de 37 por ciento. Los experimentos demostraron que la tecnología Ethernet puede transportar grandes cantidades de tráfico entre un conjunto de equipos manteniéndose estable y sin mayores problemas.

Estos autores recomiendan no tener muchas computadoras en un mismo dominio de colisión, es preferible usar switches y routers para segmentar la red en múltiples dominios de colisión. También recomiendan evitar combinar el uso de aplicaciones en tiempo real con aplicaciones para la transferencia de archivos ya que provocan retardos en estas aplicaciones.

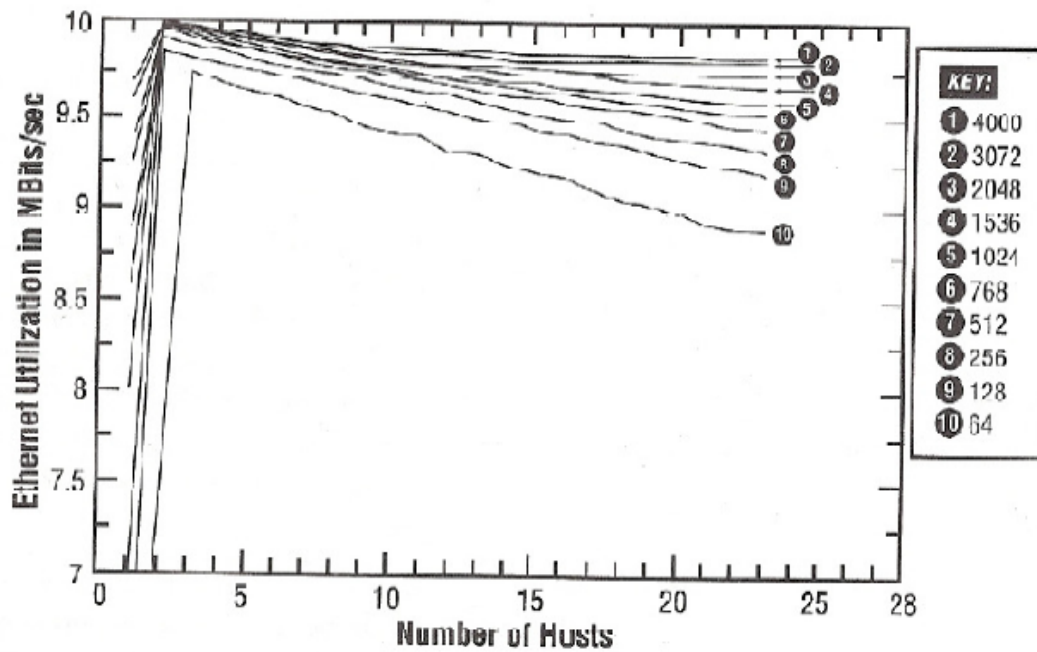


Figura 1: Experimento de utilización de Ethernet. Fuente: Spurgeon C. (Ethernet The Definitive Guide)

Metodologías del desarrollo de software

En el campo del desarrollo de software, existen dos grupos de metodologías, denominadas tradicionales (formales) y las ágiles, las primeras son un tanto riguroso, en ella se exige una documentación exhaustiva y se centra en cumplir con el plan del proyecto definido, totalmente en la fase inicial del desarrollo del mismo; mientras que la segunda enfatiza el esfuerzo en la capacidad de respuesta a los cambios, las habilidades del equipo y mantener una buena relación con el usuario. Algunos de los objetivos esenciales de las metodologías buscan cumplir con los siguientes objetivos:

- Asegurar la uniformidad y calidad tanto del desarrollo como del sistema en sí
- Satisfacer la uniformidad y calidad tanto del desarrollo como del sistema
- Ajustarse a los plazos y costes previstos en la planificación
- Conseguir un mayor nivel de rendimiento y eficiencia del personal asignado al desarrollo
- Proporcionar puntos de control y revisión
- Permitir construir un sistema documentado y que sea fácil de mantener

Componentes de la metodología

La metodología se define como la disciplina que indicará qué métodos y técnicas hay que usar en cada fase del ciclo de vida del desarrollo del proyecto, los elementos que componen la metodología son:

Fase

Son etapas del proceso de desarrollo de software, es un conjunto de actividades relacionadas con un objetivo en el desarrollo del proyecto. Se construye agrupando tareas que pueden compartir un tramo determinado del tiempo de vida de un proyecto, las fases que se observan en la metodología son: Análisis, Diseño, Desarrollo, Pruebas e Implementación.

Métodos

Son procesos que se descomponen en actividades más pequeñas, en estas actividades se definen los valores que recibirá cada fase así como los que generará y la técnica que se tendrá que usar.

Técnicas y herramientas

Indican como se deberá resolver cada actividad y que herramientas podríamos usar. Se encuentran diferentes tipos de técnicas, algunas de ellas son:

- De recopilación de datos: Formularios, entrevistas, entre otros.
- Técnicas de modelado: Desarrollo estructurado y orientado a objetos.
- Técnicas gráficas: Organigramas, Diagramas, entre otros.

Documentación

Para este proceso es necesario especificar que documentación se va a generar durante cada etapa del proceso; estos documentos deben realizarse de manera completa y usando todos los

valores de entrada y salida que se van generando, esto servirá para recoger los resultados y tomar decisiones de las diferentes situaciones planteadas.

Control y evaluación

Se deben de realizar a lo largo de todas las fases para identificar errores y corregirlos a tiempo. Consiste en realizar el seguimiento del avance de acuerdo al cronograma de trabajo; puede ser necesario tomar decisiones, como el replanteamiento de la planificación de las tareas asignadas para lograr los objetivos propuestos.

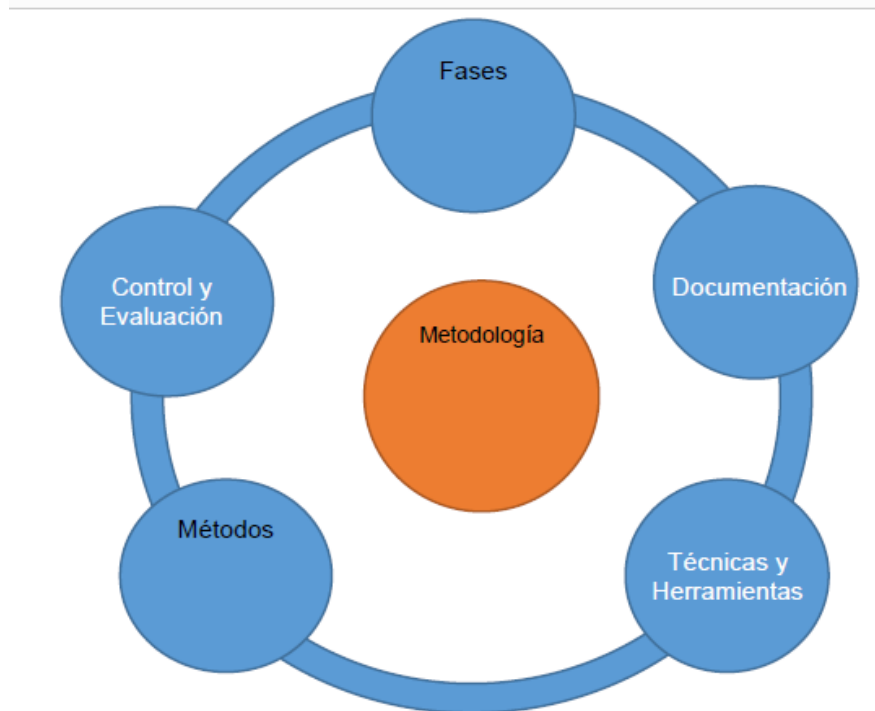


Figura 2: Elementos básicos de la metodología. Fuente: Velásquez (2017)

Ciclo de vida de la metodología

Es el conjunto de fases por las que un producto deberá pasar; se inicia con el análisis y termina en la fase de implementación, las distintas funciones que debe tener el ciclo de vida son

fundamentales, determina el orden de las fases del proceso de software, establece los criterios de transición para pasar de una fase a otra, se definen las entradas y salidas de cada fase, se describen las actividades a realizar para transformar el producto y se definen los esquemas que sirven como base para planificar, organizar y coordinar. El ciclo de vida se compone de fases sucesivas, compuestas por actividades que se pueden planificar, la sucesión de fases puede ampliarse con bucles de retroalimentación, de manera que, lo que conceptualmente se considera una misma fase se puede ejecutar más de una vez a lo largo del proyecto.



Figura 3: Ciclo de vida del desarrollo. Fuente: Velásquez (2017)

Modelo de ciclo de vida de software

El modelo de ciclo de vida de software es una vista de las actividades que se deben realizar durante el desarrollo de software, intenta determinar el orden de las etapas involucradas y los criterios de transición asociados entre estas etapas:

- Define las fases primarias esperadas de ser ejecutadas durante esas fases

- Describe las fases principales de desarrollo de software
- Provee un espacio de trabajo para la definición de un proceso detallado de desarrollo de software
- Ayuda a administrar el progreso del desarrollo

El ciclo de vida de la metodología se basa en el modelo incremental; la filosofía de este modelo consiste en incrementar las funcionalidades del software en varias iteraciones. Este modelo aplica secuencias lineales de forma escalonada mientras progresa el tiempo en el calendario. Cada secuencia lineal produce un incremento del software.

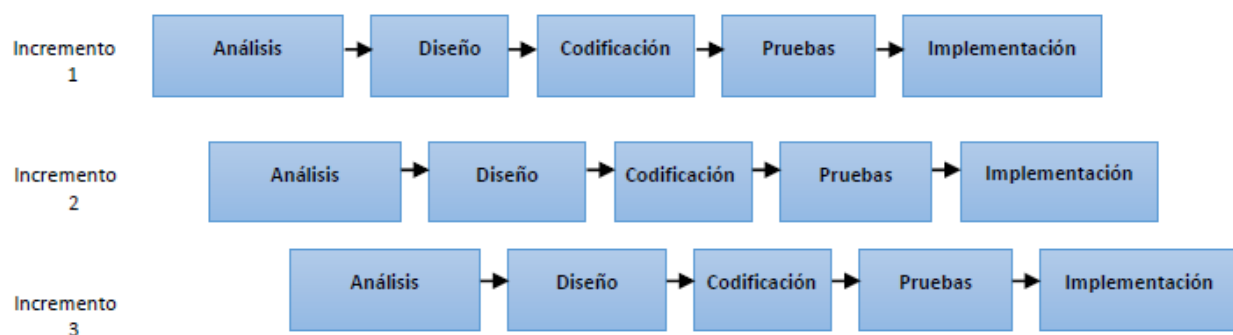


Figura 4: Modelo del proceso de desarrollo de software. Fuente: Velásquez (2017)

Análisis: Considerada como la más importante del ciclo de desarrollo (dependiendo del trabajo que se realice, se podrá comprender la naturaleza del problema correctamente) se determina que es lo que realmente se necesita hacer. Es una etapa crítica, por ello se requiere la participación de personas que cuenten con la experiencia solicitada; ya que de no realizarse un buen análisis puede traer consecuencias negativas para el proyecto, principalmente en el cumplimiento a tiempo. Para el siguiente diagrama se puede observar las actividades que se desarrollan en la etapa de análisis:

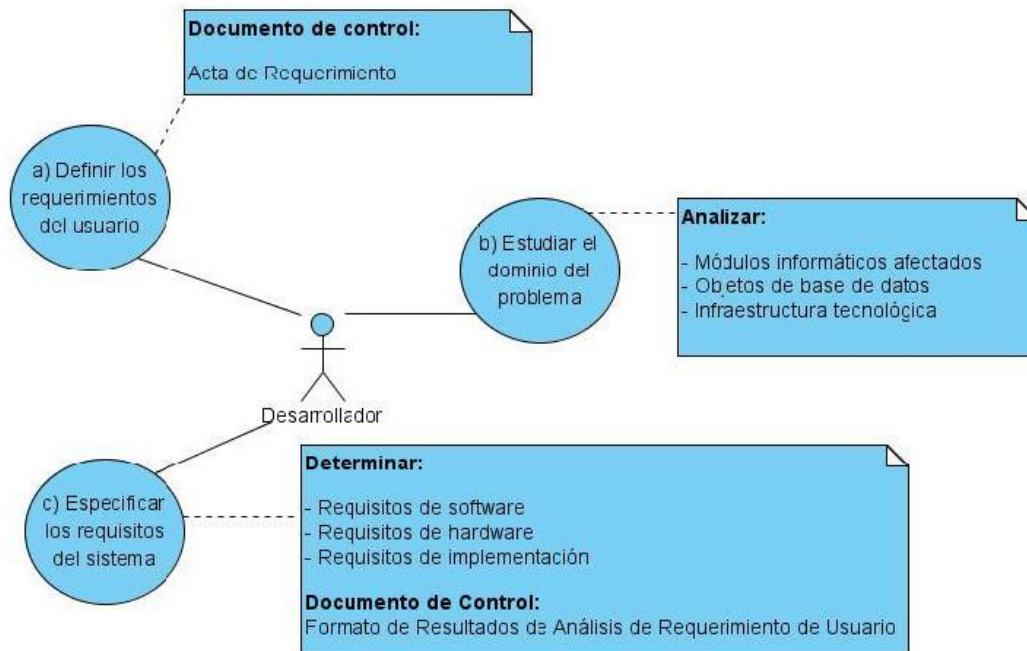


Figura 5: Actividades de la etapa de análisis. Fuente: Velásquez (2017)

Diseño: En esta fase, el desarrollador utiliza la información obtenida en el análisis y elabora el diseño lógico del producto. El diseño se enfoca en cuatro atributos: La estructura de los datos, la arquitectura del software, el detalle procedimental y la caracterización de la interfaz. Se procede a diseñar el modelo entidad-relación; que para ello se utiliza la herramienta de modelado y se diseña cada una de las entidades y atributos principales que van a necesitarse para almacenar los datos, se diseñan los prototipos de pantalla que ayudan a normalizar las funcionalidades del sistema y permiten construir interfaces uniformes para finalmente diseñar la estructura de la aplicación; donde se debe considerar la jerarquía de módulos establecidos para el producto que se va a desarrollar.

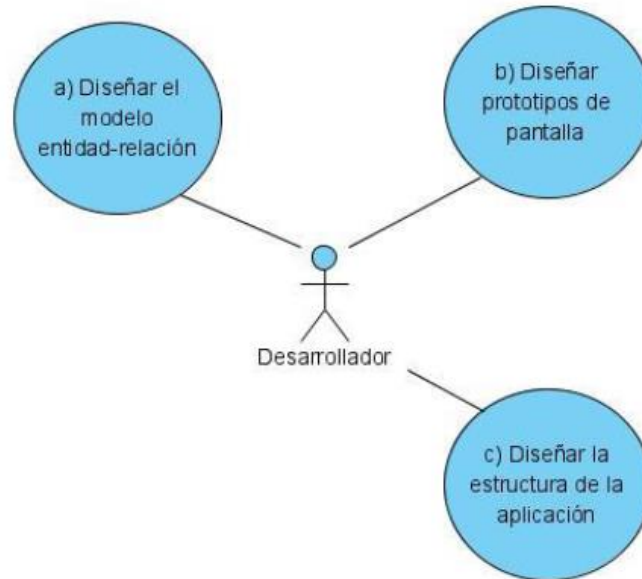


Figura 6: Actividades de la etapa de diseño. Fuente: Velásquez (2017)

Codificación: Una vez que se cuenta con los documentos de control de la fase de Análisis y del Diseño; se puede dar inicio a la fase de codificación. Para codificar se necesita conocer la sintaxis del lenguaje de programación que se vaya a emplear, en esta parte, el desarrollador deberá seguir los lineamientos impuestos en el diseño y tomando en consideración los requisitos funcionales y no funcionales. Tomando en cuenta que al escribir el código fuente el desarrollador deberá asegurarse que durante la escritura del código fuente esta siguiendo las normas y convenciones de codificación, luego se procede a realizar las pruebas unitarias que consiste en probar la funcionalidad de una parte del código, tales como rutinas, funciones, entre otros. Por ultimo las pruebas de funcionalidad, una vez que se tenga una versión terminada del producto, el desarrollador debe realizar pruebas para asegurarse que las entradas definidas producen los resultados esperados y que todos los componentes del producto funcionan correctamente.

Pruebas: Para cuando el producto se ha terminado de codificar; debe ser instalado en el ambiente de pruebas. En esta parte se busca comprobar que el producto funciona correctamente y que

cumple con los requerimientos del usuario. Las pruebas finales del producto son realizadas por el mismo usuario con la guía del desarrollador, para esto el desarrollador deberá realizar las pruebas de funcionalidad en el ambiente de pruebas para asegurarse que el producto se ha instalado correctamente (pruebas de aseguramiento) se debe exponer el nuevo entregable y validarlo, frente al comité de calidad, con el objetivo de obtener la conformidad de que se han cumplido todos los requerimientos del usuario especificados en el acta de requerimiento. Por ultimo se debe firmar el acta de conformidad siempre y cuando no existan observaciones sobre el entregable.

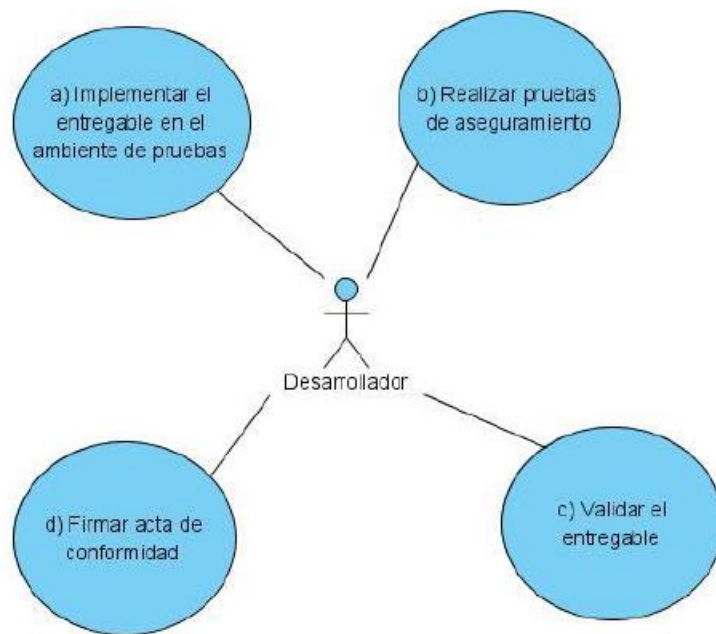


Figura 7: Actividades de la etapa de Pruebas. Fuente: Velásquez (2017)

Implementación: Tomando como punto de partida el modelo de la fase anterior, se procede a programar o implementar el producto entregable. El propósito de esta etapa es instalar el software y los requisitos necesarios para que el entregable pueda correr. En esta fase, se consideran las siguientes actividades: Desplegar los archivos fuente, implementar los objetos de la base de

datos, verificar la funcionalidad del entregable, obtener la conformidad del usuario, firmar el acta de puesta en producción, entregar documentación y firmar el acta de cierre.

Solución de problemas

Una de las acepciones que trae el diccionario de Real Academia de la Lengua Española (RAE) respecto a la palabra Problema es “Planteamiento de una situación cuya respuesta desconocida debe obtenerse a través de métodos científicos”. Con miras a lograr esa respuesta, un problema se puede definir como una situación en la cual se trata de alcanzar una meta y para lograrlo se deben hallar y utilizar unos medios y unas estrategias. La mayoría de problemas tienen algunos elementos en común: un estado inicial; una meta, lo que se pretende lograr; un conjunto de recursos, lo que está permitido hacer o utilizar; y un dominio, el estado actual de conocimientos, habilidades y energía de quien va a resolverlo (Moursund, 1999).

Los algoritmos consisten en aplicar adecuadamente una serie de pasos detallados que aseguran una solución correcta, cada algoritmo es específico de un dominio del conocimiento. La programación de computadores se apoya en este método.

Formular el problema

La solución de un problema debe iniciar por determinar y comprender exactamente en qué consiste dicho problema, en programación es frecuente que quien programa deba formular el problema a partir de los resultados esperados. Es muy importante que el desarrollador sea consciente de que cuando las especificaciones de un programa se comunican mediante lenguaje natural, estas pueden ser ambiguas, incompletas e incongruentes. Para establecer unas metas es necesario identificar la información relevante, ignorar los detalles sin importancia, entender los

elementos del problema y activar el esquema correcto que permita comprender el problema en su totalidad. Determinar con claridad cuál es el resultado final (producto) que deba devolver el programa es algo que ayuda a establecer la meta. Es necesario analizar qué resultados se solicitan y qué formato deben tener esos resultados.

Algoritmo

La palabra algoritmo tiene su origen en el nombre del matemático Persa “Mohamed Ibn Musa Al Khwarizmi” (825 D.C). Su apellido fue traducido al latín como Algorismus y posteriormente paso al español como Algoritmo. Khwarizmi fue bibliotecario en la corte del califa al-Mamun y astrónomo en el observatorio de Bagdad. Sus trabajos de álgebra, aritmética y tablas astronómicas adelantaron enormemente el pensamiento matemático y fue el primero en utilizar la expresión al-yabr (de la que procede la palabra álgebra). Su trabajo con los algoritmos introdujo el método de cálculo utilizando la numeración arábica y la notación decimal.

En el campo de la computación y la informática, los algoritmos son la herramienta que permite describir claramente un conjunto finito de instrucciones, ordenadas secuencialmente y libres de ambigüedad, que debe llevar a cabo un computador para lograr un resultado previsible. Cuando se habla de algoritmos aparecen tres tipos de pensamiento que generalmente se relacionan con ellos y que se utilizan indiscriminadamente como sinónimos: Pensamiento computacional, algorítmico y procedimental.

En términos generales, un algoritmo debe ser realizable, el proceso algorítmico debe terminar después de una cantidad finita de pasos. Se dice que un algoritmo es inaplicable cuando se ejecuta con un conjunto de datos iniciales y el proceso resulta infinito o durante la ejecución se encuentra con un obstáculo insuperable sin arrojar un resultado. El algoritmo debe ser

comprensible, debe ser claro lo que hace, de forma que quien ejecute los pasos (ser humano o máquina) sepa qué, cómo y cuándo hacerlo y por ultimo debe ser preciso, el orden de ejecución de las instrucciones debe estar perfectamente indicado. Cuando se ejecuta varias veces, con los mismos datos iniciales, el resultado debe ser siempre el mismo.

Un algoritmo está compuesto por instrucciones de diferentes tipos, organizadas secuencialmente, en forma de estructuras de control. De estas estructuras, las más comunes son:

- Secuencial.
- Iterativa (repetición).
- Condicional (decisión, selección).

Una estructura de control se define como un esquema que permite representar ideas de manera simplificada y que bajo condiciones normales, es constante (Trejos, 1999). El uso del diseño descendente en los programas, la ejecución de operaciones secuenciales, la utilización de ciclos repetitivos y, la toma de decisiones y alternativas de proceso, ofrecen amplias posibilidades para resolver problemas mediante la construcción de procedimientos.

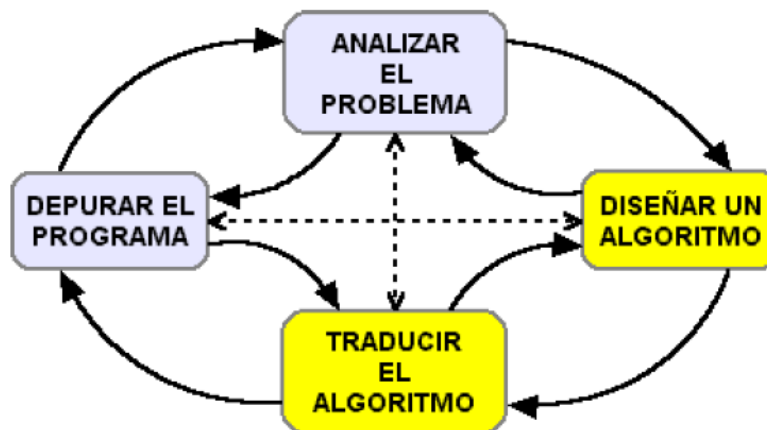


Figura 8: Fases del ciclo de programación. Fuente: López (2007)

Algoritmos genéticos

Nacieron en los 1970 gracias a John Henry Holland, son estrategias de búsqueda estocástica basados en el mecanismo de selección natural, y en algunos casos se involucran aspectos de genética natural, imitando a la evolución biológica como estrategia para resolver problemas. Los algoritmos genéticos difieren de las estrategias de búsqueda convencionales en que estos trabajan sobre un conjunto de potenciales soluciones, llamado población. Esta población es compuesta de una serie de soluciones llamadas individuos y un individuo está conformado por una serie de posiciones que representan cada una de las variables involucradas en los procesos de optimización y que son llamados cromosomas. Estos cromosomas están compuestos por una cadena de símbolos que en muchos casos esta presentada en números binarios.

En un algoritmo genético cada individuo está definido como una estructura de datos que representa una posible solución del espacio de búsqueda del problema. Las estrategias de evolución trabajan sobre los individuos, que representan las soluciones del problema, por lo que estos evolucionan a través de generaciones. Dentro de la población cada individuo es diferenciado de acuerdo con su valor de aptitud, que es obtenido usando algunas medidas de acuerdo con el problema a resolver. Para la obtención de las próximas generaciones se crean nuevos individuos, llamados hijos, utilizando dos estrategias de evolución básicas como son el operador de cruce y el de mutación (empleadas generalmente de forma aleatoria). Una nueva generación es obtenida mediante la utilización del operador de selección, que básicamente se realiza sobre los valores de aptitud de los individuos, sobre la población obtenida tras el cruce y la mutación.

Vocabulario en Algoritmos Genéticos

Toman por su definición terminología utilizada en genética natural y ciencia computacional, por lo que la literatura específica es una combinación entre los términos naturales y los artificiales, por esto se deben tener en cuenta:

- La estructura de codificación con la cual se construyen las soluciones para el problema se llaman cromosomas. Uno o más cromosomas pueden ser requeridos para conformar una población.
- El conjunto completo de cromosomas se llama genotipo y un individuo en particular, un organismo resultante, se llama fenotipo.
- Cada cromosoma contiene una serie de valores individuales llamados genes.
- Los diferentes valores de un gen son llamados alelos.

Componentes de un Algoritmo Genético

Los algoritmos genéticos se pueden caracterizar a través de los siguientes componentes:

- **Definición del problema a optimizar:** Tienen su campo de aplicación importante en problemas de optimización complejos, donde se tiene diferentes parámetros o conjuntos de variables que deben ser combinadas para su solución. También se enmarcan en este campo problemas con muchas restricciones y problemas con espacios de búsqueda muy grandes. Gracias a ciertas características de los algoritmos genéticos, estos no requieren información matemática adicional sobre optimización, por lo que pueden tomar otro tipo de funciones objetivo y todo tipo de restricciones

(lineales y no lineales) definidas sobre espacios discretos, continuos o espacios de búsqueda combinados.

- **Representación para la solución del problema:** Para la solución de un problema en particular, es importante definir una estructura del cromosoma de acuerdo con el espacio de búsqueda. En el caso de los algoritmos genéticos la representación más utilizada es la representación binaria, debido a su facilidad de manipulación por los operadores genéticos, su fácil transformación en números enteros o reales, además de la facilidad que da para la demostración de sus teoremas. La codificación de números reales en cadenas de números binarios se debe hacer teniendo en cuenta el siguiente proceso y la precisión requerida para la solución del problema:

$\mathbf{X}_i \in [\mathbf{a}_i, \mathbf{b}_i]$ en donde el rango de la variable de quedar dividida en $(\mathbf{b}_i - \mathbf{a}_i) \cdot 10^p$,

Donde p es la precisión requerida por el proceso (determinando así su potencia)

La longitud de la cadena binaria de caracteres denotada por \mathbf{m}_i para cada variable

es calculada como $2^{\mathbf{m}_i - 1} < (\mathbf{b}_i - \mathbf{a}_i) \cdot 10^p < 2^{\mathbf{m}_i}$

- **Decodificación del cromosoma:** Para determinar el número real que la cadena binaria de caracteres representa. Es importante tener en cuenta que la cadena no representa un número real, sino que este número binario etiqueta un número dentro del intervalo inicialmente fijado.
- **Evaluación de un individuo:** Nos muestra el valor de aptitud de cada uno de los individuos. Esta aptitud viene dada por una función que es la unión entre el mundo natural y el problema a resolver matemáticamente. Esta función de aptitud es particular para cada problema a resolver y representa para un algoritmo genético lo que el medio ambiente representa para los humanos.

Algoritmos evolutivos

Se describe como sistemas de resolución de problemas de optimización o búsqueda basados en el ordenador empleando modelos computacionales de algún mecanismo de evolución conocido como elemento clave en su diseño e implementación. Trabajan con una población de individuos, que representan soluciones candidatas a un problema. Esta población se somete a ciertas transformaciones y después a un proceso de selección que favorece a los mejores; cada ciclo de transformación y selección constituye una generación, de forma que después de cierto número de generaciones se espera que el mejor individuo de la población esté cerca de la solución buscada. Los algoritmos evolutivos combinan la búsqueda aleatoria, dada por las transformaciones de la población, con una búsqueda dirigida dada por la selección.

Principales componentes de los algoritmos evolutivos

- Población de individuos, que son una representación (no necesariamente directa) de posibles soluciones
- Procedimiento de selección basado en la aptitud de los individuos para resolver el problema
- Procedimiento de transformación para construir nuevos individuos a partir de los anteriores.

Estructura general de los algoritmos evolutivos

t :=0;

Inicializar P(t);

Evaluar $P(t)$

While no se termine do

Begin

$P'(t) := \text{variación } [P(t)];$

Evaluar $[P'(t)];$

$P(t+1) := \text{seleccionar } [P'(t) \cup Q];$

$t := t+1;$

end

Los algoritmos genéticos modelan el proceso de evolución como una sucesión de frecuentes cambios en los genes, con soluciones análogas a cromosomas. Trabajan con una población de cadenas binarias para la representación del problema, y el espacio de soluciones posibles es explorado aplicando transformaciones a éstas soluciones candidatas tal y como se observa en los organismos vivientes: cruce, inversión y mutación. Como método de selección emplean en mecanismo de la ruleta (a veces con elitismo). Constituyen el paradigma más completo de la computación evolutiva ya que resumen de modo natural todas las ideas fundamentales de dicho enfoque. Son muy flexibles ya que pueden adoptar con facilidad nuevas ideas, generales o específicas, que surjan dentro del campo de la computación evolutiva. Además, se pueden hibridar fácilmente con otros paradigmas y enfoques, aunque no tengan ninguna relación con la computación evolutiva. Se trata del paradigma con mayor base teórica.

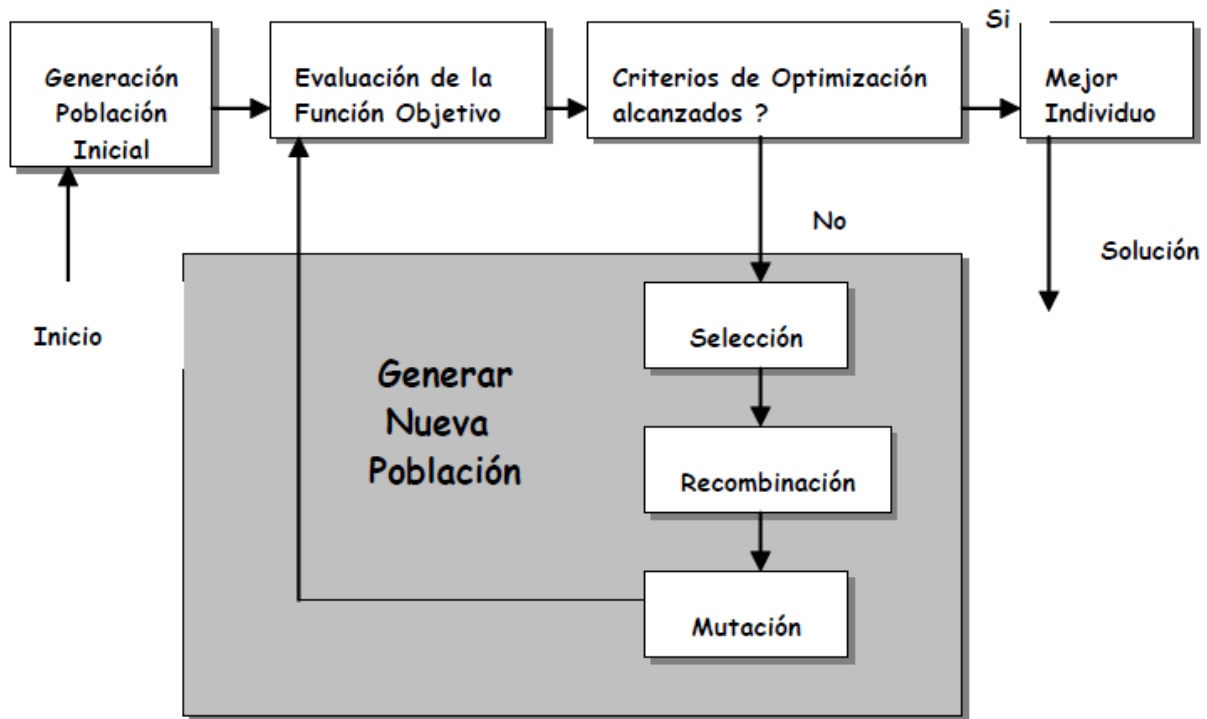


Figura 9: Diagrama de funcionamiento de los algoritmos genéticos. Fuente: Mateos (2009)

Programación Genética

Desarrollada por Koza, los individuos son programas o autómatas de longitud variable, descritos mediante Expresiones-S de LISP representadas habitualmente como árboles, y como operadores de variación emplea crossover y modificación, además de mecanismos de selección:

Algoritmo	Representación del problema	Operadores de variación	Métodos de selección
Algoritmos genéticos (Goldberg)	Cadena binaria	Mutación y crossover	Selección de rueda de ruleta (a veces con elitismo)
Estrategias de Evolución (Rechenberg/Schwefel)	Vector de reales + desviaciones estándar	Mutación gaussiana y crossover aritmético (diferentes tipos)	Diferentes tipos de selección: lambda,mu; lambda+mu...
Programación evolutiva (Fogel)	Números reales	Mutación	Diversos tipos de selección
Programación genética (Koza)	Expresiones-S de LISP representadas habitualmente como árboles	Crossover, algo de mutación	Diversos tipos de selección

DEAP (software) como herramienta para la programación genética

Los algoritmos evolutivos distribuidos en Python (DEAP) son un marco de cálculo evolutivo para la creación rápida de prototipos y pruebas de ideas. Incorpora las estructuras de datos y herramientas necesarias para implementar técnicas de computación evolutiva más comunes, como algoritmo genético, la programación genética, estrategias de evolución, la optimización de enjambre de partículas, evolución diferencial, flujo de tráfico y la estimación de algoritmo de distribución.

Optimización por enjambre de partículas

Este concepto, conocido por sus siglas en inglés PSO (Particle Swarm Optimization) hace referencia a una metaheurística que evoca el comportamiento de las partículas en la naturaleza, estos métodos se atribuyen originalmente a los investigadores Kennedy, Eberhart y Shi, en un principio fueron concebidos para elaborar modelos de conductas sociales, como el movimiento descrito por los organismos vivos en una bandada de aves o un banco de peces, posteriormente el algoritmo se simplificó y se comprobó que era adecuado para problemas de optimización.

Las partículas son denotadas como la optimización de un problema a partir de una población de soluciones candidatas, moviendo éstas por todo el espacio de búsqueda según reglas matemáticas que tienen en cuenta la posición y la velocidad de las partículas. El movimiento de cada partícula se ve influido por su mejor posición local hallada hasta el momento, así como por las mejores posiciones globales encontradas por otras partículas a medida que recorren el espacio de búsqueda. El fundamento teórico de esto es hacer que la nube de partículas converja rápidamente hacia las mejores soluciones.

Programación genética haciendo uso de la herramienta DEAP

- **Módulos:** Antes de escribir funciones y algoritmos, se debe importar algún módulo de la biblioteca estándar y de DEAP.

```
import operator
import random

import numpy

from deap import base
from deap import benchmarks
from deap import creator
from deap import tools
```

Ejemplo 1: Fundamentos de optimización por enjambre de partículas. Fuente: Matheus J.

(2019)

- **Representación:** El objetivo de la partícula es maximizar el valor de retorno de la función en su posición, dichas partículas se describen esencialmente como posiciones en un espacio de búsqueda de dimensiones. Cada partícula también tiene un vector que representa la velocidad de la partícula en cada dimensión. Finalmente, cada partícula guarda una referencia al mejor estado en el que ha estado hasta ahora, esto se representa en DEAP por las siguientes dos líneas de código:

```
creator.create("FitnessMax", base.Fitness, weights=(1.0,))
creator.create("Particle", list, fitness=creator.FitnessMax, speed=list,
               smin=None, smax=None, best=None)
```

Ejemplo 2: Fundamentos de optimización por enjambre de partículas. Fuente: Matheus J.

(2019)

Tal como lo muestra el ejemplo, se crearon dos nuevos objetos en el espacio **creator**. Primero se crea un objeto que llamaremos **FitnessMax** y especificamos el **weights** con los valores **(1.0,)**, esto significa que queremos maximizar el valor de la aptitud de nuestras partículas. El segundo objeto que creamos representa nuestra partícula. Lo definimos como una **list** a la que agregamos cinco atributos. El primer atributo es la aptitud de la partícula, el segundo es la velocidad de la partícula que también será una lista, el tercero y cuarto son el límite del valor de la velocidad, y el quinto atributo será una referencia a una copia del mejor estado que se ha observado hasta ahora en la partícula. Dado que la partícula no tiene un estado final que se haya evaluado, la referencia se establece en **None**. Los límites de velocidad también se configuran **None** para permitir el mismo a través de la función **generate()**.

- **Operadores:** El algoritmo PSO utiliza tres operadores: inicializador, actualizador y evaluador. La inicialización consiste en generar una posición aleatoria y una velocidad aleatoria para una partícula, la siguiente función crea una partícula e inicializa sus atributos, excepto el atributo **best**, que se establecerá solo después de la evaluación:

```
def generate(size, pmin, pmax, smin, smax):  
    part = creator.Particle(random.uniform(pmin, pmax) for _ in range(size))  
    part.speed = [random.uniform(smin, smax) for _ in range(size)]  
    part.smin = smin  
    part.smax = smax  
    return part
```

Ejemplo 3: Fundamentos de optimización por enjambre de partículas. Fuente: Matheus J.

(2019)

La función **updateParticle()** calcula la velocidad, luego limita los valores de velocidad entre **smin** y **smax**, finalmente calcula la nueva posición de la partícula.

```

def updateParticle(part, best, phi1, phi2):
    u1 = (random.uniform(0, phi1) for _ in range(len(part)))
    u2 = (random.uniform(0, phi2) for _ in range(len(part)))
    v_u1 = map(operator.mul, u1, map(operator.sub, part.best, part))
    v_u2 = map(operator.mul, u2, map(operator.sub, best, part))
    part.speed = list(map(operator.add, part.speed, map(operator.add, v_u1, v_u2)))
    for i, speed in enumerate(part.speed):
        if speed < part.smin:
            part.speed[i] = part.smin
        elif speed > part.smax:
            part.speed[i] = part.smax
    part[:] = list(map(operator.add, part, part.speed))

```

Ejemplo 4: Fundamentos de optimización por enjambre de partículas. Fuente: Matheus J.

(2019)

Los operadores están registrados en la caja de herramientas con sus parámetros. El valor de la partícula al principio está en el rango (y), la velocidad está limitada en el rango a través de toda la evolución [-100, 100] pmin pmax [-50, 50]

La función de evaluación **h1()** es de [Knoek2003]. La función ya está definida en el módulo de puntos de referencia, por lo que podemos registrarla directamente.

```

toolbox = base.Toolbox()
toolbox.register("particle", generate, size=2, pmin=-6, pmax=6, smin=-3, smax=3)
toolbox.register("population", tools.initRepeat, list, toolbox.particle)
toolbox.register("update", updateParticle, phi1=2.0, phi2=2.0)
toolbox.register("evaluate", benchmarks.h1)

```

Ejemplo 5: Fundamentos de optimización por enjambre de partículas. Fuente: Matheus J.

(2019)

- **Algoritmo:** Una vez que los operadores están registrados en la caja de herramientas, podemos activar el algoritmo creando primero una nueva población y luego aplicar el algoritmo original de PSO. La variable contiene la **best** partícula (se conoce como gbest en el algoritmo original).

```

def main():
    pop = toolbox.population(n=5)
    stats = tools.Statistics(lambda ind: ind.fitness.values)
    stats.register("avg", numpy.mean)
    stats.register("std", numpy.std)
    stats.register("min", numpy.min)
    stats.register("max", numpy.max)

    logbook = tools.Logbook()
    logbook.header = ["gen", "evals"] + stats.fields

    GEN = 1000
    best = None

    for g in range(GEN):
        for part in pop:
            part.fitness.values = toolbox.evaluate(part)
            if not part.best or part.best.fitness < part.fitness:
                part.best = creator.Particle(part)
                part.best.fitness.values = part.fitness.values
            if not best or best.fitness < part.fitness:
                best = creator.Particle(part)
                best.fitness.values = part.fitness.values
        for part in pop:
            toolbox.update(part, best)

        # Gather all the fitnesses in one list and print the stats
        logbook.record(gen=g, evals=len(pop), **stats.compile(pop))
        print(logbook.stream)

    return pop, logbook, best

```

Ejemplo 6: Fundamentos de optimización por enjambre de partículas. Fuente: Matheus J.

(2019)

Desarrollo de los sistemas expertos

Fases en el desarrollo del sistema experto:

- Identificación de la aplicación y concepción de la solución.
- Desarrollo de los prototipos.
- Ejecución y construcción del sistema completo.
- Integración del nuevo sistema con otros y asimilación del uso de nueva tecnología.

Clasificación de herramientas de desarrollo según etapa de proyecto en que se usan

- **Estudio de viabilidad:** El uso de herramientas flexibles y sencillas, no específicas para resolver el problema apropiado para el sistema experto.
- **Desarrollo del prototipo:** Utilizando las herramientas con una serie de características a nivel de estructuras de control, procediendo a probar y comparar los diferentes enfoques de la resolución del problema y facilidades a los ingenieros del conocimiento.
- **Desarrollo del sistema:** Explorando todos los campos de eficiencia en el sistema experto, además de los módulos explicativos evolucionados.

Tipos de representación del conocimiento (Según sea el problema)

- **Representación procedural:** Expresa explícitamente las relaciones entre los fragmentos de conocimiento, son consideradas complejas de modificar.
- **Representación declarativa:** Se enfoca al cálculo de predicados, a las reglas de producción y las redes semánticas. El conocimiento queda estructurado en fragmentos independientes unos de otros, así son fácilmente modificables y se combinan con un mecanismo general.
- **Representación mixta:** Objetos y marcos. Emplea los dos métodos anteriores.

Ejemplos de representación

- **Reglas de producción:** Consiste en la representación de una relación, una información semántica o una acción condicional. Es lo más habitual (sea intuitivo, rápido y flexible), si existe una premisa entonces existe una consecuencia; cuantas más reglas se puede proporcionar más potencia, sin embargo, demasiadas reglas puede poner en riesgo de perder la coherencia.

- **Lógica proposicional:** Es similar a reglas de producción, pero separa los elementos que componen la base de conocimiento de los que controlan la operación del sistema.
- **Redes semánticas:** Representa el conocimiento en forma de redes o grafos; los nodos representan conceptos u objetos, y los arcos describen relaciones entre ellos.
- **Marcos:** Permite asociar las propiedades a los nodos que representan los conceptos u objetos. Las propiedades son descritas en términos de atributos y los valores asociados.
- **Objetos:** A diferencia de los marcos, en un entorno de programación orientada a objetos, pueden actuar como entidades independientes. Así, el control del sistema se consigue enviando mensajes del sistema a los objetos y entre ellos mismos.
- **Representaciones múltiples:** Permite combinar dos o más formas de representar el conocimiento en el mismo sistema.

Lenguajes y generadores de sistemas expertos

- **Lenguajes de propósito general:** Se encuentran lenguajes como C, PASCAL, entre otros.
- **LISP:** Siendo el más antiguo, componiéndose de estructuras de datos, listas, implantación rápida de modelos y fácil modificación, tomando en cuenta como desventaja su lentitud.
- **Prolog:** Es un lenguaje declarativo, es decir no describe secuencialmente el algoritmo, a partir de datos deduce nuevos hechos y resuelve problemas, incluyendo el motor de inferencia, su principal aplicación es el manejo de bases de datos relacionales.

- **Lenguajes orientados a objetos:** Son habituales por su buen rendimiento, tomando como ejemplo principal a C++, sus características o utilidades más resaltantes son las posibilidades para el desarrollo de prototipos, un motor de inferencia, un método o varios para la representación del conocimiento, una o varias técnicas para el manejo de la incertidumbre y un generador de interfaces gráficas.

Integración y migración de los sistemas expertos: Existen dos tipos básicos de arquitectura de integración.

- El sistema experto forma parte de otro sistema principal (comunicación con el sistema experto directa o red).
- El sistema experto es el sistema principal y está conectado a otros sistemas que le ayudan (ejemplo: subsistemas de complejos cálculos matemáticos necesarios en razonamiento), comunicación fluida (crítico en sistemas expertos en tiempo real), especial relevancia en conexión con bases de datos, portabilidad de la herramienta.

Tendencias futuras de los sistemas expertos

Según el alcance de la herramienta:

- **Herramientas de propósito general:** No especialmente para problemas específicos.
- **Orientadas a dominios y problemas específicos:** Abordan problemas específicos.

Según lenguaje o técnica para el que sirve la herramienta:

- **Herramientas CBR (Case-Based Reasoning):** Basadas en casos, para a partir de ellos inducir las reglas y criterios por similitud.

- **Herramientas para LISP, Prolog, C++:** Son entornos de desarrollo; proporcionan las utilidades necesarias para programar cómodamente en estos lenguajes.
- **SHELLS:** Entornos de desarrollo completo y particular, incluyen motor de inferencia, y usan métodos propios de presentación del conocimiento cercanos a LISP o Prolog. Especialmente para prototipos.

Campos actuales de investigación y de aplicación

- **Aprendizaje:** Que el propio ordenador adquiera el conocimiento.
- **Redes neuronales:** Consiste en nodos conectados con otros mediante enlaces simulando conexiones de neuronas (visión artificial, razonamiento, aprendizaje, lenguaje).
- **Redes Bayesianas:** Es una técnica para tratar el razonamiento con incertidumbre. Una red donde los nodos son hechos ciertos o no y los enlaces entre nodos son las probabilidades condicionadas de unos hechos con otros. Propagando las probabilidades, se pueden obtener los resultados más probables a partir de los hechos.
- **Otros:** Encontramos el data mining, para obtener conocimiento en bases de datos. Los agentes inteligentes para recuperar información en internet.

Aplicaciones de los sistemas expertos

- **Factores que justifican el desarrollo de un sistema experto:** Existen factores que permiten el desarrollo de un SE para un caso específico, el coste para formar nuevos expertos es muy alto, los riesgos de perder los conocimientos y la experiencia en posesión de los expertos, los expertos son escasos, disponer de expertos sin

restricciones de tiempo y lugar y por último, los expertos tienen que trabajar en ambientes hostiles.

- **Tipos de problemas que hacen apropiado el desarrollo de un sistema experto:** Se requiere principalmente tratamiento y razonamiento simbólico, se hace apropiado el desarrollo de un SE cuando las tareas admiten el uso de métodos Heurísticos, las tareas no son demasiado fáciles o las tareas tienen un valor eminentemente práctico. Por otro si el número de conceptos a ser manejados es limitado y no requiere saber de muchas áreas diferentes, se puede desarrollar un SE.
- **Campo de aplicaciones de los sistemas expertos:** Según el tipo de problema que se requiere resolver; los sistemas de ayuda a la toma de decisiones sirven para cierta problemática, sugieren la solución que consideran más idónea a partir del conocimiento, incluido en el sistema. La configuración permite seleccionar y planificar los componentes necesarios en un proceso. Para el diagnóstico se utiliza a partir de unos “síntomas” que determinan las causas que lo producen, la planificación que establece las etapas y recursos necesarios para alcanzar un objetivo, las interfaces inteligentes que hacen de puente entre las personas y equipos complejos y de difícil utilización (como el caso de las interfaces inteligentes de acceso a base de datos). La interpretación y análisis para tratar grandes volúmenes de información y por último la monitorización de procesos dando una salida de control en tiempo real.

Simulación

Se entiende por simulación el acto que consiste en imitar o fingir que se está realizando una acción cuando en realidad no se está llevando a cabo. En el campo de la computación, la simulación, tiene todavía mayor significado especializado: Alan Turing usó el término

“simulación” para referirse a lo que pasa cuando una computadora digital corre una tabla de estado (corre un programa) que describe las transiciones de estado, las entradas y salidas de una máquina sujeta a discreto-estado. En programación, un simulador es a menudo usado para ejecutar un programa que tiene que correr en ciertos tipos de inconvenientes de computadora o en un riguroso controlador de prueba de ambiente. En los sistemas expertos y su proceso de simulación se llevan a cabo un método, de 3 fases, que consisten en actividades de dos tipos:

- **Actividades B (Incondicionales):** Se puede decir en qué instante han de comenzar (pueden programarse). Pueden ser tratadas por el ejecutivo cada vez que llega su instante.
- **Actividades C (Condicionadas):** Cuando su realización depende de condiciones en cada instante (cooperativas) su ejecución puede depender de la cooperación de diferentes entidades o de la satisfacción de condiciones específicas.

Simulación en cola simple:

- **Llegada de un cliente (Actividad B):** Encadenando llegadas puede hacerse que la llegada N-ÉSIMA se produzca al final del intervalo o entre el cliente N-1 y el N.
- **Final del servicio (Actividad B):** Cuando se inicia un servicio puede determinarse a partir de la distribución de probabilidad su finalización y las actividades consecuentes (abandono sistema y liberación de servidor) ocurrirán entonces.
- **Inicio del servicio (Actividad C):** Sólo si se satisfacen dos condiciones o reglas, debe haber al menos un cliente esperando y el servidor debe estar libre.

Ejecutivo del planteamiento de las tres fases

Fase A (Exploración del tiempo): Determina cuándo debe ocurrir el siguiente suceso B, cuál debe ejecutarse, avanza el reloj a ese instante, se mantiene allí hasta completar B y C.

Fase B (llamada a las actividades B): En esta fase se verifica que actividad B debe ejecutarse y la realiza (varias, todas y prioridades).

Fase C (exploración de las actividades C): Intenta activar cada Actividad C y realiza las que se satisfacen condiciones.

Corriendo como Sistema Experto

En base de conocimientos: Actividades C (reglas) y actividades B (“demons”)

En base de datos o base de hechos: Información sobre el estado actual del sistema. Aquí se corre sobre los hechos como estados de las entidades (usados por las reglas para extraer inferencias) y metas, como elementos de mecanismo temporal que representan programación de Actividades B.

En motor de inferencia: El ejecutivo controla el tiempo, las condiciones de interrupción del proceso, se llama a los “demons” B y verifica las reglas C. Permite encontrar las reglas satisfechas por los datos de la base de hechos en ese momento y aplica una estrategia de selección que procede a una aplicación secuencial de meta reglas.

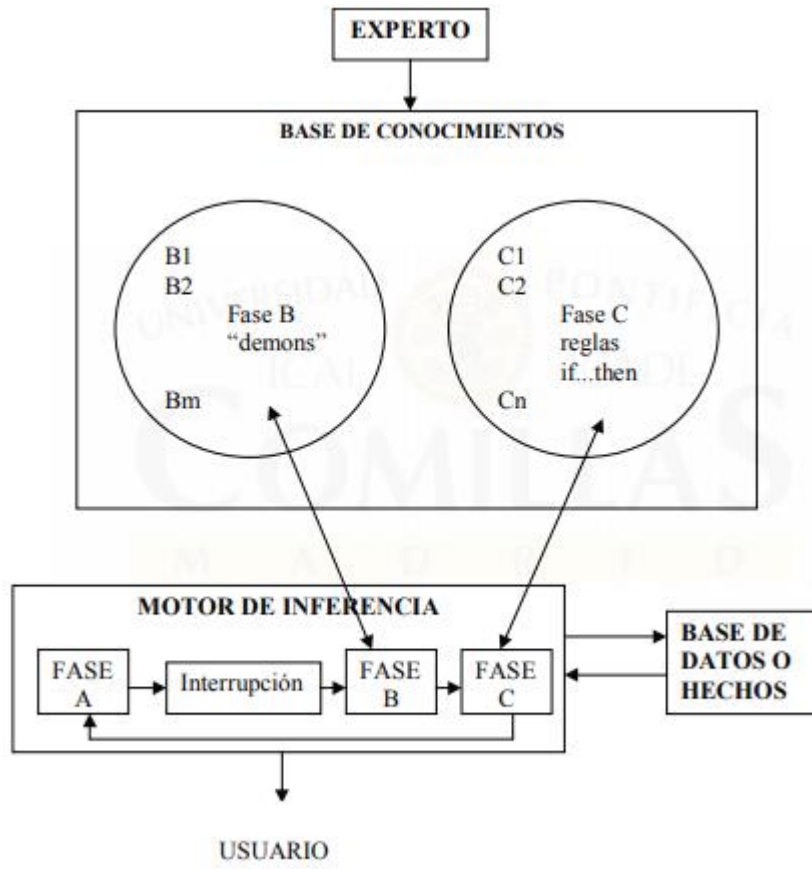


Figura 10: Simulación de un sistema experto. Fuente: Comillas (2011)

CAPÍTULO III

MARCO METODOLÓGICO

Tipo y diseño de la investigación

Acorde con el Manual de Trabajos de Grado de especialización y Maestría y Tesis Doctorales de la UPEL (2016), esta investigación entra en la modalidad de proyecto factible, ya que el objetivo principal fue el desarrollo de una propuesta de un modelo operativo viable para la solución de problemas, necesidades o peticiones requeridas.

Diseño de la investigación

Esta investigación está fundamentada en un diseño mixto que se compone de acceso al campo y lo documental, tomando principalmente la documentación y el efecto prioritario que corresponde a la resolución de problemas sistemáticos, y en segundo lugar el análisis e interpretación del problema, todo esto con el propósito de describirlos, interpretarlos, comprender el origen y factores constituyentes, entender sus causas, efectos o predecir ciertos eventos.

Se requiere de una investigación de campo para cumplir con estos fines, ya que se realiza un análisis sistemático de los problemas, con el fin de describir y entender su naturaleza, y dar con sus posibles causas y efectos. Con el fin de llegar a una solución ante el problema que se plantea, se requiere una recolección de datos de forma directa a la realidad, para llegar a una hipótesis que permita resolverlo. Para efectuar este análisis de datos es necesario, clasificarlos de acuerdo a una teoría, ordenarlos, correlacionarlos estadísticamente e interpretarlos.

Población y muestra

Población

Para hacer posible el proceso de investigación, es fundamental definir la población objeto de estudio. Según Chávez (2007), la población “es el universo de estudio de la investigación, sobre el cual se pretende generalizar los resultados, constituida por características o estratos que le permiten distinguir los sujetos, unos de otros”.

De igual manera, Corbetta (2007), define a la población como “Un conjunto de N unidades, que constituyen el objeto de un estudio; donde N es el tamaño de la población”.

En esta investigación la población estuvo conformada por 220 estudiantes de ingeniería en computación, y 15 integrantes del personal que integran el centro de cómputo, donde se pueden encontrar todas las redes de comunicaciones, perteneciente a la sede Mirabel, sumando así un total de 235 personas.

Esta población fue seleccionada por el hecho de que al integrar parte de la universidad deben poseer conocimiento de las condiciones que se ofrece en los servicios tanto del sistema académico como de la página principal, tiempos de respuesta y tiempos de actividad e inactividad, entre otras cosas.

Muestra

En cuanto a la muestra, Parra (2003), define la muestra como “Una parte (sub-conjunto) de la población obtenida con el propósito de investigar propiedades que posee la población”.

Esto significa que el tamaño de la muestra está relacionada con los objetivos del estudio y las características de la población; así como también con los recursos y el tiempo que se dispone. En esta investigación se realizó un muestreo no probabilístico.

Según Arias, F. (2006), expresa “Es un procedimiento de selección en el que se desconoce la probabilidad que tienen los elementos de la población para integrar la muestra”. Este tipo de muestreo se clasifica en casual, intencional y por cuotas. Se utilizó de esta clasificación, el muestreo intencional, según Arias F. (2006) expone “En este caso los elementos son escogidos con base en criterios o juicios preestablecidos por el investigador”.

La muestra permitió aplicar el cuestionario destinado a medir el grado de conocimiento que poseen las personas específicas con respecto a las capacidades y rendimiento que tiene esta parte de la institución para proveer un servicio favorable.

Se presenta la fórmula para el cálculo de la muestra de la población a estudiar:

$$n = \frac{Z^2 * N * p * q}{e^2 * (N - 1) + Z^2 * p * q}$$

Primera Ecuación

Donde:

n: Tamaño de la muestra

N: Es el total de elementos que integran la población = 235

Z: zeta crítico = $2^2 = 4$

p: Es la probabilidad de éxito = 50% = 0.5

q: probabilidad de fracaso = 50% = 0.5

e^2 = error estándar = 10% = 0.1

Los datos que se presentan se pueden interpretar de la siguiente forma:

Según Arias, F. (2006), Zeta crítico “es un valor determinado por el nivel de confianza adoptado, elevado al cuadrado. Para un grado de confianza de 95% el coeficiente es igual a 2”.

En el caso de las probabilidades, viene dada por Ramírez Tulio (1999), La probabilidad de ocurrencia de una combinación específica de eventos independientes y mutuamente excluyentes se determina con la distribución binomial, es aquella donde hay dos posibilidades, que se denominan éxito y fracaso.

El error estándar se define como la desviación estándar de la distribución muestral de un estadístico. Por ende, sustituyendo en la fórmula anterior, se obtiene a continuación:

$$n = \frac{4 * 235 * 0.5 * 0.5}{0.1^2 * (235 - 1) + 4 * 0.5 * 0.5} = 70.3 \approx 70 \text{ personas}$$

Con el resultado obtenido, se tiene la muestra necesaria para realizar el estudio en la Universidad Valle Del Momboy, sede Mirabel.

Técnicas e Instrumentos para la recolección de datos

Técnica

Se conoce como la destreza de cómo se obtendrán los datos requeridos para una conclusión final, de acuerdo con Hernández et al. (2006), después de haber definido el diseño de la investigación y su respectiva población, se procedió a la recolección de datos sobre las variables objeto de estudio, para lo cual se desarrolló un instrumento que fue validado previamente por expertos. Se aplicó y se obtuvieron las respuestas respectivas, las cuales fueron registradas, posteriormente se produjo el análisis de los resultados obtenidos.

Para esta investigación se utilizó la encuesta como técnica, para obtener la información necesaria sobre los procesos esenciales de los servidores dentro de la red, y de sus procesos dentro de la institución.

La técnica de encuesta es utilizada extensamente como procedimiento de investigación, ya que permite obtener y elaborar datos de modo rápido y acertado. Permite conocer las dudas más comunes que pueden tener los estudiantes con respecto a los procesos del centro de redes que involucran los servicios y respuestas en el sistema.

Instrumento

Como instrumento fundamental para la obtención de datos requeridos, se utilizó el cuestionario. En relación a esto, Bavaresco (2006), señala que “Es el instrumento que más contiene los detalles de la población que se investiga tales como: variables, dimensiones e indicadores”.

Adicionalmente, Hernández et al. (2006), expresa “El cuestionario es un conjunto de preguntas respecto a una o más variables están sujetas a mediciones sobre lo que se pretende medir”.

Para el cuestionario se elaboraron diez (10) preguntas cerradas, de dualidad, donde se divide y se analizan los datos obtenidos, consultar en (Anexo I). El cuestionario fue utilizado para diagnosticar los conocimientos sobre el procesamiento de datos en redes, como se maneja dicho proceso, así como también los beneficios que aporta el manejo de IA dentro de estos procesos.

Validez y confiabilidad de los instrumentos

Validación del instrumento

Es un instrumento de recolección de datos requerido en un proceso de investigación científica, en relación a este punto, Chávez (2007), mide a la validez como la eficacia con que un instrumento mide lo que pretende el investigador; es decir, la validez de una escala va a estar relacionadas con la confiabilidad del instrumento.

Por su parte, Méndez (2007), define la validez como “El grado en que una prueba mide lo que se propone medir”, este punto es de gran relevancia, ya que, asegurarán al investigador que la información obtenida servirá en su propósito.

Luego de analizar estas opiniones, se considera que para llevar a cabo este proceso, el cuestionario fue analizado, conjuntamente con un formato expuesto de información relacionada con el objeto de estudio, por expertos en la materia, para un total de dos (02) expertos, que permitieron pasar a una fase de revisión constante, permitiendo perfeccionar toda la base de recolección de datos que le darán forma y vida a un sistema experto, dando una aplicación definitiva y concisa del mismo.

Confiabilidad

Con respecto a la confiabilidad de los instrumentos de recolección de datos, Sánchez y Guarisma (1995) plantean que una medición es confiable o segura, cuando es aplicada repetidamente a un mismo individuo o grupo, o al mismo tiempo por investigadores diferentes, da iguales o parecidos resultados”.

Por ello con el fin de revisar, evaluar y determinar la confiabilidad del instrumento, se procede a determinar la confiabilidad del cuestionario a través de una prueba piloto a un grupo de individuos que no fueron incluidos en la muestra, en varias oportunidades, realizadas las aplicaciones se compararon los resultados obtenidos y no se detectaron discrepancias, por ende, se consideró confiable el instrumento de recolección de datos.

Técnica de procesamiento y análisis de datos

Consiste en la técnica de recolección de datos primarios de entrada, son evaluados y ordenador para obtener información útil, que luego serán analizados por el usuario final, para que pueda tomar las decisiones o realizar las acciones que estime conveniente. Según Tamayo (2003) la técnica del procesamiento de datos, “No es otra cosa que el registro de los datos obtenidos por los instrumentos empleados mediante una técnica analítica en la cual se comprueba la hipótesis y se obtienen las conclusiones.

Encinas (1993), expresó “los datos en sí mismos tienen limitada importancia, es necesario hacerlos hablar, en ello consiste, en esencia, el análisis e interpretación de datos”. Es fundamental especificar el tratamiento que se dará a los datos.

De manera que los datos puedan ser clasificados, codificados y de ser posible establecer categorías precisas con ellos se desprende de la siguiente manera:

- Aplicando el instrumento, se procede a revisar cada uno de los cuestionarios aplicados para finalmente estructurarlos, presentando los datos numéricos para la codificación.
- Se procede a realizar la tabulación, que consiste en elaborar tablas o listados de datos que permitan el agrupamiento y contabilización para cada uno de los ítems de los instrumentos.
- La construcción de los cuadros estadísticos denominados: frecuencia y porcentaje, que permitirán una mayor facilidad de entendimiento por el lector no especializado por el tema; cada cuadro debe tener un título y fuente de origen de la información.
- Realización de las gráficas que consisten en presentar los valores numéricos que aparecen en los cuadros.
- Se abre paso al análisis de cada respuesta de los ítems, redactándose en forma clara y precisa, acorde a las preguntas formuladas en el instrumento.

De esta manera, se arma una propuesta para la creación de un sistema experto que permita el diagnóstico de fallas en la red de servidores en la Universidad Valle Del Momboy.

Metodología seleccionada para el desarrollo del sistema experto

Para el desarrollo de software se utilizó el modelo incremental o ciclo de vida incremental, este modelo facilita la tarea del desarrollo permitiendo al desarrollador producir un módulo particular. Es una repetición del ciclo de vida en cascada, aplicándose este ciclo en cada funcionalidad del programa a construir. Al final de cada ciclo se hace entrega de una versión al

cliente que contiene una nueva funcionalidad. Este ciclo de vida nos permite realizar una entrega al cliente antes de terminar el proyecto.

Haciendo uso del lenguaje de programación Java (lenguaje orientado a objetos) mediante el entorno de desarrollo integrado NetBeans (hecho principalmente para el lenguaje de programación Java). La metodología aplica secuencias lineales de forma escalonada mientras progresa el tiempo en el calendario. Cada secuencia lineal produce un incremento del software. Genera algunos beneficios tales como:

- Construir un sistema pequeño que permita no percibir un riesgo alto.
 - Al desarrollar independientemente las funcionalidades, es más fácil relevar los requerimientos del usuario.
 - Si se detecta un error grave, sólo desechamos la última iteración.
 - No es necesario disponer de los requerimientos de todas las funcionalidades en el comienzo del proyecto y además facilita la labor del desarrollo con la conocida filosofía de divide y conquista.
-
- Incremento I : Análisis – Diseño – Codificación – Pruebas – Implementación.
 - Incremento II: Análisis – Diseño – Codificación – Pruebas – Implementación.
 - Incremento III: Análisis – Diseño – Codificación – Pruebas – Implementación.

CAPÍTULO IV

ANÁLISIS DE LOS RESULTADOS

Esta parte de la investigación se realiza luego de emplear el instrumento, al terminar la recolección de datos para posteriormente aplicar el análisis de los datos y de esta manera dar respuesta a las interrogantes de la investigación. Arias (2004) expone “En este punto se describen las distintas operaciones a las que serán sometidos los datos que se obtengan”.

Se presenta una determinación en la evaluación crítica de los resultados, para ello, se entrelazan los datos y resultados que se encontraron en la investigación con los datos o información de la base teórica y los antecedentes. Azuaje (1997), expone que el análisis cualitativo, consiste en “la búsqueda de significados y sentido a la información con relación al contexto dentro del cual se desarrolla el estudio”.

El entendimiento de los procesos y servicios entrelazados en redes, por parte de los estudiantes de la institución, el resguardo de los datos y prevención de procesos que puedan perjudicar el correcto funcionamiento de los servidores, se abrirá paso a esto con la capacidad de respuesta que se pueda tener con esta situación. Es fundamental conocer sobre este campo para así moldear una solución acertada y concreta.

ANÁLISIS DE LOS RESULTADOS DEL INSTRUMENTO (CUESTIONARIO)

1. ¿Conoce usted lo que es un servidor?

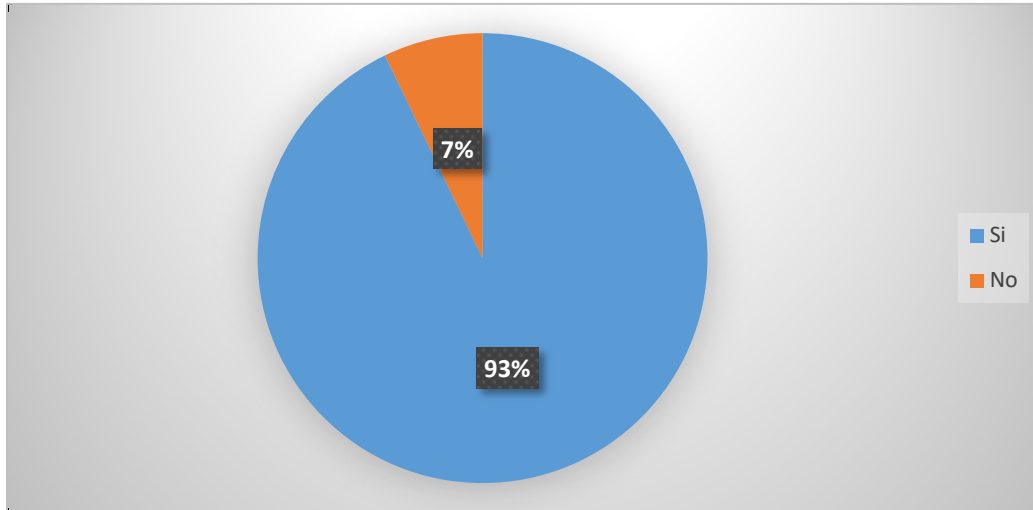


Gráfico 1. Resultados del instrumento. Fuente: Matheus J (2019)

Indicadores	Frecuencia	Porcentaje
Si	65	93%
No	5	7%

Respuesta: Con los datos expuestos anteriormente tenemos que 65 personas encuestadas que representan el 93% de la población poseen conocimientos sobre los servidores. Lo que significa que un 7% de la población desconocen del mismo.

2. ¿Conoce usted lo que es un sistema experto?

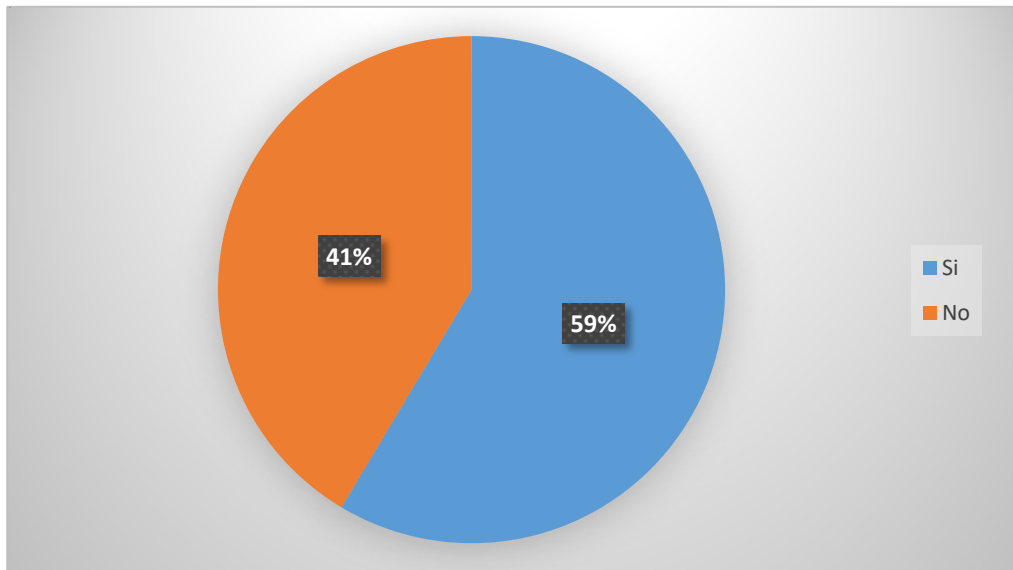


Gráfico 2. Resultados del instrumento. Fuente: Matheus J (2019)

Indicadores	Frecuencia	Porcentaje
Si	41	59%
No	29	41%

Respuesta: Con este resultado presentado se puede apreciar que el 59% de los encuestados tienen conocimientos de lo que es un sistema experto, a diferencia de los que los desconocen con un 41%.

3. ¿Cree usted que diagnosticar fallas en los servidores, de manera anticipada, ayudaría a mejorar los procesos de red?

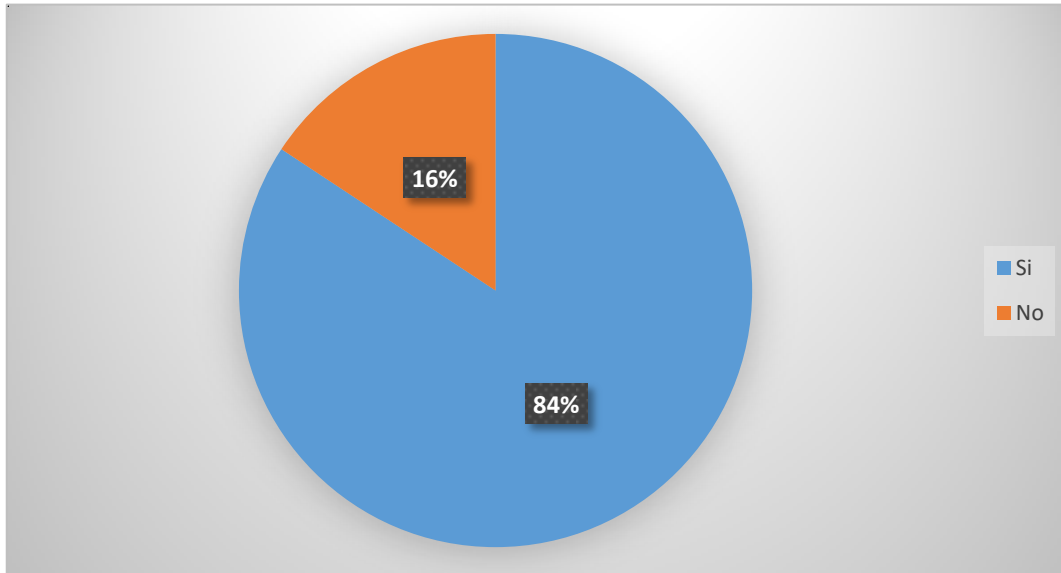


Gráfico 3. Resultados del instrumento. Fuente: Matheus J (2019)

Indicadores	Frecuencia	Porcentaje
Si	59	84%
No	11	16%

Respuesta: Con la gráfica anterior, podemos ver que 59 personas encuestadas que representan un 84% consideran fundamental anticipar fallas en los servidores, y 11 personas que representan el 16% del total consideran que no es prioritario realizar este tipo de prácticas.

4. ¿Consideras que un sistema experto estaría a la altura de cumplir con esta expectativa?

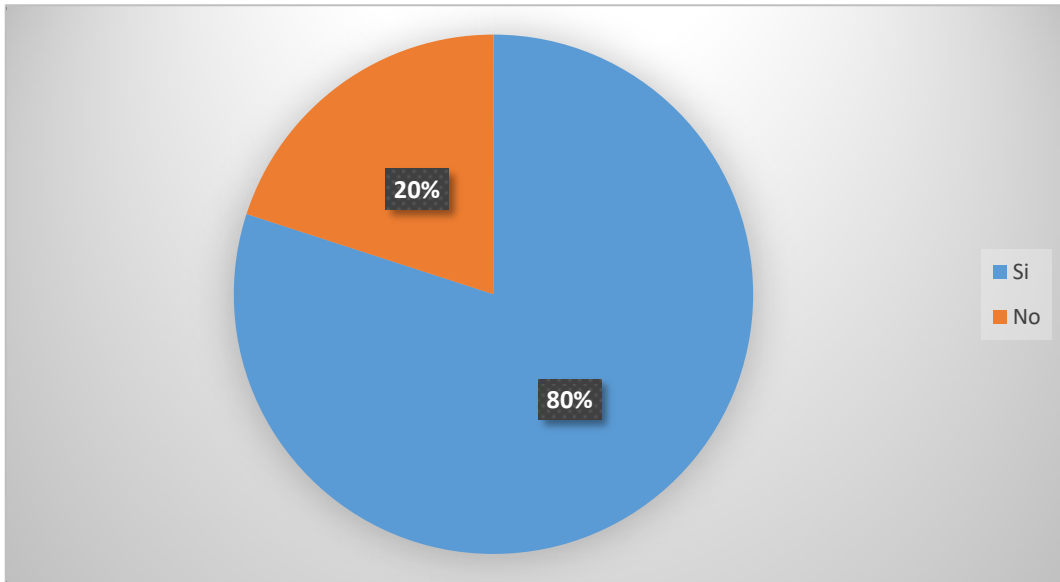


Gráfico 4. Resultados del instrumento. Fuente: Matheus J (2019)

Indicadores	Frecuencia	Porcentaje
Si	56	80%
No	14	20%

Respuesta: Según la gráfica se afirma que 56 personas encuestadas que representan el 80%, consideran que un sistema experto podría aportar soluciones debido a que conocen sus beneficios. Sin embargo 14 personas que representan un 20% consideran que esto no sería posible con un sistema experto.

5. ¿Cree usted que en la universidad se atienden de manera eficiente los problemas de red?

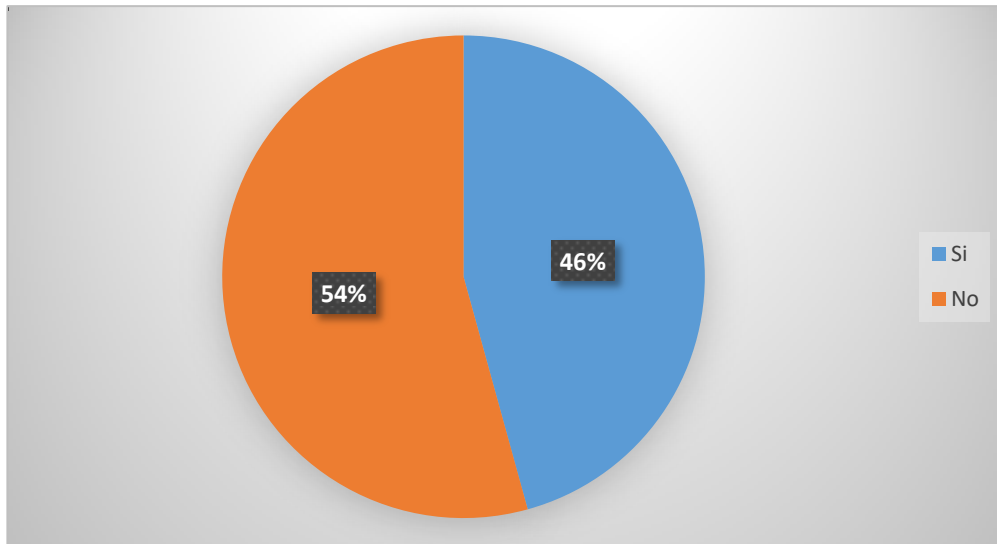


Gráfico 5. Resultados del instrumento. Fuente: Matheus J (2019)

Indicadores	Frecuencia	Porcentaje
Si	32	46%
No	38	54%

Respuesta: Existe un 54% que representan 38 personas encuestadas, que consideran que en la universidad no se atienden de manera eficiente los problemas de red. Por lo que es un porcentaje considerable que presenta inquietud con este servicio, y un 46% que representan 32 personas encuestadas afirma que la universidad si atiende de manera eficiente estos problemas.

6. ¿Estás de acuerdo con que una monitorización basada en inteligencia artificial permita anticipar problemas de red?

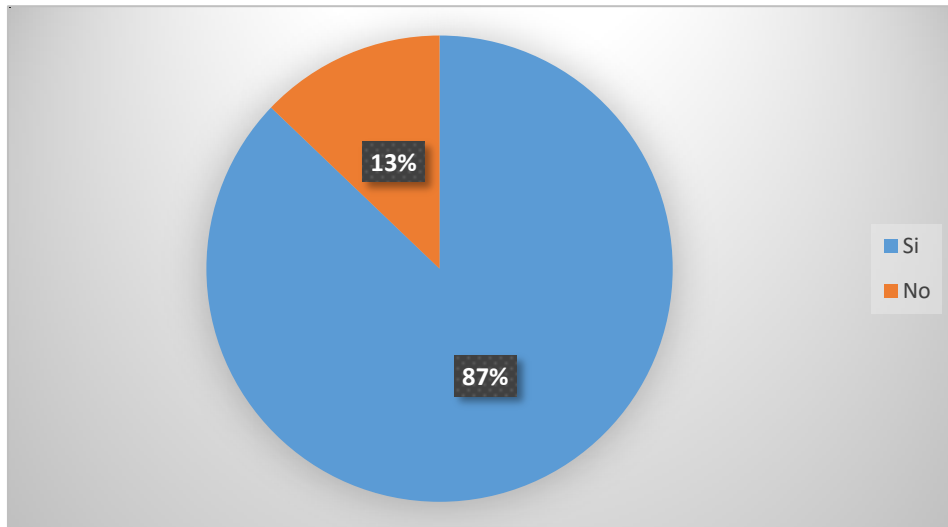


Gráfico 6. Resultados del instrumento. Fuente: Matheus J (2019)

Indicadores	Frecuencia	Porcentaje
Si	61	87%
No	9	13%

Respuesta: En los datos presentados se puede apreciar que un 87% que representa 61 personas encuestadas están de acuerdo con que la monitorización basada en inteligencia artificial haga su aporte por anticipar problemas de red. Y un 13% que representa 9 personas encuestas no están de acuerdo con esto.

7. ¿Consideras que Java es un lenguaje apropiado para poner en marcha la construcción del sistema experto?

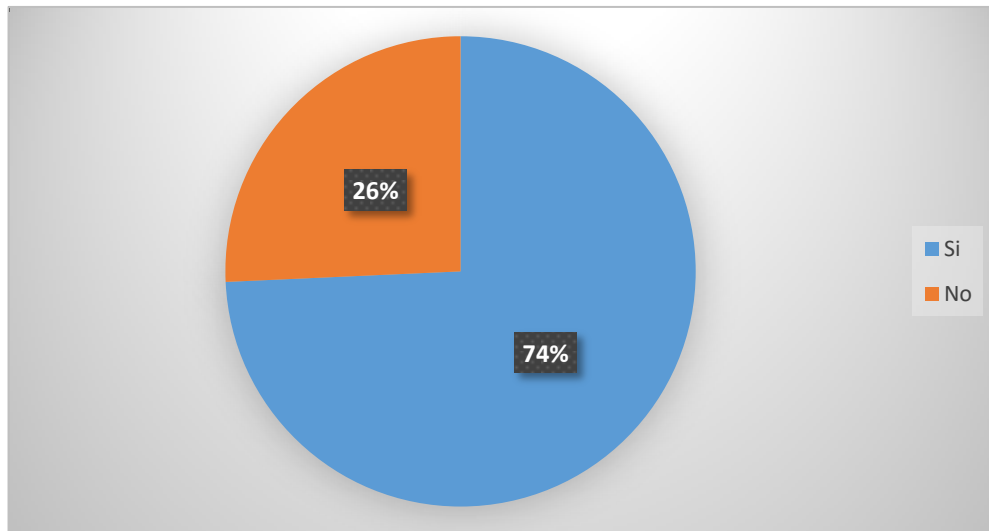


Gráfico 7. Resultados del instrumento. Fuente: Matheus J (2019)

Indicadores	Frecuencia	Porcentaje
Si	52	74%
No	18	26%

Respuesta: Dada la gráfica presentada anteriormente, se puede apreciar que el 74% que representa 52 personas encuestadas considera a Java un lenguaje de programación capaz de cumplir con los requisitos planteados por un sistema experto. Un 26% que representa 18 personas encuestadas no consideran que este lenguaje de programación sea el adecuado.

8. ¿Estás de acuerdo con que la inteligencia artificial sea objeto de estudio en la universidad para poner en marcha nuevos proyectos?

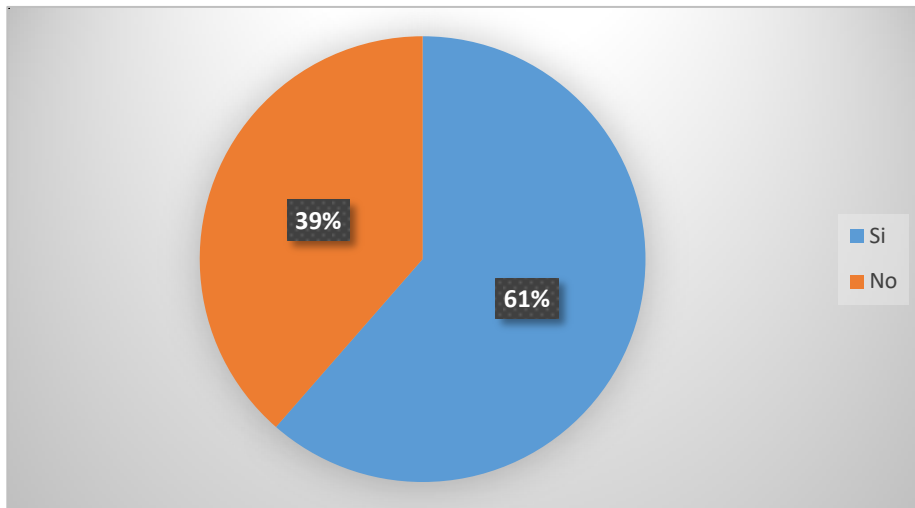


Gráfico 8. Resultados del instrumento. Fuente: Matheus J (2019)

Indicadores	Frecuencia	Porcentaje
Si	43	61%
No	27	39%

Respuesta: En los datos presentados se puede apreciar que un 61% que representa 43 personas encuestadas consideran fundamental que se profundice el estudio de la IA en la universidad para el desarrollo de proyectos basados en este campo. Un 39% que representa 27 personas encuestadas no consideran fundamental que se estudie a profundidad este campo.

9. ¿Estás de acuerdo con la automatización de más procesos dentro de la universidad?

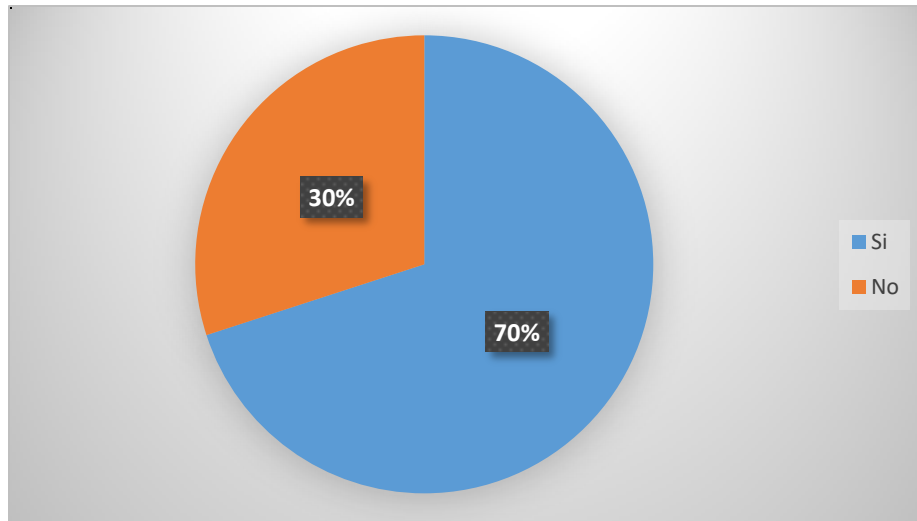


Gráfico 9. Resultados del instrumento. Fuente: Matheus J (2019)

Indicadores	Frecuencia	Porcentaje
Si	49	70%
No	21	30%

Respuesta: Dada la gráfica presentada se puede apreciar que el 70% que representa 49 personas encuestadas están de acuerdo con la automatización de otros procesos dentro de la universidad.

Mientras que el 30% que representa 21 personas encuestadas no están de acuerdo.

10. ¿Le darías apoyo a este proyecto, y reforzarlo de ser necesario?

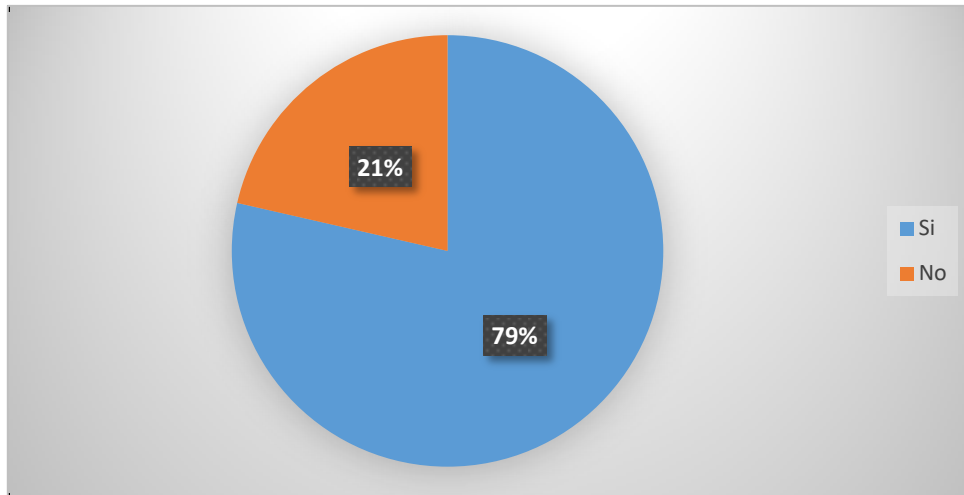


Gráfico 10. Resultados del instrumento. Fuente: Matheus J (2019)

Indicadores	Frecuencia	Porcentaje
Si	55	79%
No	15	21%

Respuesta: Dada la gráfica presentada podemos apreciar que un 79% que representa 55 personas encuestadas seguirían dándole continuidad a este proyecto y otros de este campo, reforzando el desarrollo basado en inteligencia artificial. Mientras que el 21% que representa 15 personas encuestadas no le darían apoyo a este proyecto.

CAPÍTULO V

DESARROLLO DEL SOFTWARE

Finalmente en este capítulo se presenta la metodología propuesta para el desarrollo del sistema experto, que se determinó como la más acertada para cubrir todas sus fases, con el fin de cumplir con el objetivo, procesar los datos a través de preguntas que permitan buscar la solución más óptima de manera inteligente, y cumpliendo con los estándares de sistema experto, a una situación dada (un problema de red).

En el desarrollo de software, la metodología que se expone comprende fases esenciales para la construcción de la misma, a continuación se presenta como se llevó a cabo el proceso para la elaboración del mismo, para posteriormente explicar a detalle los resultados.

Primera fase: Análisis

Es fundamental organizar correctamente las ideas al momento de diseñar el software, para obtener un producto progresivamente correcto, óptimo y seguro se necesita encontrar un análisis que plasme de manera intuitiva la información donde se da a conocer los distintos objetivos a lograr en el desarrollo de software.

Para llevar un proceso estable se deben conocer las variables que se observan de manera directa, y dar con el problema que se desea resolver, dichos problemas en los servidores van orientados a lo siguiente:

- Apagones de manera constantes dentro de un site (cuarto de servidores o cuarto frío).
- Servidor apagado y no enciende.
- El servidor dejó de recibir peticiones de los equipos.

- El servidor no responde el comando Ping.
- Retardo al atender las peticiones de los usuarios.

Segunda fase: Diseño del software

En esta fase se estructura como se verá el software, se aplica de manera independiente al modelo de software que se utilice, una vez analizado y especificado los requisitos, el diseño de software es la acción que corresponde dentro de la actividad de modelado, la cual permite establecer una plataforma para la construcción (como generación de código y prueba). Se diseñaron los prototipos de pantalla que ayudan a normalizar las funcionalidades del sistema, permitiendo construir una interfaz uniforme para finalmente diseñar la estructura de la aplicación.

La interfaz fue desarrollada en el entorno de desarrollo integrado NetBeans, hecho principalmente para el lenguaje de programación Java. Se compone de los siguientes elementos:

Una pantalla inicial, donde se da inicio a las funcionalidades del sistema experto, con una breve descripción, posterior a esto se abre un form que contiene una serie de preguntas que el usuario responde seleccionando la opción que se adapta más al problema presentado con el servidor, empezando a ejecutar así la base de reglas, y con cada respuesta del usuario pasa al siguiente form que llama a la función correspondiente a dicha regla, aquí es donde se contienen los valores que a su vez pasan por la base de conocimiento, que pasan por el motor de inferencia dando así un resultado que será nuestra solución o posible solución al problema presentado en el servidor.

Tercera fase: Codificación

En esta fase se da inicio a lo que vendría siendo el código, tomando en cuenta que se ha diseñado con anterioridad los algoritmos que posteriormente se podrán traducir dichos algoritmos a un lenguaje de programación específico, acciones definidas se convertirán en instrucciones (Haciendo uso de lenguaje de programación Java). El entorno gráfico podrá ser ocupado por el usuario para interactuar con el mismo para finalmente; la parte interna del programa procesará toda la información que se introduzca por dichos usuarios.

Cuarta fase: Pruebas

La metodología que se ha propuesto permite la correcta depuración del sistema al finalizar cada ciclo, con la finalidad de comprobar la integración de las funciones del sistema, su ejecución adecuada con respecto a todos sus atributos, la base de conocimiento, base de reglas, operatividad en el motor de inferencia y procesamiento de información, así como también la posibilidad de encontrar fallas que puedan ser solucionadas o posibles mejoras al código para incrementar su rendimiento. Todo este proceso se hizo respectivamente con el usuario, con la guía del desarrollador para poner en práctica las pruebas de funcionalidad, a su vez obtener mejores resultados a la hora de verificar fallos que ameritan ser atendidos,

Quinta fase: Implementación

Se procede con la programación e implementación del producto entregable que para esta fase se instala el software y los requisitos necesarios para que el producto final

pueda correr, se despliegan los archivos fuente, se verifica la funcionalidad del entregable, con pruebas hechas por el usuario y a su vez obteniendo la conformidad del mismo.

CAPÍTULO VI

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

Tal como esta investigación lo ha demostrado, es fundamental conocer el estado de los sistemas que se manejan dentro de una institución para garantizar el correcto funcionamiento de los mismos, de manera que no se pierda tiempo y dinero. En base a los objetivos específicos planteados se ha logrado identificar las fallas procedentes de la red de servidores de la Universidad, haciendo uso del instrumento analizado (cuestionario) donde se proporcionó información fundamental para el desarrollo de la base de conocimiento, base de reglas y motor de inferencia que permite el funcionamiento lógico del sistema. Generando de esta manera una propuesta que permita una eficiente toma de decisiones con el uso de software inteligente (Sistema Experto).

Con este proyecto realizado se busca dar una respuesta a los problemas presentados en la red de servidores que ocasionan fallas en diversos procesos de red en la Universidad Valle Del Momboy; buscando un diagnóstico que permita emular lo que sería una tarea que realizaría un experto.

Existen casos en donde el comportamiento de la red es inusual o simplemente deja de prestar el servicio y ahí es cuando el sistema experto entra en juego como una herramienta de apoyo que nos ayudará a detectar problemas y dar atención oportuna a sucesos de mayor relevancia. Al brindar información al sistema experto, podemos darnos cuenta en donde se encuentra el problema de una manera más específica. El contar con uno o varios servidores dentro de una organización hace que la monitorización de los mismos sea un

asunto importante; en un sistema automatizado se debe discriminar la información recopilada de estos para que solo se de aviso de los eventos que necesiten atención.

Recomendaciones

Dado lo expuesto anteriormente, se hace fundamental continuar con lo que vendría siendo el desarrollo y avance del sistema experto para el diagnóstico y falla de la red de servidores, resaltando lo siguiente:

- Organizar el área de desarrollo para inteligencia artificial, sistemas expertos, para abarcar todas las áreas que ameriten ser atendidas.
- Darle continuidad al proyecto de sistema experto para el diagnóstico y falla de la red de servidores con la realización de proyectos dentro de la rama de programación, y la elaboración de trabajos de grado similares a nivel universitario.
- Contar con un equipo de expertos en el área de inteligencia artificial, algoritmos genéticos, evolutivos y sistemas expertos para facilitar el mantenimiento de los proyectos.

BIBLIOGRAFÍA

Roger S. Pressman. “Ingeniería del software: Un enfoque práctico” (1988)

David J. Barnes, Michael Kölling. “Programación orientada a objetos con Java” (2003)

Paul Deitel. “Cómo programar en Java” (1996)

Arias, G Fidas. “El Proyecto de Investigación, Introducción a la Metodología Científica.” (2012).

Cristóbal Pareja Flores Manuel Ojeda Aciego Ángel Andeyro Quesada Carlos Rossi Jiménez. “Algoritmos y Programación en Pascal”.

Loebener. “Loebner Contest” (2009).

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest y Clifford Stein. “Introducción a los algoritmos” (1989).

Stair R. M. “Principios de Sistemas de Información: Enfoque Administrativo” (2000).

Turing, A. “Computing Machinery and Intelligence” (1950).

USR.CODE. Implementación y debugging, “Ciclo de Vida del Software”.

Richard Frost. “Bases de datos y sistemas expertos: ingeniería del conocimiento” (1989).

ANEXOS

Anexo 1. Instrumento

Cuestionario para estudiantes y personal del centro de cómputo y redes.

1. ¿Conoce usted lo que es un servidor?

Sí__ No__

2. ¿Conoce usted lo que es un sistema experto?

Sí__ No__

3. ¿Cree usted que diagnosticar fallas en los servidores, de manera anticipada, ayudaría a mejorar los procesos de red?

Sí__ No__

4. ¿Consideras que un sistema experto estaría a la altura de cumplir con esta expectativa?

Sí__ No__

5. ¿Cree usted que en la universidad se atienden de manera eficiente los problemas de red?

Sí__ No__

6. ¿Estás de acuerdo con que una monitorización basada en inteligencia artificial permita anticipar problemas de red?

Sí__ No__

7. ¿Consideras que Java es un lenguaje apropiado para poner en marcha la construcción del sistema experto?

Sí__ No__

8. ¿Estás de acuerdo con que la inteligencia artificial sea objeto de estudio en la universidad para poner en marcha nuevos proyectos?

Sí__ No__

9. ¿Estás de acuerdo con la automatización de más procesos dentro de la universidad?

Sí__ No__

10. ¿Le darías apoyo a este proyecto, y reforzarlo de ser necesario?

Sí__ No__

TABLA DE FIGURAS

Figura 1. Experimento de utilización de Ethernet. Fuente: Spurgeon C. (Ethernet The DefinitiveGuide).....	28
Figura 2. Elementos básicos de la metodología. Fuente: Velásquez (2017)	31
Figura 3. Ciclo de vida del desarrollo. Fuente: Velásquez (2017).....	32
Figura 4. Modelo del proceso de desarrollo de software. Fuente: Velásquez (2017).....	33
Figura 5. Actividades de la etapa de análisis. Fuente: Velásquez (2017).....	34
Figura 6. Actividades de la etapa de diseño. Fuente: Velásquez (2017).....	35
Figura 7. Actividades de la etapa de Pruebas. Fuente: Velásquez (2017).....	36
Figura 8. Fases del ciclo de programación. Fuente: López (2007).....	39
Figura 9. Diagrama de funcionamiento de los algoritmos genéticos.....	45
Figura 10. Simulación de un sistema experto. Fuente: Comillas (2011).....	58

TABLA DE EJEMPLOS

Ejemplo 1. Fundamentos de optimización por enjambre de partículas.....	47
Ejemplo 2. Fundamentos de optimización por enjambre de partículas.....	47
Ejemplo 3. Fundamentos de optimización por enjambre de partículas.....	48
Ejemplo 4. Fundamentos de optimización por enjambre de partículas.....	49
Ejemplo 5. Fundamentos de optimización por enjambre de partículas.....	49
Ejemplo 6. Fundamentos de optimización por enjambre de partículas.....	50

