



innovación y software





Vol. 2 N°. 1 2021 Marzo - Agosto

ISSN N°: 2708-0935

DOI: 10.48168/innosoft.s5

ARK: ark:/42411/s5

PURL: 42411/s5

Depósito Legal: 2023-08884

Periodicidad: Semestral

Publicado: 30/03/2021

Editado por:

Universidad La Salle

RUC: 20456344004

Av. Alfonso Ugarte N° 517, Cercado, Arequipa

COMITÉ EDITORIAL

Editor jefe:	Dr. Yasiel Pérez Vera
Editores asociados:	MSc. Anié Bermudez Peña MSc. Percy Oscar Huertas Niquén
Miembros del Consejo Editorial	Dr. José Manuel Patricio Quintanilla Paulet Hno. Jacobo Meza Rodríguez Dr.C José Javier Zavala Fernández Dr.C Cristian José López del Álamo Dr.C Álvaro Rodolfo Fernández del Carpio MSc. Paul Mauricio Mendoza del Carpio
Corrección de estilos	MSc. Orlando Alonso Mazeyra Guillén
Maquetación	Ronald Fabricio Centeno Cárdenas



EDITORIAL

Las investigaciones y el desarrollo tecnológico en la Ingeniería de Software y las Ciencias de la Computación

p. 4 - 5

ARTÍCULOS ORIGINALES

Design Thinking aplicado al diseño de experiencia de usuario

Autores: Bragean Luis Vargas Márquez, Luis Angel Inga Hanampa, Mauricio Gonzalo Maldonado Portilla

p. 6 - 19

Componente de revisión de estándar de arquitectura de datos para el gestor de bases de datos SQLite

Autores: Sandra Lima Torres

p. 20 - 32

Clasificador de estrellas de Neutrones con una red neuronal multicapa utilizando R

Autores: Luis Angel Aliaga Marica, Edison Wanser Herrera Villa, José Mejía Huayhua, Lizette Quispe Flores

p. 33 - 42

Aplicación del método Design Thinking en el área de requerimientos de software

Autores: Gerson Italo Huarcaya Zapana, Nicolas Herencia Castro, Miguel Angel Sarmiento Tico, Elber Diego Chalco Choquehuanca, Alex Daniel Ticona Bejarano

p. 43 - 52

Dipositivo de hardware libre para Interacción en Entornos 3D

Autores: Jorge Enrique Rodriguez Jimenez, Iván Pérez Mallea, Arianna Rodriguez Jiménez

p. 53 - 63

Proyecto de gestión de redes en BioCen

Autores: Ihosvany Rodríguez González, Mónica Peña Casanova, Anié Peña Bermudez, Ariel O.Famadas García, Alicia C. Blanco Domínguez, Ludibel Mauri Sedeño, Arley Pérez Almario

p. 64 - 82



La Revista Innovación y Software de la Facultad de Ingeniería, en la Universidad La Salle, se complace en presentar este primer número de su segundo volumen que tiene como objetivo el promover investigaciones, los cambios y usos de nuevos elementos tecnológicos y su interrelación con la Ingeniería de Software y la Ciencia de la Computación.

Se puede observar que estas tecnologías han revolucionado el mundo en todos los campos del desarrollo como la sociedad, la economía, la política y la educación, por lo que la importancia de la educación en este campo es cada vez mayor. Este número se centra en investigadores, académicos y estudiantes sobre el avance de la investigación y el desarrollo tecnológico en diferentes campos; propone el avance de la resolución de problemas prácticos en base a diferentes principios.

En la actual pandemia mundial, se hacen evidentes factores clave como el talento y la tecnología respaldados por actividades de investigación destinadas a encontrar soluciones adecuadas a estos problemas. Despierta la sabiduría de las personas y les permite encontrar soluciones viables a problemas que respaldan el uso de modelos, tecnologías y estándares metodológicos de ingeniería de software y ciencias de la computación.

En el futuro, debemos centrarnos en la importancia de la formación y la educación en diferentes campos, en lugar de descuidar las encuestas que apoyan los modelos de apoyo a la ingeniería de software y la informática.

El primer artículo se enfoca en la utilización del *Design Thinking* orientado en el diseño de interfaces de usuario. Generar empatía con nuestros clientes implica poner sus problemas, más que nuestras soluciones, en primer lugar. El diseño de experiencia de usuario es la tarea que debe llevar a cabo el diseñador, que no sabe si el usuario encontrará amigable la interfaz propuesta para este, es por eso que Design Thinking es una de los métodos más usados con el fin de ponerse en la posición del usuario final.

En el segundo artículo se desarrolla un componente para la revisión de la arquitectura de base de datos SQLite. Con el incremento del número de usuarios y de los datos almacenados se hace necesario contar con bases de datos más robustas y seguras. En el presente se realiza un estudio de sistemas gestores de bases de datos homólogos, las herramientas de pruebas a las bases de datos existentes y cómo está conformado un estándar de arquitectura de bases de datos. Se realizan pruebas funcionales al componente para verificar su correcto funcionamiento. Finalmente se obtiene un componente capaz de comprobar automáticamente la arquitectura de las bases de datos del gestor SQLite en las aplicaciones informáticas.

El tercer artículo muestra la investigación realizada utilizando un modelo de inteligencia artificial para clasificar estrellas. Un pulsar es una estrella que emite radiación muy intensa a



intervalos cortos y regulares. Este tipo de estrellas de neutrones que emite radiación periódica y son algo complicadas de detectar. Para esto en el presente artículo se presenta un modelo de red neuronal multicapa capaz de identificarlas con un 98% de precisión.

El cuarto artículo se enfoca en la utilización del *Design Thinking* orientado a la etapa de requisitos de software. Se muestra como integrar las herramientas, características y fases de Design Thinking dentro de las actividades del análisis de requerimientos de software. De esta manera se ayuda en la obtención, análisis, especificación y definición de los requerimientos de software mejorando así el proceso de desarrollo de software.

En el artículo quinto se muestran se presentan dos modelos de diseño y construcción de un dispositivo para la simulación de los movimientos e interacción de los miembros superiores e inferiores en entornos tridimensionales. Se utiliza la plataforma de hardware Arduino como centro de adquisición de datos físicos. Se realiza la selección de sensores y actuadores, así como un análisis de factibilidad y funcionalidad de los mismos.

Finalmente, el sexto artículo trata de la implementación de un proyecto de gestión de redes informáticas. Se presentan las herramientas de gestión utilizadas para el control y monitoreo de los servicios que se prestan en la red, el tráfico interno y externo de la misma; así como la detección de incidentes. Se muestran ejemplos de monitoreo y control con las herramientas de gestión que se aplican en algunas áreas funcionales, así como el diseño de la red local en una organización. Finalmente se abordan las pruebas realizadas al sistema de red, las auditorías internas, el análisis estadístico de los servicios y la respuesta a incidentes.

Comité Editorial



Design Thinking aplicado al diseño de Experiencia de Usuario

6

Design Thinking applied to User Experience Design

Bragean Luis Vargas Márquez

Universidad Nacional de San Agustín

@ bvargasm@unsa.edu.pe

id <https://orcid.org/0000-0001-9552-041X>

Luis Angel Inga Hanampa

Universidad Nacional de San Agustín

@ linga@unsa.edu.pe

id <https://orcid.org/0000-0001-5341-1477>

Mauricio Maldonado Portilla

Universidad Nacional de San Agustín

@ mmaldonadopo@unsa.edu.pe

RECIBIDO 10/10/2020 • ACEPTADO 15/02/2021 • PUBLICADO 30/03/2022

RESUMEN

Hoy en día estamos inmersos en la tecnología y la manera en la que nos comunicamos con esta es la interfaz de usuario, para viabilizar la interacción entre el usuario y un artefacto, sistema o máquina, por tanto, las grandes empresas utilizan técnicas de *Design Thinking* y Diseño de interfaces de usuario, para acercarse más a los clientes de manera empática, así poder conocer sus necesidades reales y estar realmente interesados en los detalles de sus vidas. Generar empatía con nuestros clientes implica poner sus problemas, más que nuestras soluciones, en primer lugar. El problema que se busca resolver en la parte de diseño de experiencia de usuario, es que el diseñador no sabe si el usuario encontrará amigable la interfaz propuesta para este, es por eso que *Design Thinking* es una de los métodos más usados con el fin de ponerse en la posición del usuario final.

En este artículo pretendemos mostrar como *Design Thinking* se puede aplicar al diseño de UX experiencias de usuario. También se explica cada una de las fases que conlleva al diseño de experiencia de usuario: Investigación, Organización, Prototipado, Pruebas y Diseño.

Palabras claves: Cliente, diseñador, experiencia de usuario, interfaz, tecnología, pensamiento de diseño, solución.

ABSTRACT

Today we are immersed in technology and the way in which we communicate with this is the user interface, to enable the interaction between the user and an artifact, system or machine, so large



companies use Design Thinking techniques and Design of user interfaces, to get closer to customers empathically, so that they can know their real needs and be really interested in the details of their lives. Generating empathy with our customers means putting their problems, more than our solutions, first. The problem that is sought to be solved in the user experience design part, is that the designer does not know if the user will find the proposed interface friendly, that is why Design Thinking is one of the most used methods in order to put yourself in the position of the end user. In this article we intend to show how Design Thinking can be applied to the generation of UX user experiences. Each of the phases involved in the generation of the user experience is also explained: Research, Organization, Prototyping, Testing and Design.

Keywords: Customer, design thinking, designer, interface, solutions, technology, user experience.

INTRODUCCIÓN

Los emprendedores y empresarios buscan la forma de hacernos la vida más fácil, como, por ejemplo, transportarnos de un lugar a otro de forma más rápida, como bajar de peso en determinado periodo de tiempo, etc. En otras palabras, se busca diseñar nuestra vida. Se busca diseñar productos y servicios de acuerdo a sus clientes llevando a la pregunta de ¿Qué podríamos hacer para facilitar la vida de nuestros clientes? Es aquí donde las técnicas de *Design Thinking* y Diseño de Experiencia de Usuario entran para cambiarnos la manera de entender nuestros clientes y sus necesidades.

La mayoría de las empresas y negocios crean servicios y productos que luego tratan de “meterle” a su grupo de clientes objetivo, a través de estrategias agresivas de ventas y de publicidad emotiva. Por ejemplo, me pregunto ¿Si la Coca-Cola hubiera sido creada pensando en las necesidades de sus consumidores, sería tan necesaria su multimillonaria inversión anual en publicidad? Todos sabemos lo nocivo que puede ser su consumo excesivo y que realmente no quita la sed, de manera que nos venden felicidad. Este enfoque tradicional ve a los clientes como categorías de compradores y no como personas.

A través del *Design Thinking* y el Diseño de Experiencias de Usuario, nos acercamos de una manera más empática a nuestros clientes, pues esto implica entender sus necesidades reales y estar realmente interesados en los detalles de sus vidas. Generar empatía con nuestros clientes implica poner sus problemas, más que nuestras soluciones, en primer lugar.

ESTADO DEL ARTE

Según [1], se discute áreas donde la nueva investigación y el nuevo plan de estudios para desbloquear el pensamiento de diseño. Para comprender mejor por qué hemos presenciado poca innovación en el diseño, se realizó una encuesta a los profesionales actuales de UX (User



Experience) con respecto a cómo se visualizan y desarrollan los nuevos servicios de ML (Machine Learning) en la práctica de UX. La encuesta probó cómo ML pudo o no haber sido parte de su educación de diseño UX, cómo trabajan para crear cosas nuevas con los desarrolladores y sobre los desafíos que han enfrentado al trabajar con este material.

En [2], acierta que para innovar es preciso no solamente buscar nuevas soluciones tecnológicas, sino que también explorar nuevos mercados. *Design Thinking* se centra en las personas, considerando multidisciplinariedad, colaboración y concreción de pensamientos y procesos, los cuales son caminos que llevan a soluciones innovadoras.

En [3] se describen lo que implicaría introducir las mejores prácticas de diseño e innovación en grandes empresas que se centran en productos de software empresarial. Se propone una teoría de por qué las empresas existentes tienden a centrarse principalmente en los factores de desarrollo técnico e ignoran en gran medida la facilidad de uso de las soluciones resultantes, y aun así han tenido éxito comercial hasta la fecha.

Hacer que los productos salgan a la calle con una experiencia de usuario (UX) fantástica es cada vez más importante en todos los aspectos del mundo empresarial. Las grandes empresas han elevado el estándar en productos de consumo en términos de diseño de experiencia de usuario, que se ha filtrado a organizaciones y contextos no relacionados con el consumidor. Según [4] para lograr un excelente diseño UX no es solo una función o talento de las personas, es una característica organizacional. Comprender el nivel de "madurez" de la organización es un primer paso necesario para mejorar la entrega efectiva del diseño de experiencia de usuario y para permitir que la organización avance al proverbial "siguiente nivel".

La producción rápida de una solución es una necesidad en el mundo de los negocios competitivos de consultoría de TI hoy en día. En los compromisos en los que se necesitan maquetas de diseño de interfaz de usuario para visualizar los procesos de negocio propuestos, la necesidad de crear rápidamente la interfaz de usuario se vuelve muy importante al principio del proceso.

En [5] plantea el objetivo de acelerar el proceso de diseño de la interfaz de usuario, permitiendo la creación rápida de un diseño de interfaz de usuario de baja fidelidad con un pensamiento de diseño tradicional centrado en el usuario, pero con diferentes conceptos de herramientas. Este documento explica el enfoque y la lógica detrás de nuestro modelo y herramientas. Un punto focal clave es aprovechar los modelos de procesos de negocio como punto de partida del proceso de diseño de la interfaz de usuario. El otro punto focal es utilizar un enfoque basado en modelos con herramientas centradas en el diseñador para eliminar algunos gastos generales de diseño, para ayudar a administrar un gran espacio de diseño y para hacer frente a los cambios en los requisitos.



En [6] nos dice que es importante tener en consideración como actividad, a la creación de prototipos. Los prototipos incorporan hipótesis de diseño y permiten a los diseñadores probarlas. El diseño de Framin como una actividad de pensamiento por hacer pone en primer plano la iteración como una preocupación central. Se da a conocer un conjunto de herramientas que incorpora un enfoque centrado en el diseño iterativo para la creación de prototipos de dispositivos de información.

Algunos autores han trabajado el área del diseño de interfaces gráficas de usuario desde diferentes perspectivas. Por ejemplo, Bentley [7] presenta catorce pasos que funcionan como una guía para crear interfaces en sistemas de tipo SAS. Esos pasos tienen en cuenta la selección de un equipo de diseño en el cual puede participar el usuario final, el análisis de las actividades y los usuarios, métodos de creación de ideas, selección de los componentes de la interfaz, prototipado y una evaluación que se realiza después de haber lanzado el sistema.

En [8] presentan una metodología para crear software con altos niveles de usabilidad teniendo en cuenta las metodologías para el diseño de interfaces, 12 arquetipos, análisis en estrella y Análisis y Diseño Orientado a Objetos Complejos (COOAD por sus siglas en inglés), Para identificar las necesidades y habilidades del usuario se hace una descripción y análisis de su contexto y para asegurar la efectividad de las interfaces propuestas se realizan evaluaciones de diferentes prototipos antes de la implementación tecnológica final.

De otro lado, Morales [9] presenta un método que busca evitar errores asociados al descuido en algunos aspectos importantes de las GUI y que se dan por los procesos de creatividad e iniciativa de los diseñadores. La propuesta tiene que ver con capas definidas al inicio del proyecto para que el diseñador tenga en cuenta la organización y relación entre los elementos de la interfaz, cómo el usuario va a interactuar con los componentes y cómo se va a informar al usuario de cambios en la interfaz. Los autores del trabajo proponen crear mínimo un prototipo para cada capa del proceso de desarrollo.

En [10] presenta una interfaz de usuario temprana, que acompaña a la metodología de Desarrollo Temprano de Interfaces de Usuario (EUID por sus siglas en inglés), que tiene como objetivo permitir a los desarrolladores que se concentren en la lógica de la aplicación. La metodología tiene en cuenta una serie de requerimientos que son prototipos y evaluados por el usuario, que, al ser aceptados, los prototipos son diseñados y construidos para el lanzamiento de la interfaz. Este proceso se da de forma iterativa.

Implementación de herramientas de *Design Thinking* y métodos en la resolución de problemas estratégicos y operativos aprendidos por empleados y gerentes. Este enfoque ha sido realizado por GE Healthcare. El objetivo de la empresa era equipar a los empleados con herramientas para gestionar la resolución de problemas con mucha más imaginación y creatividad que antes. La



organización se ha centrado mucho en Six Sigma como un método de mejora, pero quiere cambiar el enfoque de solo la eficiencia de las operaciones hacia la "imaginación en el trabajo". La implementación del nuevo enfoque tiende a ser difícil para algunos, mucho debido a la incomodidad incluida en repensar las formas de trabajar y en considerarse a sí mismo como diseñador de la organización. Los resultados de su mayor actividad de diseño se han descrito como un refuerzo de la innovación y, por lo tanto, el resultado final [3].

DEFINICIONES

Experiencia de Usuario (UX Design)

Una vez tomamos conciencia del objetivo que perseguimos - experiencias de usuario plenas y satisfactorias, la siguiente pregunta lógica es plantearnos el procedimiento, cómo vamos a alcanzar ese objetivo [11]. UX Design hace referencia a una visión o filosofía del diseño en la que el proceso está conducido por información acerca de la audiencia objetiva del producto. La principal diferencia frente a otros enfoques es que su proceso no es secuencial o lineal, sino que presenta ciclos en los que iterativamente se prueba el diseño y se optimiza hasta alcanzar el nivel de calidad requerido.

- **Planificación/Investigación:** Se define conceptualmente el producto en base a la investigación de la audiencia objetiva (necesidades, motivaciones, características, hábitos, modelo mental, actividades...) y al análisis competitivo (qué otros productos existen con audiencias y funciones similares) [11].
- **Diseño/Prototipado:** Se toman decisiones de diseño partiendo de su dimensión más general (arquitectura de información y diseño de interacción) hasta su dimensión más específica (diseño gráfico en detalle y micro-interacciones). Estas decisiones se documentan y se prototipa con objetivos de evaluación [11].
- **Evaluación:** Aquellas decisiones de diseño y procesos críticos del producto se ponen a prueba mediante métodos de evaluación que pueden involucrar a usuarios [11].
- **Implementación:** Una vez el diseño ha alcanzado el nivel requerido de calidad, se procede a su implementación o puesta en producción [11].
- **Monitorización:** Una vez lanzado el producto se estudia el uso que de él hacen los usuarios, con el fin de identificar oportunidades de mejora [11].

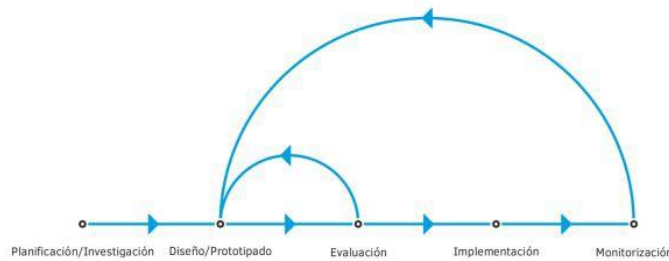


Figura1: Diagrama del UX Design. Fuente Elaboración propia

El UX Design es una filosofía de diseño basada en evidencias. El proceso está, por una parte, conducido por el conocimiento empírico de la audiencia específica a la que se dirige, y, por otra parte, dirigido por principios de diseño que la experiencia y la investigación científica nos ofrece. Frente a otros enfoques como el Diseño Participativo, los usuarios no participan en la toma de decisiones. Es el equipo de diseño el que tiene esa responsabilidad en exclusiva, y es su capacidad para transformar su conocimiento sobre el usuario en decisiones de diseño acertadas la que determinará la experiencia de uso del producto [11].

Design Thinking

Design Thinking o pensamiento de diseño es un modelo de cómo enfocar la innovación en entornos inciertos de forma ágil y radical. *Design Thinking* tiene una serie de herramientas que se utilizan a lo largo del proceso de crear productos y servicios innovadores, en función de la fase en la que se encuentre. Se puede utilizar *Design Thinking* siempre, porque tiene su base en la resolución de problemas, desde el punto de vista del usuario [10].

- Empatizar: Empatía es la base del proceso de diseño que está centrado en las personas y los usuarios. Lo básico para ser empático es:
 - Observar: Mira a los usuarios y sus comportamientos en el contexto de sus vidas. Debemos siempre tratar de observar desde el exterior sin entrometerse, las mejores ideas vienen en estas situaciones así.
 - Involúcrate: Generar una conversación, esta puede ser desde una pregunta de pasillo, breve o una conversación más estructurada. Prepara algunas preguntas para ir manejando la conversación siempre manteniendo levemente estructurada. Lo importante es siempre preguntar “¿Por qué?” ya que eso descubre nuevos significados, preguntar una y dos veces si es necesario ¿Por qué? ¿Por qué?
 - Mira y Escucha: Lo mejor siempre es combinar estas dos, la conversación y el *engagement*. Pídele también que te explique como hace algunas cosas y que vaya vocalizando lo que pasa por su mente cuando esté en su trabajo. Ten una conversación mientras trabaja y esté en su contexto.



Como Diseñador, los problemas que tratas de resolver no son los tuyos, son problemas de otras personas. Para diseñar para estas personas debes adquirir la empatía por lo que ellos son como personas y lo que es importante para ellos. La empatía que es el elemento esencial del proceso de diseño. Entonces entras en un modo, en un estado de observación que es el modo empatía. Que es básicamente el trabajo que haces para entender a los usuarios dentro del contexto del cual estás diseñando. Es el esfuerzo por comprender las cosas que hacen y porqué, sus necesidades físicas y emocionales, como conciben el mundo y que es significativo para ellos. Son las personas en acción las que inspiran al diseñador y direccionan una idea una idea en particular. A esta etapa se le llama "*immerse*" ya que el diseñador debe hundirse en un mar de aprendizaje [6].

- Definir: Este modo "definición" es todo sobre traer claridad y enfoque al espacio de diseño en que se definen y redefinen los conceptos. Es preciso determinar bien el desafío del proyecto basado en lo aprendido del usuario y su contexto. Después de transformarse en un experto instantáneo del problema adquiriendo una empatía invaluable por la persona de la cual estás diseñando, esta etapa es sobre crear coherencia sobre la variada información que se ha reunido.

El modo definición es crítico para el proceso de diseño ya que la meta de esta etapa es moquetear un "Point of View" (POV) que significa crear una declaración de problema viable y significativo y que será guía para enfocarse de mejor manera a un usuario en particular. Los *insights* no aparecen de la nada y repentinamente como por arte de magia. Estos insights nacen al procesar y sintetizar la información y enfrentando el problema para hacer conexiones y descubrir patrones racionales [6]. Esta debe cumplir con ciertos criterios para que funcione bien:

- Enmarcar un problema con un enfoque directo.
- Que sea inspirador para el equipo.
- Que genere criterios para evaluar ideas y contrarrestarlas. - Que capture las mentes y corazones de las personas que has estudiado.
- Que ayude a resolver el problema imposible de desarrollar conceptos que sirven para todo y para todos.
- Idear: Aquí empieza el proceso de diseño y la generación de múltiples ideas. Esta etapa se entrega los conceptos y los recursos para hacer prototipos y crear soluciones innovadoras. Todas las ideas son válidas y se combina todo desde el pensamiento inconsciente y consciente, pensamientos racionales y la imaginación.



Es un espacio para desarrollar *brainstorms* y construir ideas sobre previas ideas. En esta etapa se conciben una gran cantidad de ideas que dan muchas alternativas de donde elegir como posibles soluciones en vez de encontrar una sola mejor solución. También se puede trabajar con métodos como croquis, *mindmaps*, prototipos y *storyboards* para explicar la idea de la mejor manera. Pero el utilizar todas no significa éxito e incluso puede ser peor. A su vez, es necesario también separar el área de generación de ideas con el área de evaluación de ideas [6]. La creación de múltiples ideas permite atacar distintos focos:

- Pensar sobre soluciones que son obvias y por lo tanto aumenta el potencial de innovación del set de posibilidades.
- Aprovechar de mejor manera las distintas visiones de cada equipo de trabajo y el trabajo colectivo.
- Descubrir áreas inesperadas de exploración creando mayor volumen y mayores opciones para innovar.
- Prototipar: El modo Prototipos es la generación de elementos informativos como dibujos, artefactos y objetos con la intención de responder preguntas que nos acerquen a la solución final. no necesariamente debe ser un objeto sino cualquier cosa con que se pueda interactuar. Puede ser un *post-it*, un cartón doblado o una actividad e incluso un storyboard. Idealmente debe ser algo con que el usuario pueda trabajar y experimentar.

Es un proceso de mejora o sea en las fases iniciales de cada proyecto puede ser un poco amplio y el prototipado debe ser de manera rápida y barata de hacer pero que puedan entregar tema para debatir y recibir *feedback* de usuarios y colegas. Este proceso se va refinando mientras el proyecto avanza y los prototipos van mostrando más características como funcionales, formales y de uso [6].

¿Por qué hacer prototipos?

- Para inventar y construir para pensar en resolver el problema.
- Para comunicar. Si una imagen vale mil palabras, un prototipo vale mil imágenes.
- Para empezar conversaciones. Las conversaciones con los usuarios son más eficientes cuando están concentradas sobre algo con que conversar como un objeto.
- Para cometer errores antes y de manera barata.
- Para evaluar las alternativas. Ayuda a desarrollar bien distintas ideas sin tener que comprometerse con una demasiado temprano.



- Para controlar el proceso de la creación de soluciones. Ayuda a identificar distintas variables para poder descomponer grandes problemas que se puedan evaluar y arreglar de mejor forma.

¿Como hacer prototipos?

- Empieza construyendo: Aun cuando no sepas lo que estás haciendo, el solo acto de recoger un material será suficiente para empezar a andar.
 - No le dediques demasiado tiempo a un prototipo: Déjalo ir antes de que te involucres demasiado emocionalmente.
 - Identifica las variables: Cada prototipo debe ir respondiendo preguntas cuando se esté evaluando. Se debe estar atento a las respuestas de la interacción del objeto con el usuario.
 - Trabaja los prototipos con un usuario en la mente: pregúntate ¿Qué esperar evaluar con el usuario? ¿Qué tipo de comportamientos esperas?
- Testear: Este paso consiste en solicitar *feedback* y opiniones sobre los prototipos que se han creado de los mismos usuarios y colegas además de ser otra oportunidad para ganar empatía por las personas de las cuales estas diseñando de otra manera. Una buena regla es siempre hacer un prototipo creyendo que estamos en lo correcto, pero debemos evaluar pensando que estamos equivocados. Esta es la oportunidad para refinar las soluciones y poder mejorarlas. Idealmente se debe evaluar y testear en el contexto mismo del usuario [6].

¿por que evaluar?

- Para refinar prototipos y soluciones. Informa los siguientes pasos y ayuda a iterar, lo que algunas veces significa volver a la mesa de dibujo.
- Para aprender más sobre el usuario. Es otra oportunidad para crear empatía a través de observaciones y *engagement*. Muchas veces entrega inesperados insights.
- Para refinar el Proceso. Algunas veces la evaluación revela que no solo nos equivocamos en la solución, pero también en enmarcar bien el problema.

¿Como evaluar?

- No lo digas, muéstralo: Dales a los usuarios tus prototipos sin explicar nada. Deja que la persona interprete el objeto y observa tanto el uso como el mal uso de lo que le entregas y cómo interactúan con él, posteriormente escucha todo lo que tengan que decir al respecto y responde las preguntas que tengan.



- Crea Experiencias: No es suficiente solo entregarles el objeto, lo ideal es crear el ambiente y recrear la experiencia para tener una visión más acabada del contexto.
- Pídele al usuario que compare: Esto es, entregarle distintos prototipos para probar dándole al usuario una base para poder comparar, esto revela necesidades potenciales.

MODELOS DE DESIGN THINKING

En esta sección, analizamos tres modelos de modelos de pensamiento reportados en la literatura para resaltar las similitudes entre estos modelos y cómo se relacionan con la resolución de problemas que conducen a la creatividad y la innovación, ver las similitudes los puntos fuertes y débiles con el enfoque de la aplicación hacia el diseño de Experiencia de usuario.

Uno de los primeros modelos también llamado diseño basado en investigación Divergente-Convergente o modelo (DCIDT) que describe el pensamiento de diseño como divergente o convergente, el modelo (DCIDT) se muestra en la Figura 2.

En este modelo, los requisitos de diseño se transforman a través de Preguntas de diseño generativo (GDQ) en una serie de conceptos de diseño. Los GDQ se usan para crear, sintetizar y expandir conceptos, que posteriormente se transformaron en decisiones de diseño o especificaciones a través de Preguntas de Razonamiento Profundo (DRQ). El propósito de los DRQ es analizar, evaluar y validar los conceptos de diseño hacia especificaciones viables de decisiones de diseño.

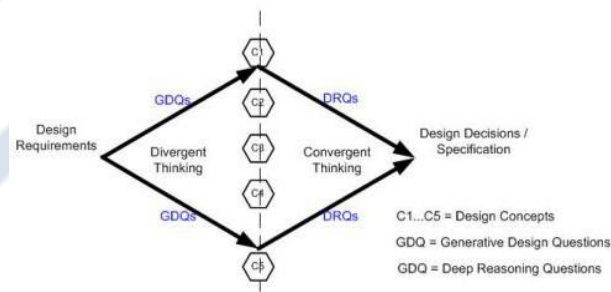


Figura 2: Modelo DT Divergente-Convergente. Fuente [2].

El modelo presentado por Dunn y Martín [6] consta de cuatro actividades, a saber: Abducción, Deducción, Prueba e Inducción (ver Figura 3).



En este modelo, la actividad de Abducción se enfoca en generar ideas y durante la actividad de Deducción, esas ideas serán analizadas para predecir posibles consecuencias. Todas las predicciones serán probadas y el resultado válido se generalizó durante la etapa de inducción.

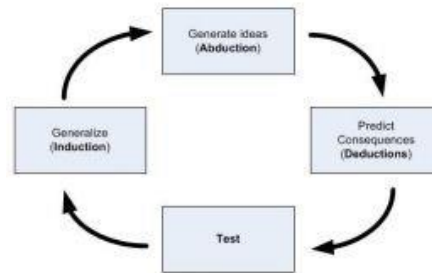


Figura 3: Modelo DT Dunn and Martin. Fuente [6].

El modelo de pensamiento de diseño presentado por Brown [10] detalla cómo ocurre el pensamiento de diseño por medio de tres espacios superpuestos, a saber: Inspiración, Ideación e Implementación. Hay una serie de actividades secundarias en cada espacio, que se describen como un sistema de espacios en lugar de una serie predefinida de pasos ordenados. Estas actividades y cómo se superponen entre espacios se muestran en la Figura 4.

El espacio de "inspiración" motiva a explorar el contexto con empatía y entusiasmo para identificar problemas y oportunidades a través de la observación y comprensión directas; La "ideación" es para generar, desarrollar y probar ideas hacia soluciones, y la "implementación" es el espacio que da cuenta de las soluciones viables para el contexto.

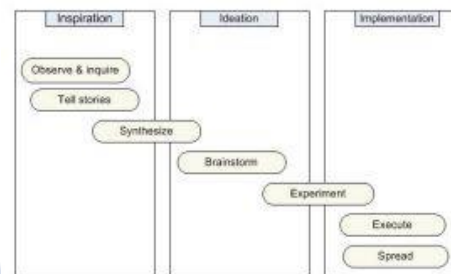


Figura 4: Modelo DT Brown. Fuente [10].



Todos estos tres modelos comparten características comunes como la exploración contextual holística, la generación de ideas basadas en el pensamiento integrador, la creación y evaluación de soluciones viables y la realización de soluciones en el contexto. La naturaleza del proceso de pensamiento de diseño plantea el potencial natural para replantear nuevas soluciones o mejorar soluciones con mejores ideas, de ahí las soluciones de diseño centradas en la innovación para problemas contextuales.

APLICACIÓN HACIA EL DISEÑO DE EXPERIENCIA DE USUARIO

El problema que se busca resolver en la parte de diseño de experiencia de usuario, es que el diseñador no sabe si el usuario encontrará amigable la interfaz propuesta para este, es por eso que *Design Thinking* es una de los métodos más usados con el fin de ponerse en la posición del usuario final y así explorar lo que una interfaz podría hacerle sentir si esta será fácil de usar si esta le brinda información relevante u otros factores que hacen que una interfaz de usuario sea exitosa

Una vez dado el problema la definición de actores principales es el diseñador gráfico ya que este es el responsable de realizar la interfaz según la investigación realizada, para esto se define 5 pasos básicos para llegar a un diseño final que se pueda mostrar al usuario final. En esta sección vamos a aplicar *Design Thinking* a cada uno de los pasos para crear un diseño con el fin de mejorar los resultados al final de crear el diseño, con este enfoque se procederá a explicar cada una de las fases para el diseño de experiencia de usuario ligado a una fase de *Design Thinking*.

- Investigación - Empatizar : La fase de investigación como se explicó, en esta etapa se busca recaudar la mayor cantidad posible de información mediante técnicas tradicionales, *Design Thinking* ofrece sacar mayor cantidad de datos por medio de la empatía es decir ponerse en el lugar del usuario final para juntos conversar y sacar mejor información, ponerse como si el diseñador fuera el usuario final y preguntarse qué es lo que a él le gustaría que exista en la interfaz que haga que sea más fácil y útil al momento de usarlo.
- Organización - Definir e Idear: La fase de organización el diseñador usa criterios técnicos para organizar la información obtenida, aplicando los conceptos de definir e idear se plantea separar la información de forma que cada conjunto de información con relación se vuelva una idea que será tomada en cuenta para las fases siguientes como las de prototipado.
- Prototipado - Prototipar: Como podemos observar esta fase es de prototipado está muy relacionada con la fase de *Design Thinking*, la cual tienen el mismo enfoque de generar un prototipo que el usuario final pueda percibir y dar un *feedback* con el fin de corregir añadir o eliminar ciertas características que al usuario no le causan una buena sensación.



- Pruebas - Testear: La fase de Pruebas también es muy similar debido que los prototipos en UX son interfaces lógicas que están dispuestas al cambio, esta fase de *Design Thinking* se aplica muy bien con la misma meta de buscar una mejora continua.
- Diseño: En esta fase el diseño debe estar completo al haber pasado por una serie de etapas en las cuales se han corregido errores se han añadido características se han eliminado otras que no eran necesario, en esta etapa es el fin del *Design Thinking* por el cual todo el proceso de *Design Thinking* nos lleva a un producto tangible como lo es el diseño final.

CONCLUSIONES

El proceso de generación de interfaces y experiencia de usuario es un proceso subjetivo debido que el diseñador tiene que entender los gustos a su usuario final, si bien el proceso de diseño de experiencia de usuario ya está definido y da buenos resultados la aplicación de técnicas de *Design Thinking* podría mejorar el diseño final y acortar el tiempo de investigación del usuario final, esto aplicando cada una de las fases de *Design Thinking* a su correspondiente fase de diseño de experiencia de usuario, enfocándose en la primera fase de empatizar debido que las demás fases tiene mucha correlación con las fases de DT, en lo cual este trabajo ofrece una metodología para el desarrollo de interfaces y experiencia de usuario enfocado en el usuario final mediante *Design Thinking*.

REFERENCIAS

- [1] J. Gothelf, J. Seiden, and E. Ries, *Lean UX: cómo aplicar los principios Lean a la mejora de la experiencia de usuario*. Universidad Internacional de La Rioja, 2014.
- [2] M. Piasecki and K. Piezka, "Conceptual methodology of developing the user interface," Institute of Applied Informatics, Wroc law University of Technology, 2006.
- [3] G. N. Nedeltcheva and E. Shoikova, "Coupling design thinking, user experience design and agile: towards cooperation framework," in *Proceedings of the International Conference on Big Data and Internet of Thing*, 2017, pp. 225–229.
- [4] R. Villarroel, H. Spencer, and R. Muñoz, "Aplicación de design thinking de manera interdisciplinaria en la asignatura de ingeniería de software," in *Memorias XXX Congreso SOCHEDI*, 2017, pp. 1–9.
- [5] J. Bentley, "Steps to a Good GUI," 14.




- [6] H. Plattner, Guía del proceso creativo. Mini guía: una introducción al Design Thinking+ Bootcamp bootleg. 2018.
- [7] D. Dunne and R. Martin, "Design thinking and how it will change management education: An interview and discussion," *Academy of Management Learning & Education*, vol. 5, no. 4, pp. 512–523, 2006.
- [8] C. Labib, E. Hasanein, and O. Hegazy, "Early development of graphical user interface (GUI) in agile methodologies," *Journal of Computational Methods in Sciences and Engineering*, vol. 9, no. s2, pp. S239–S249, 2009.
- [9] L. Morales, "Structured user interface design methodology," in *CHI'01 extended abstracts on Human factors in computing systems*, 2001, pp. 51–52.
- [10] T. Brown and J. Wyatt, "Design thinking for social innovation," *Development Outreach*, vol. 12, no. 1, pp. 29–43, 2010.
- [11] Y. H. Montero, "Experiencia de usuario: principios y métodos," *Experiencia de Usuario: Principios y Métodos*, vol. 9, 2015.

Componente de revisión de estándar de arquitectura de datos para el gestor de bases de datos SQLite

Data Architecture Standard Review Component for SQLite Database Manager

Sandra Lima Torres

Universidad de las Ciencias Informáticas

 slima@uci.cu

 <https://orcid.org/0000-0003-0025-8725>

RECIBIDO 18/10/2020 • ACEPTADO 10/01/2021 • PUBLICADO 30/03/2021

RESUMEN

La preocupación por la calidad de los sistemas basados en software ha aumentado a medida que estos se integran en cada aspecto de nuestras vidas. Con el incremento del número de usuarios y de los datos almacenados se hace necesario contar con bases de datos más robustas y seguras. Las pruebas de control de calidad han progresado de manera sistemática desde lo manual hasta las aplicaciones automatizadas. La Empresa de Tecnologías de la Información para la Defensa cuenta con el Centro de Calidad, Estándares y Seguridad encargado de certificar y evaluar técnicamente productos informáticos, garantizando la calidad en los mismos. Actualmente el proceso de revisión de las bases de datos en el gestor SQLite es de forma manual, dificultando dicho proceso, donde puede existir un margen de error debido a la cantidad de comprobaciones que se deben revisar. El presente trabajo tiene como objetivo el desarrollo de un componente de revisión de arquitectura de datos para el servidor de gestión SQLite. Para ello se realiza un estudio de sistemas gestores de bases de datos homólogos, las herramientas de pruebas a las bases de datos existentes y cómo está conformado un estándar de arquitectura de bases de datos. Se realizan pruebas funcionales al componente para verificar su correcto funcionamiento. Finalmente se obtiene un componente integrado al marco de trabajo Zeolides, capaz de comprobar automáticamente la arquitectura de las bases de datos del gestor SQLite en las aplicaciones informáticas desarrolladas en la empresa.

Palabras claves: Bases de datos, calidad de software, arquitectura de datos, revisión, SQLite..

ABSTRACT

Concern for the quality of software-based systems has grown as software is integrated into every aspect of our lives. With the increase in the number of users and the data stored, more robust and secure databases are necessary. Quality control testing has progressed systematically from manual to automated applications. The Information Technology for Defense Company has the Center for Quality Standards and Security in charge of certifying and technically evaluating computer products, guaranteeing their quality. Currently the database review process in the SQLite manager is manual, making this process difficult, where there may be a margin of error due to the amount of checks that must be reviewed. The present work aims to develop a data



architecture review component for the SQLite management server. For this, a study of a homologous database management system is carried out, the tools for testing existing databases and how a database architecture standard is formed. Functional tests are carried out on the component to verify its correct operation. Finally, a component integrated o the Zeolides framework is obtained, capable of automatically checking the architecture of the databases of the SQLite manager in the computer applications developed in the company

Keywords: *Databases, software quality, data architecture, revision, SQLite.*

INTRODUCCIÓN

Incrementar o mantener la productividad se deben gestionar eficientemente todos los datos. Son los Sistemas Gestores de Bases de Datos (SGBD) los que permiten almacenar y posteriormente acceder a los datos de forma rápida y estructurada [1].

A lo largo del ciclo de vida del desarrollo de software se toman medidas para asegurar la calidad de los sistemas de bases de datos, uno de los instrumentos que facilita esta tarea es la adopción de estándares de diseño de BD. El uso de los mismos tiene innumerables ventajas, entre ellas está asegurar la legibilidad del modelo de datos, incluso en etapas de análisis y diseño, así como facilitar la tarea de los programadores en el desarrollo de los sistemas [2].

En Cuba existen empresas que se encargan de realizar pruebas de control de calidad. Estas han progresado de manera sistemática desde lo manual (la forma más básica) hasta las aplicaciones automatizadas para los diferentes sistemas informáticos donde se encuentran las BD. La Empresa de Tecnologías de la Información para la Defensa (XETID), posee un Centro de Calidad, Estándares y Seguridad (CCES) encargado de certificar y evaluar técnicamente productos informáticos de producción nacional o importada. Evalúa y certifica procesos de desarrollo y organizaciones para la industria de software en las Fuerzas Armadas Revolucionaria (FAR), según normas nacionales e internacionales [3]. Por la importancia que tiene la calidad para esta empresa, el CCES utiliza estándares y herramientas automatizadas para evaluar sus productos de software. Una de las pruebas de calidad que se realizan en CCES es a las BD de las aplicaciones, donde se evalúa que el estándar de arquitectura de datos que utiliza la XETID sea empleado correctamente. Actualmente existe una herramienta automatizada que realiza este tipo de pruebas a las BD en el gestor PostgreSQL, llamada Componente de revisión de Estándar y Arquitectura de Datos (CEAD) [4].

El auge del sistema operativo Android ha causado gran impacto en el desarrollo de aplicaciones móviles a nivel nacional e internacional. Sobre todo, por la facilidad y comodidad de usar las aplicaciones desde cualquier dispositivo móvil en el momento y lugar que se necesite. Logrando una amplia aceptación en el mercado [5].

La XETID se encarga también de la realización de aplicaciones Android, tales como EnZona o Participación popular. El CCES realiza la revisión y certificación de estos productos, los cuales



cuentan con BD realizadas con el gestor SQLite, ya que se ejecuta en muchas plataformas y sus BD pueden ser fácilmente portadas sin configuración o administración.

La empresa no cuenta con un sistema que compruebe que la arquitectura de datos en el gestor SQLite esté acorde a un estándar de nomenclatura dado, por lo que resulta muy engorroso el proceso de revisión de estas BD. Actualmente, en el CCES, uno de los factores que afecta es el tiempo que se dedica a las pruebas de calidad que se realizan a la arquitectura de las BD de SQLite, pues son de forma manual, donde puede existir un margen de error debido a la cantidad de comprobaciones que se deben realizar.

El presente trabajo tiene como objetivo general: Desarrollar una herramienta que contribuya a la comprobación automática de la arquitectura de las BD gestionadas con SQLite.

En la siguiente sección se exponen las bases de la investigación, se describe el diseño de la misma y un trabajo relacionado con el tema. En la sección de resultados y discusión se analizan los resultados de validar y aplicar la herramienta propuesta.

MÉTODOS Y MATERIALES

Para dar solución al problema identificado se hace necesario establecer un grupo de conceptos referentes a la estandarización de los SGBD, especialmente los desarrollados en SQLite.

Sistema Gestor de Bases de Datos

El SGBD es un conjunto de programas, lenguajes y procedimientos, que permite almacenar, acceder y recuperar los datos en la BD, proporcionándole al usuario herramientas para describir y manipular dichos datos de manera eficiente y práctica. Un SGBD actúa como un intermediario entre la BD y el usuario, ya que opera como interfaz entre estas dos entidades [6].

Según la apreciación de Ramos [7] el objetivo primordial de un gestor es proporcionar eficiencia y seguridad a la hora de extraer o almacenar información en las BD. Los SGBD son aplicaciones de software diseñadas para facilitar tareas. A medida que crecen el volumen de los datos y el número de usuarios (actualmente son habituales los centenares de gigabytes y los millares de usuarios) el apoyo de los SGBD se vuelve indispensable. En la actualidad existen muchos SGBD, los más utilizados son PostgreSQL, MySQL, MongoDB, Oracle, y SQLite [8].

El sistema gestor de Bases de Datos SQLite

González [9] define que "SQLite es un SGBD relacional, famoso por su pequeño tamaño. A diferencia de otros sistemas cliente-servidor el motor de SQLite no es un proceso independiente, lo que hace que la latencia sea menor y el acceso más eficiente. Debido a su facilidad de uso, su pequeño tamaño y su versatilidad SQLite es utilizado en una gran variedad de aplicaciones.



Su uso ha sido muy popular en las aplicaciones para *Smartphone* con sistema operativo Android o iOS [4]. Teniendo en cuenta la bibliografía consultada se puede afirmar que este gestor es mucho más rápido y ligero que MySQL y PostgreSQL. Se ejecuta en muchas plataformas, es de dominio público y por tanto sin costo. Por lo cual, según López [10] las empresas de desarrollo de software en nuestro país tienen un aumento considerable de la producción de aplicaciones para móviles con sistema operativo Android que utilizan este SGBD, como por ejemplo EnZona, Participación Popular, ONAT, Transfer Móvil.

Estándar de arquitectura de BD

Una BD debe ser lo bastante sencilla como para que los interesados puedan comprenderla, además de coherente y estable, por lo que hay que cuidar su diseño y puesta en marcha [11]. Por tanto, se hace necesaria la creación o adopción por parte de la entidad de un estándar de arquitectura de datos [12]. Sánchez [13] hace referencia a que desde la aparición de los primeros SGBD se intentó llegar a un acuerdo para que hubiera una estructura común para todos ellos. Los intentos por conseguir una estandarización han estado promovidos por organismos de todo tipo. Algunos son estatales, otros privados y otros promovidos por los propios usuarios. Entre estos organismos se encuentra la *International Organization for Standardization* (ISO), la cual se encarga de la definición de estándares de gran prestigio internacional.

La arquitectura de datos se diseña y se desarrolla durante la etapa de planificación de un nuevo sistema para establecer la manera en que se procesarán, almacenarán y utilizarán los datos, y cómo se podrá acceder a ellos. En ella se integran los modelos, políticas y reglas que rigen qué datos se van a recopilar; cómo van a ser almacenados, clasificados y explotados mediante la infraestructura tecnológica disponible [12]. Los estándares señalan claramente el comportamiento esperado y deseado en las aplicaciones y son utilizados como guías para evaluar su funcionamiento y lograr el mejoramiento continuo de los servicios [14].

Estándares de arquitectura de BD a aplicar en las pruebas

ISO/IEC 25012

Uno de los estándares más utilizados en el mundo en relación al proceso de desarrollo de BD es la norma ISO/IEC 25012, de ella es preciso conocer que surge de la ISO/IEC 25000 conocida como SQuaRE (*System and Software Quality Requirements and Evaluation*). Es una familia de normas que tiene por objetivo la creación de un marco de trabajo común para evaluar la calidad del producto software, se encuentra compuesta por cinco divisiones [15]:

- ISO/IEC 2500n – División de Gestión de Calidad
- ISO/IEC 2501n – División de Modelo de Calida



- ISO/IEC 2502n – División de Medición de Calidad
- ISO/IEC 2503n – División de Requisitos de Calidad
- ISO/IEC 2504n – División de Evaluación de Calidad

Dentro de estas cinco divisiones se encuentra la ISO/IEC 2501n – División de Modelo de Calidad. Las normas de este apartado presentan modelos de calidad detallados incluyendo características para calidad interna, externa y en uso del producto software. Actualmente esta división se encuentra formada por:

- ISO/IEC 25010 - Sistemas y modelos de calidad de software.
- ISO/IEC 25012 - Modelo de calidad de datos: el cual define un modelo general para la calidad de los datos, aplicable a aquellos datos que se encuentran almacenados de manera estructurada y forman parte de un Sistema de Información.

Estándares de arquitectura de datos para la XETID

El estándar a utilizar en la revisión automática de la arquitectura de BD en el gestor SQLite es Estándar de arquitectura de datos para la XETID en su versión 1.0. Este estándar realizado por el CCES de la empresa XETID en el año 2017, constituye una guía para el desarrollo de las BD. Desde el punto de vista arquitectónico tiene el propósito de brindar una visión de la estrategia de desarrollo del Centro. El propósito de este estándar consiste en establecer los fundamentos para la puesta en práctica del diseño e implementación de las BD.

Entre las pautas que contiene dicho estándar de arquitectura de BD están:

- Tipo de BD
- Base tecnológica para la arquitectura de datos
- SGBD
- Herramienta de diseño
- Herramientas de administración
- Seguridad de datos
- Estándar de nomenclatura
- Tipos de datos
- Políticas de indexado
- Integridad
- Normalización
- Rendimiento
- Preparación de las BD para la replica.



Estos estándares son revisados en su mayoría de forma automática. No abundan a nivel mundial sistemas elaborados que en la búsqueda de alcanzar la perfección en las realizaciones de BD realizan este tipo de pruebas basadas en los anteriores estándares de calidad. Teniendo en cuenta la necesidad de una aplicación que contribuya a la comprobación automática de la arquitectura de las BD creadas en el gestor SQLite, se realiza a continuación una propuesta de una herramienta informática que podría utilizar el CCES.

CEAD PostgreSQL

Sus autores [4] aseguran que la creación del CEAD viabiliza comprobar una correcta arquitectura de datos automáticamente, además de que parte de un estándar conocido, el “Estándar de arquitectura de datos para la XETID” antes descrito.

Este componente, para los probadores de la calidad del Centro, representa una poderosa herramienta que permite optimizar el tiempo de trabajo empleado en la revisión de la arquitectura de las BD. Además, realiza una búsqueda exhaustiva de no conformidades proporcionando un pequeño margen de error.

La herramienta no es adaptable a cualquier estándar, actualmente está desarrollada para la revisión de las BD en PostgreSQL. Por esta razón no es posible su utilización en la verificación de los productos con el gestor SQLite. Si bien es cierto que el código de CEAD se puede modificar incluyendo extensas líneas de código, esta modificación trae consigo una sobrecarga de información a manejar por el probador en una misma interfaz.

Por tanto, afecta el tiempo de familiarización de los trabajadores con la nueva interfaz. Para evitar estos problemas el cliente desea dos componentes separados (CEAD PostgreSQL y CEAD SQLite), ambos componentes iguales en apariencia, pero cada uno con sus funcionalidades específicas. Se desea que el probador utilice el componente que necesite según el tipo de BD que esté revisando en ese momento [4].

De lo anterior se resume que la adherencia de una BD a un estándar de arquitectura de datos en particular es un área en la que se han presentado relativamente pocos trabajos. De cualquier forma, automatizar las pruebas de adherencia a un estándar es una ventaja para el desarrollo y la certificación de calidad de un producto de software.

Al mismo tiempo se evidencia que el sistema analizado no fue creado para realizar comparaciones entre un estándar y la arquitectura de una BD para el gestor SQLite. Por tanto, se procede a la creación de una herramienta para lograr automatizar la revisión de la arquitectura de datos en el gestor SQLite. Sin embargo, el análisis realizado posibilita la identificación de funcionalidades



y tecnologías que contribuye al desarrollo de la propuesta de solución que se propone a continuación.

RESULTADOS Y DISCUSIÓN

Componente de revision de arquitectura de datos para SQLite

La solución se basa en una herramienta de revisión de estándar y arquitectura de datos para SQLite, la cual visualmente estará estructurada en dos áreas de trabajo fundamentales (ver Figura 1).

La primera permitirá adicionar, modificar o eliminar una prueba determinada y listar todas las adicionadas anteriormente.

Además, brindará la opción de darle seguimiento a una prueba determinada, así como ver los detalles de esta en específico y exportar el listado de pruebas como informe.

El sistema también establecerá un lenguaje sencillo y común, que permitirá al usuario la selección de los elementos que desea revisar de la arquitectura (entre indexado, tipo de datos y nomenclatura).

Para ello es necesario que el componente no permita que se realice la revisión hasta que no exista una conexión a una BD.

También facilitará la obtención de incidencias y detalles de una prueba determinada. Una vez realizada la revisión, se podrá utilizar la siguiente área.

La segunda área de trabajo brindará la posibilidad de visualizar los errores encontrados una vez que se haya realizado el reconocimiento de errores a la BD, mostrándose detalles de la misma.

Esta permitirá generar las no conformidades encontradas automáticamente, y ser exportadas en formato Word o PDF.



Componente de revisión de estándar de arquitectura de datos para el gestor de bases de datos SQLite

27

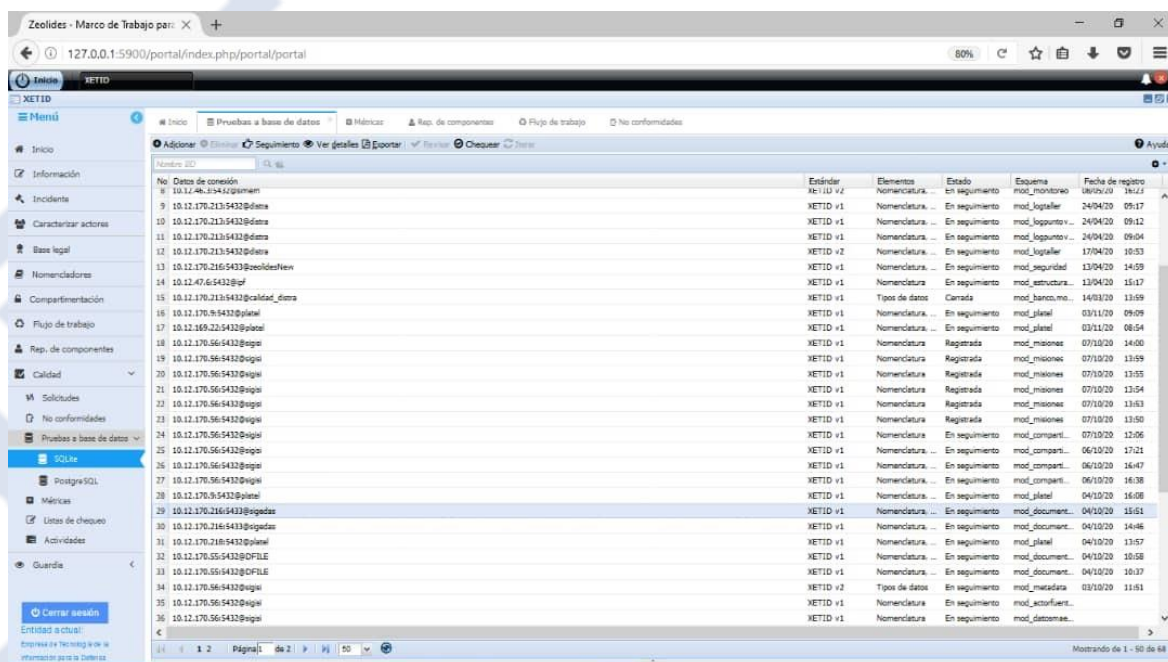


Figura 1. Interfaz de usuario seleccionando la BD a probar.

Una vez seleccionada una BD, como en la imagen anterior, se presiona el botón Revisar, se muestra una ventana emergente solicitando al usuario esperar unos minutos y luego se muestra una nueva ventana en la parte inferior que lista los errores encontrados (ver Figura 2).

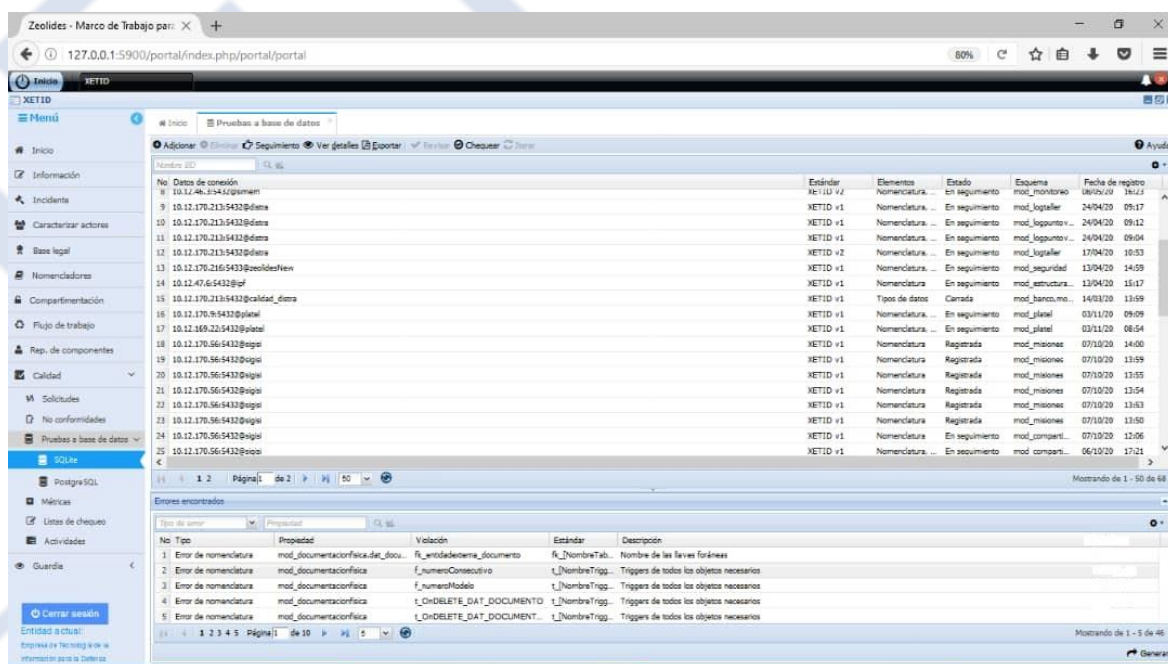


Figura 2. Interfaz de usuario listando errores encontrados.



PRUEBAS REALIZADAS AL SISTEMA

Prueba unitaria

Se realizó la prueba unitaria haciendo uso de la herramienta RIPS; que permite realizar esta automatización del proceso de identificación de potenciales funciones vulnerables en el código fuente de aplicaciones PHP, mediante su análisis estático. En dicha prueba, como se puede observar en la Figura 3, no se encontraron vulnerabilidades.

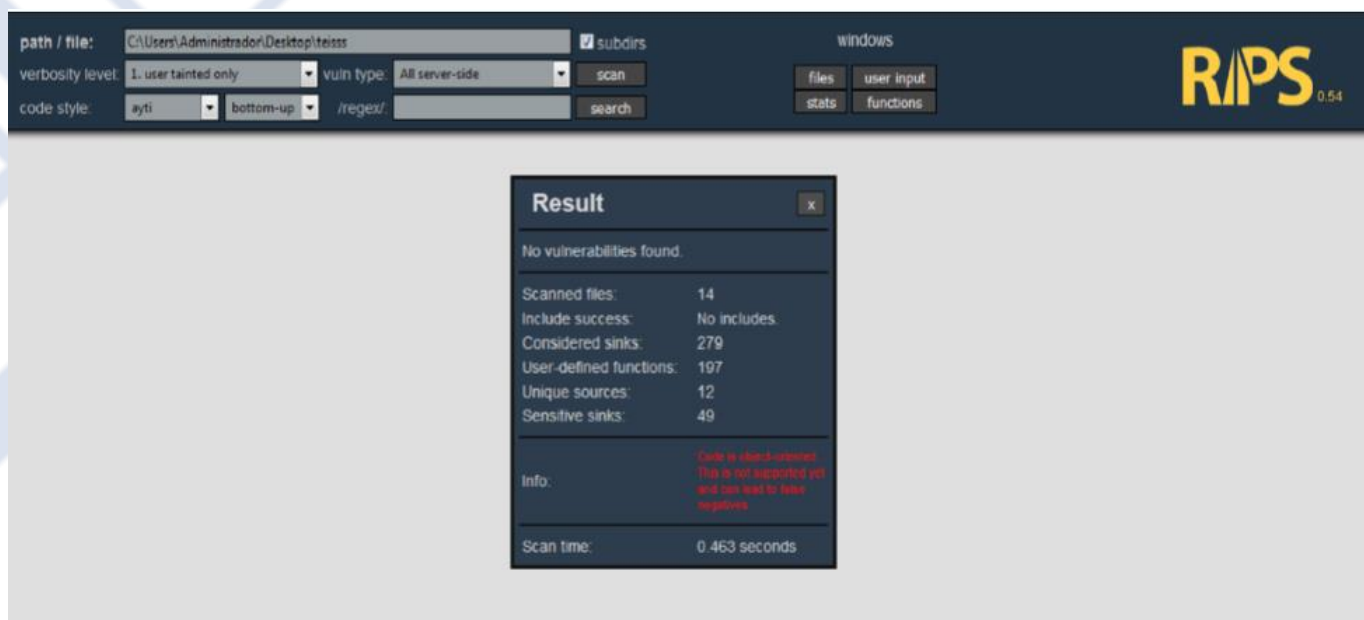


Figura 3. Resultado de la prueba unitaria.

Prueba de seguridad

Los usuarios de negocios quieren la seguridad como un aspecto necesario de la calidad general. Una parte crítica de la creación de este compromiso es comunicarse con ellos acerca de los riesgos y beneficios, y establecer objetivos mutuamente acordados. Las actividades que se pueden realizar para hacer las pruebas de seguridad son diversas y se orientan a varios ámbitos, especialmente en lo relativo a asegurar el funcionamiento y disponibilidad de los servicios web y contenidos publicados. Al realizar la prueba de seguridad a través de la herramienta RIPS no se detectaron violaciones de seguridad (ver Figura 4).

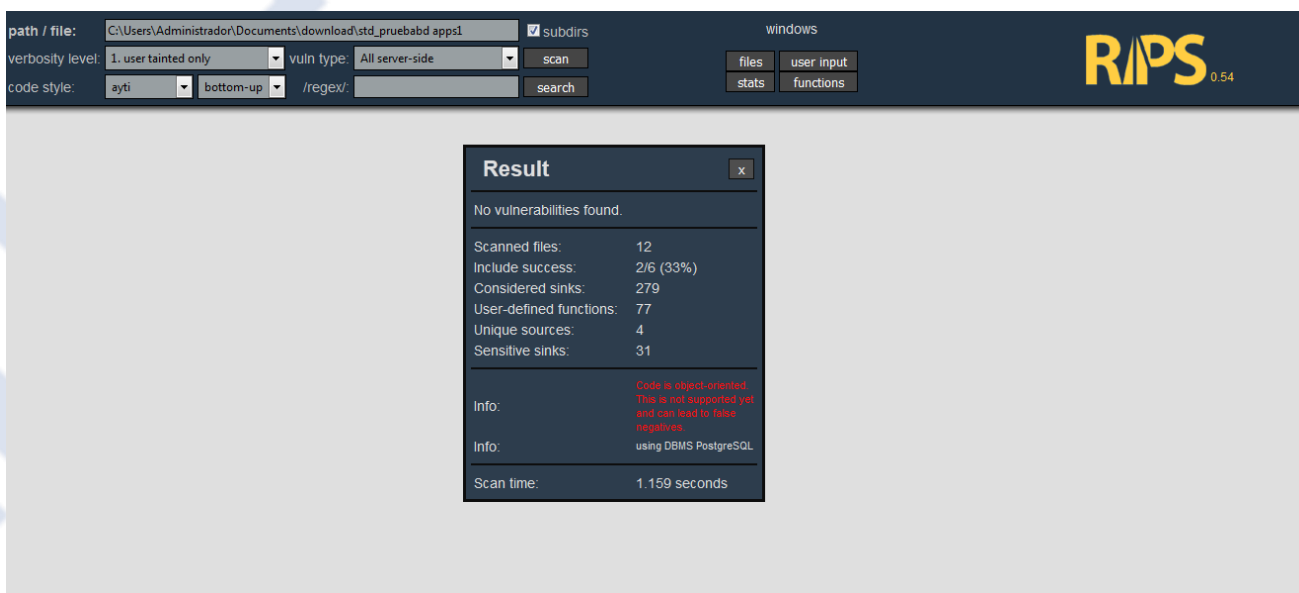


Figura 4. Resultado de la prueba de seguridad.

Se realizaron un total de 3 iteraciones de pruebas que arrojaron como resultados un total de 7 no conformidades, divididas en 2 de ortografía, 2 de redacción, 2 de funcionalidad y 1 de validación para la primera iteración, las cuales quedaron resueltas. En una segunda iteración se identifican 3 nuevas no conformidades siendo 2 de funcionalidad y 1 de validación, las cuales fueron resueltas. En una tercera iteración no se identifican nuevas inconformidades, obteniendo, de esta manera, resultados satisfactorios. La Figura 5 muestra los resultados antes descritos.

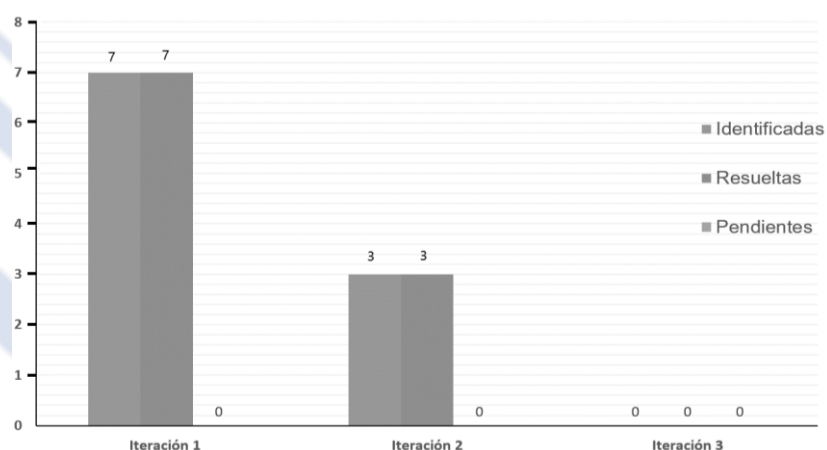


Figura 5. No conformidades por iteraciones.



Con la implementación de este sistema automatizado, el equipo de probadores de BD del CCES disminuye notablemente el tiempo que se destina a probar las aplicaciones que cuentan con SGBD SQLite. Ya que pasaron de una revisión manual, que tardaba días y/o semanas a una automatizada, que toma minutos. Dejando atrás la forma manual de comprobación se gana también en exactitud a la hora de identificar no conformidades, disminuyendo el margen de error. El CCES se beneficia con un programa de elaboración propia con el cual no es necesario pagar licencias ni actualizaciones, de fácil uso y rápido aprendizaje, con el que los probadores tienen cierto nivel de familiarización ya que cuentan con uno similar, el CEAD PostgreSQL.

Teniendo en cuenta las pruebas realizadas, así como otras consideraciones expresadas por el cliente, se puede comprobar que la solución implementada tiene un nivel satisfactorio de aceptación para el centro CCES. Además, se obtuvo un conjunto de recomendaciones y valoraciones que aportan mejoras a la propuesta de solución, en función de ampliar las funcionalidades de la misma. Finalmente se obtuvo como resultado un componente integrado al marco de trabajo Zeolides [16], capaz de comprobar automáticamente la arquitectura de las BD del gestor SQLite en las aplicaciones informáticas desarrolladas en XETID.

CONCLUSIONES

Con el desarrollo del componente de revisión de estándar de arquitectura de datos para el gestor SQLite se obtienen los siguientes resultados relevantes. Con la sistematización del marco teórico de la investigación se identificaron los principales conceptos asociados al componente de revisión de arquitectura de datos del SGBD SQLite y las relaciones entre estos, lo que permitió alcanzar una mayor comprensión de la propuesta de solución. El análisis de sistemas homólogos y de los estándares de arquitectura de BD permitió identificar las tendencias en cuanto al desarrollo de herramientas revisión de estándar de arquitectura de datos e identificar las deficiencias que impiden sea utilizada. La integración de diversas áreas del conocimiento como son la ingeniería y gestión de software, BD, programación, entre otras, permitió el análisis, diseño e implementación del Componente de revisión de estándar de arquitectura de datos para el gestor SQLite. La solución fue validada a partir de la definición de una estrategia de pruebas, que permitió comprobar el correcto funcionamiento del CEAD SQLite a partir de los requerimientos definidos por el cliente. De esta manera se ha cumplido con el objetivo principal de la investigación, siendo el CEAD SQLite una herramienta que cumple con las necesidades del cliente y satisface los requerimientos del problema planteado.



REFERENCIAS

- [1] Y. De La Paz Milán, «Desarrollo de un componente de monitoreo para el Servidor de Gestión PostgreSQL,» Universidad de las Ciencias Informáticas, La Habana, 2011.
- [2] M. Ruiz Gómez, «Creación de una base de datos estandarizada para la flota eólica de Brasil,» 2017.
- [3] XETID, «Proceso de Desarrollo de Software v1.7,» 2017.
- [4] M. Guanche Cañizares, D. Pérez Rojas y J. Reyes Pérez, «CEAD Componente de revisión de arquitectura de datos de la Empresa de Tecnología de la Información para la Defensa (XETID),» 2017
- [5] O. Tatés Pérez, «Implementación de una aplicación móvil Android para el seguimiento de asistencia de los estudiantes de la CISICI, utilizandio Android Studio,» Ibarra, Ecuador, 2018.
- [6] D. Almonacid Inzunza, «Comparación entre gestores de bases de datos relacionales,» 2016. [En línea]. Available: <http://repositoriodigital.ucsc.cl/handle/25022009/1092>.
- [7] M. Ramos, A. Ramos y F. Montero, «Sistemas Gestores de Bases de Datos,» 2016
- [8] Y R. Marín, «Los gestores de bases de datos (SGBD) más usados,» 2019.
- [9] D. González Márquez, «Creación de una práctica de bases de datos relacionales con SQLite,» 2019. [En línea]. Available: https://repositorioinstitucional.ceu.es/bitstream/10637/10242/1/Creacion_DavidGonzalez_2019.pdf.
- [10] P. López Herrera, «Comparación del desempeño de los Sistemas Gestores de Bases de Datos MySQL y PostgreSQL,» 2016.
- [11] B. Gros, A. Escofet y M. Marimón, «The design patterns as tools to guide the practice of teachers,» *Revista Latinoamericana de Tecnología Educativa*, vol. 15, nº 3, pp. 11-25, 2016.
- [12] Y Cognodata, «Arquitectura de datos: la base de una estrategia diferenciadora,» [En línea]. Available: <https://www.cognodata.com/blog/arquitectura-datos-estrategia-diferenciadora>.
- [13] J. Sánchez Asenjo, «Gestión de Base de Datos. Sistemas gestores de bases de datos,» Administración de Sistemas Informáticos en Red, 2008.



[14] D. Mera Mero y C. Villamarin Zambrano, «Categorías de beneficios de estándares y procedimientos,» 2012.

[15] International Organization for Standardization, «NORMAS ISO 25000,» 2005.

[16] XETID, «Tecnologías y Ciberseguridad. Marco de Trabajo para el Desarrollo de Aplicaciones Web (Zeolides),» 2019.



Clasificador de estrellas de Neutrones con una red neuronal multicapa utilizando R

33

Neutron star classifier with a multilayer neural network using R

Luis Angel Aliaga Marica

Universidad Nacional de San Agustín

@ laliagam@unsa.edu.pe

Edilson Wanser Herrera Villa

Universidad Nacional de San Agustín

@ eherrerav@unsa.edu.pe

José Mejía Huayhua

Universidad Nacional de San Agustín

@ jmejia@unsa.edu.pe

Lizette Quispe Flores

Universidad Nacional de San Agustín

@ lquispeflore@unsa.edu.pe

RECIBIDO 02/11/2020 • ACEPTADO 22/01/2021 • PUBLICADO 30/03/2021

RESUMEN

En este trabajo lo que se realizará es analizar el ejercicio "Clasificador de estrellas de Neutrones" para ello lo primero se expondrá una breve introducción de nuestro ejercicio planteado seguidamente realizaremos los conceptos básicos de un red neuronal ya que es el escogido para la resolución del presente ejercicio, pero este está clasificado por redes neuronales artificiales según la topología red y redes según el método de aprendizaje, donde se ha visto por conveniente realizarlo con la red neuronal multicapa – perceptrón multicapa, después se tendrá la limpieza de datos, transformación de casos, selección de casos, selección de un lenguaje de datos así mismo los paquetes, librerías framework que se utilizará, para seguidamente realizar la ejecución de la técnica de entrenamiento, modelo entrenado fase de comprobación , análisis de los resultados y análisis del cliente; finalmente llegar a las conclusiones

Palabras claves: Estrellas, multicapa, neuronas, perceptrón, red.

ABSTRACT

In this work what will be done is to analyze the exercise "Neutron star classifier" for this, the first thing will be presented a brief introduction of our proposed exercise, then we will carry out the basic concepts of a neural network since it is the one chosen for the resolution of this exercise, but this is classified by artificial neural networks according to the network topology and networks according to the learning method, where it has been seen to be convenient to do it with the multilayer neural network - multilayer perceptron, then you will have the data cleaning, transformation of cases, selection of cases, selection of a data language as well as the packages,



framework libraries that will be used, to then carry out the execution of the training technique, trained model, verification phase, analysis of the results and analysis of the client; finally come to conclusions.

Keywords: *Multilayer, neurons, network, perceptron, stars.*

INTRODUCCIÓN

Un púlsar del acrónimo en inglés de *pulsating star* (estrella que emite radiación muy intensa a intervalos cortos y regulares) es una estrella de neutrones que emite radiación periódica. Los púlsares poseen un intenso campo magnético que induce la emisión de estos pulsos de radiación electromagnética a intervalos regulares relacionados con el periodo de rotación del objeto. Las estrellas de neutrones pueden girar sobre sí mismas hasta varios cientos de veces por segundo; un punto de su superficie puede estar moviéndose a velocidades de hasta 70 000 km/s. Se pretende realizar un software que incluirá en el próximo telescopio que lance al espacio con el objetivo de identificar a las estrellas de neutrones de este tipo. Para esto cuenta con una base de datos histórica de 17 898 observaciones de estrellas en las que se clasifican en pulsares o no. En dicha base de datos se tiene como atributos de cada observación:

- Promedio de la radiación emitida.
- Desviación estándar de la radiación emitida.
- Índice de expansión de su masa.
- Índice del aumento de la radiación emitida.
- Promedio de la velocidad de rotación.
- Desviación estándar de la velocidad de rotación.
- Índice de ruido que posee la emisión de radio frecuencia.
- Fuerza centrífuga generada durante la rotación.

MATERIALES Y MÉTODOS Ó METODOLOGÍA COMPUTACIONAL

Como lo menciona [1], las redes neuronales artificiales han recibido un interés particular como una tecnología para minería de datos, puesto que ofrece los medios para modelar de manera efectiva y eficiente problemas grandes y complejos. Son modelos computacionales inspirados en las neuronas biológicas, y que están conformadas por un conjunto de unidades de cómputo básico



(neuronas) las cuales están conectadas entre ellas de múltiples maneras. Estas conexiones estarán definidas por unos pesos los cuales determinarán la fuerza o importancia de dichas conexiones, y durante el proceso de aprendizaje o entrenamiento de la red, serán estos pesos los que se ajustarán con el fin de producir la salida adecuada según la entrada que se aplique a la red.

Estructura de los modelos de redes neuronales: Las unidades de procesamiento se organizan en capas. Hay tres partes normalmente en una red neuronal: una capa de entrada, con unidades que representan los campos de entrada; una o varias capas ocultas; y una capa de salida, con una unidad o unidades que representa el campo o los campos de destino. Las unidades se conectan con fuerzas de conexión variables (o ponderaciones). Los datos de entrada se presentan en la primera capa, y los valores se propagan desde cada neurona hasta cada neurona de la capa siguiente. al final, se envía un resultado desde la capa de salida [2].

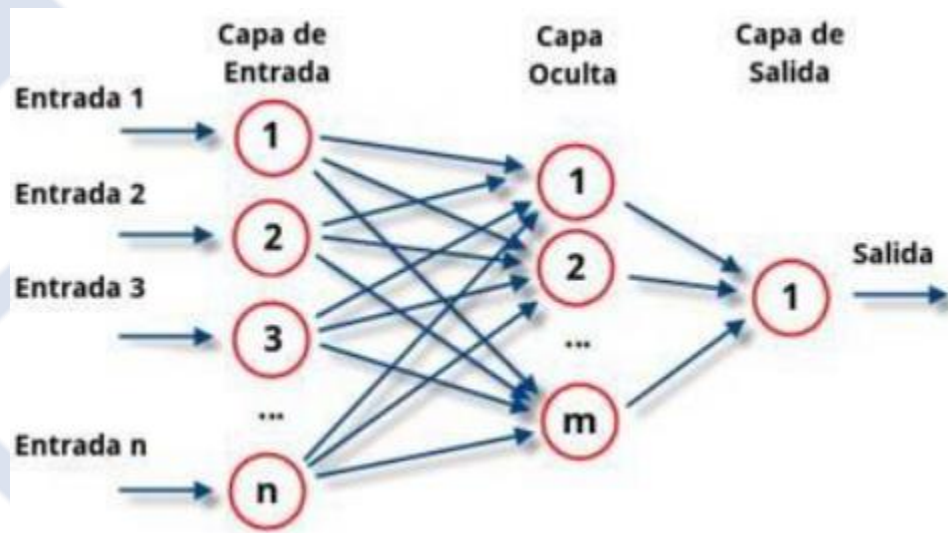


Figura 1: Estructura de una red neuronal.

Clasificación de redes neuronales artificiales según la topología red: Existen varios como lo menciona [3] en este caso solo se nombrarán a algunos: Red neuronal Monocapa-Perceptrón simple:

- Esto corresponde a una red neuronal más simple, está compuesta por una capa de neuronas que proyectan las entradas a una capa de neuronas de salida donde se realizan los diferentes cálculos.

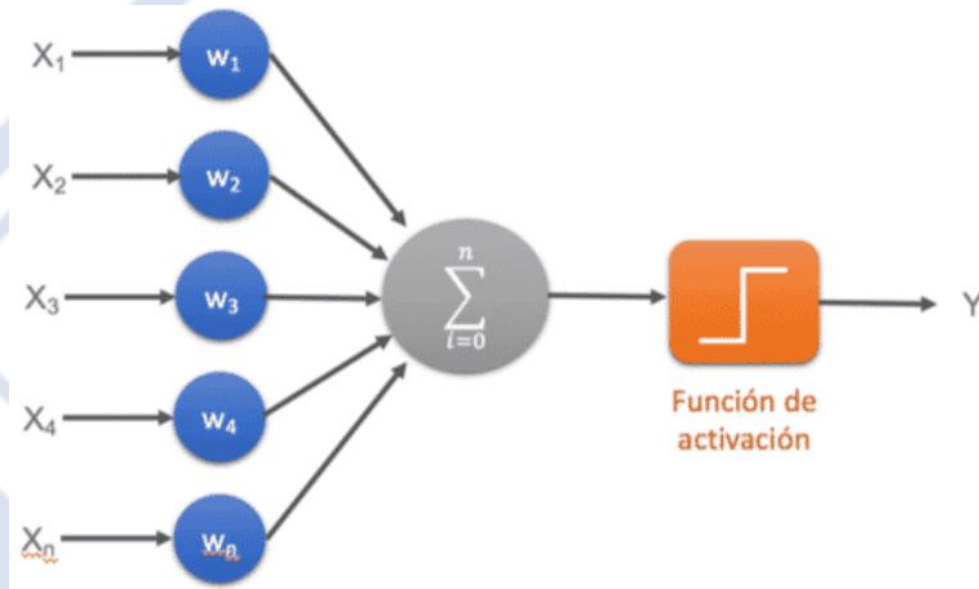


Figura 2: Gráfica de una red neuronal Monocapa-Perceptrón simple.

- Red neuronal Multicapa- Perceptrón multicapa: Es una generalización de la red neuronal monocapa, la diferencia reside en que mientras la red neuronal monocapa está compuesta por una capa de neuronas de entrada y una capa de neuronas de salida, esta dispone de un conjunto de capas intermedias (capas ocultas) entre la capa de entrada y la de salida. Dependiendo del número de conexiones que presente la red esta puede estar total o parcialmente conectada.

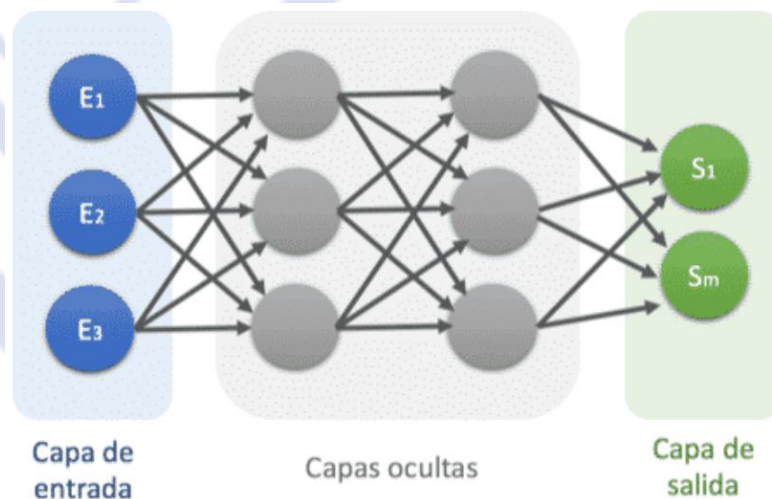


Figura 3: Gráfica de red neuronal multicapa.



- Red neuronal recurrente: No tienen una estructura de capas, sino que permiten conexiones arbitrarias entre las neuronas, incluso pudiendo crear ciclos, con esto se consigue crear la temporalidad, permitiendo que la red tenga memoria.

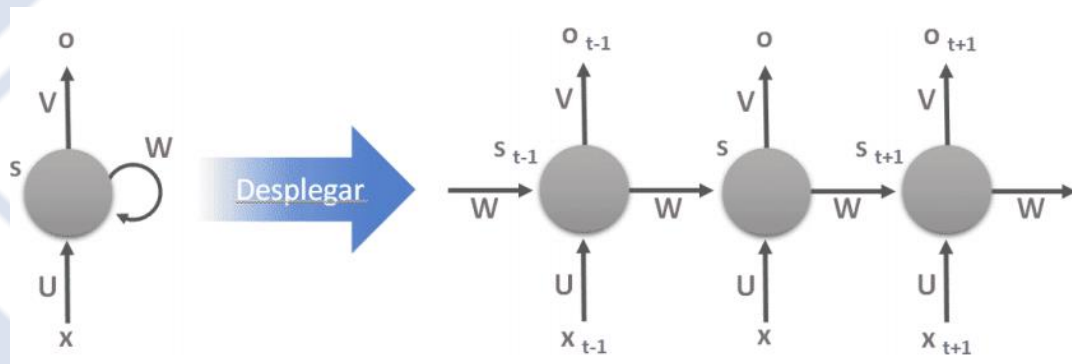


Figura 4: Grafica de red neuronal recurrente.

Clasificación de redes según el método de aprendizaje: Existen varios como lo menciona [4] pero en este caso solo se mencionarán algunos:

- Aprendizaje por corrección de error: Consiste en ajustar los pesos de las conexiones de la red en función de la diferencia entre los valores deseados y los obtenidos en la salida. Ejemplos de algoritmos:
 - Perceptrón.
 - Delta o Mínimo error cuadrado (LMS Error: Least Mean Squared).
 - Backpropagation o Programación hacia atrás (LMS multicapa).
- Aprendizaje estocástico: Consiste básicamente en realizar cambios aleatorios en los valores de los pesos y evaluar su efecto a partir del objetivo deseado y de distribuciones de probabilidad. Una red que utiliza este tipo de aprendizaje es la red Boltzman Machine, ideada por Hinton, Ackley y Sejnowski en 1984 y la red Cauchy Machine desarrollada por Szu en 1986 [4].

RESULTADOS Y DISCUSIÓN

Se utilizó como lenguaje de programación R a través del entorno de desarrollo como RStudio [5] o RStudioCloud [6]. Lo primero que se realizara es cargar el dataset con el siguiente comando:



```
#Forma1
dataset <- read.csv("pulsar_stars.csv")
#Forma2
data = read.csv("dataset.csv", fileEncoding = "Latin1", check.names = F)
```

Figura 5: Código para cargar el dataset.

El proceso de limpieza de datos permite identificar datos incompletos, incorrectos, inexactos, no pertinentes, etc. y luego substituir, modificar o eliminar estos datos sucios. Después de haber realizado el ingreso de datos se prosigue a analizar que las variables de entrada influyen en el resultado final, las imágenes analizadas son:

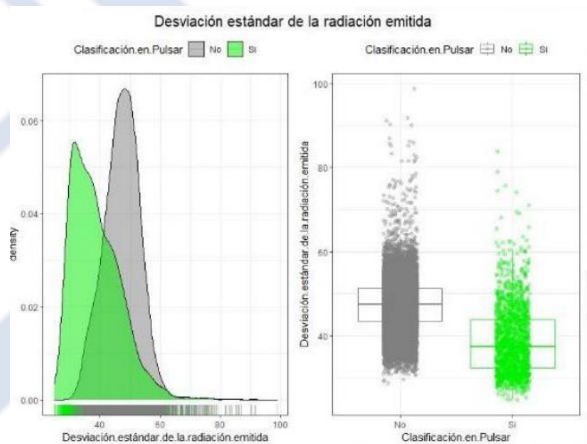


Figura 6: Gráfica de desviación estándar de la radiación emitida.

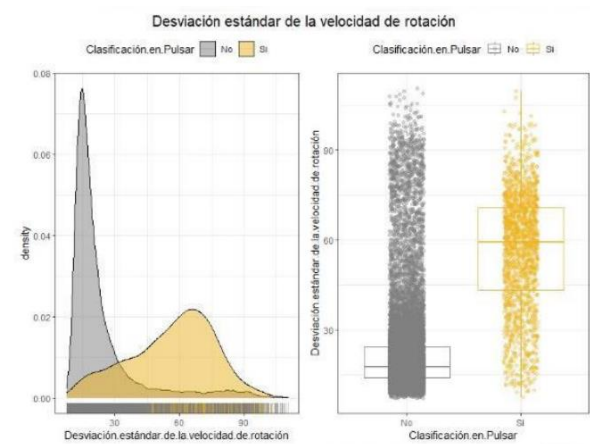


Figura 7: Gráfica de desviación estándar de la velocidad de rotación.

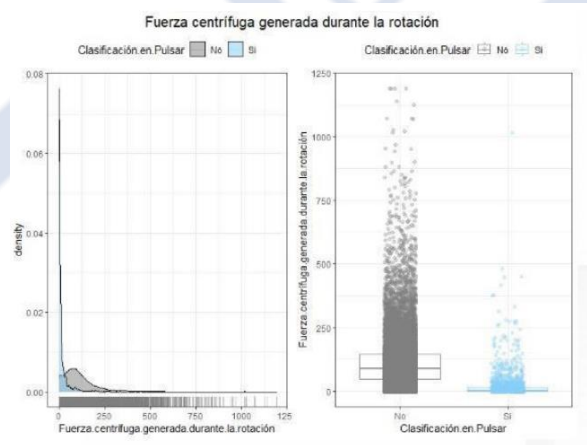


Figura 8: Gráfica de fuerza centrífuga generada durante la rotación.

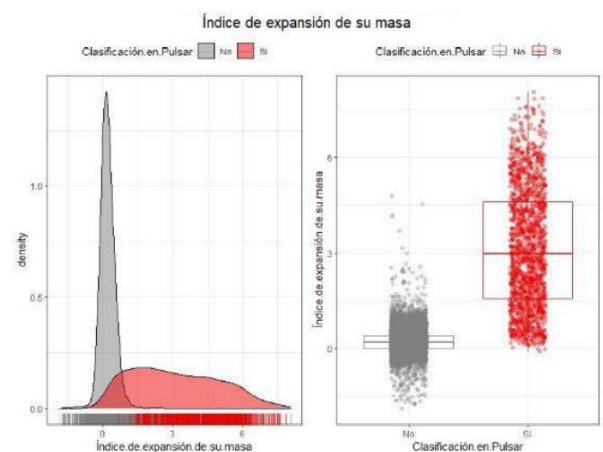


Figura 9: Gráfica de índice de expansión de la masa.

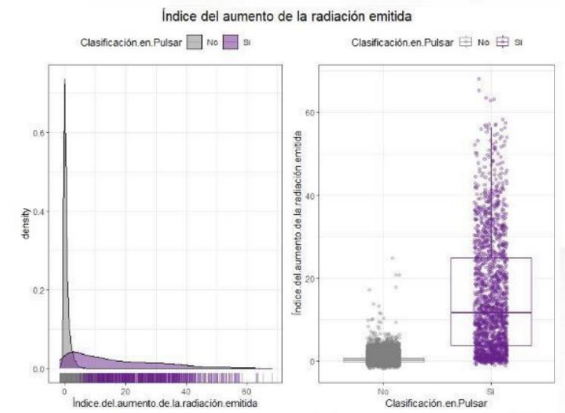
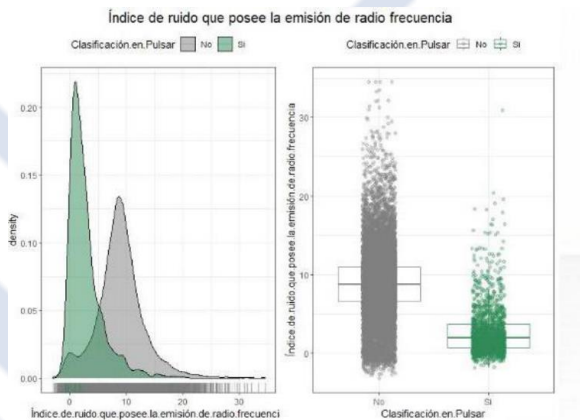


Figura 10: Gráfica de índice de ruido que posee la emisión de radio frecuencia. Figura 11: Gráfica de índice de aumento de la radiación emitida.

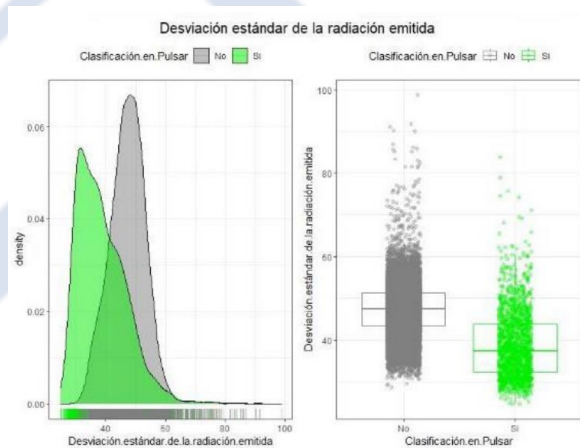


Figura 12: Gráfica de promedio de la velocidad de rotación.

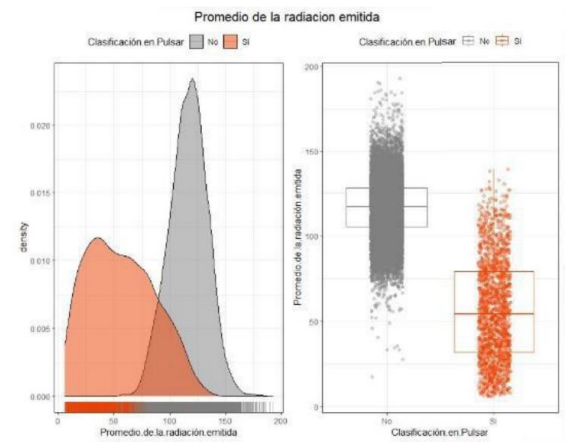


Figura 13: Gráfica de promedio de la radiación emitida.

No se cuenta con ninguna columna vacía ni fila vacía, tampoco se tiene que sustituir ninguna columna ya que todas son influyentes en el resultado final, para ello se comprueba que todas las columnas están totalmente llenas. A continuación, se muestran los datos estadísticos de cada una de nuestras variables de entrada. Una vez preparados los datos se procedió a utilizar la red neuronal artificial, para esto fue necesario descargar el paquete "h2o" [7], el cual es una librería de análisis predictivo y *machine learning* que incorpora la funcionalidad para crear RNA así como modelos de *deep learning*. Ahora podemos hacer uso de nuestro objeto clasificador mediante la función `h2o.deeplearning`.



```
> summary(data)
Promedio.de.la.radiación.emitada Desviación.estándar.de.la.radiación.emitada Índice.de.expansión.de.su.masa
Min. : 5.812 Min. :24.77 Min. : -1.8760
1st Qu.:100.930 1st Qu.:42.38 1st Qu.: 0.0271
Median :115.078 Median :46.95 Median : 0.2232
Mean :111.080 Mean :46.55 Mean : 0.4779
3rd Qu.:127.086 3rd Qu.:51.02 3rd Qu.: 0.4733
Max. :192.617 Max. :98.78 Max. : 8.0695
Índice.del.aumento.de.la.radiación.emitada Promedio.de.la.velocidad.de.rotación
Min. : -1.7919 Min. : 0.2132
1st Qu.: -0.1886 1st Qu.: 1.9231
Median : 0.1987 Median : 2.8018
Mean : 1.7703 Mean : 12.6144
3rd Qu.: 0.9278 3rd Qu.: 5.4643
Max. :68.1016 Max. :223.3921
Desviación.estándar.de.la.velocidad.de.rotación Índice.de.ruido.que.posee.la.emisión.de.radio.frecuencia
Min. : 7.37 Min. : -3.139
1st Qu.: 14.44 1st Qu.: 5.782
Median : 18.46 Median : 8.434
Mean : 26.33 Mean : 8.304
3rd Qu.: 28.43 3rd Qu.:10.703
Max. :110.64 Max. :34.540
Fuerza.centrífuga.generada.durante.la.rotación Clasificación.en.Pulsar
Min. : -1.977 Length:17898
1st Qu.: 34.961 Class :character
Median : 83.065 Mode :character
Mean : 104.858
3rd Qu.: 139.309
Max. :1191.001
```

Figura 14: Gráfica de promedio de la velocidad de rotación.

En esta función el parámetro *activation* hace referencia a la función de activación que utilizará cada una de las neuronas que en este caso es Rectifier (rectificador). Mientras que *hidden* hace referencia a la cantidad de capas ocultas, así como neuronas en cada capa de ellas, en este caso se usaron dos capas ocultas con 7 neuronas en cada una de ellas. Por último, *epoch* hace referencia a la cantidad de veces que se le pasaran los datos de entrenamiento a fin de aplicar el algoritmo de aprendizaje.

```
> classifier = h2o.deeplearning(y = 'TipoEstrella',
+                               training_frame = as.h2o(training_set),
+                               activation = 'Rectifier',
+                               hidden = c(6, 6),
+                               epochs = 500,
+                               train_samples_per_iteration = -2)
```

Figura 15: Creación del modelo de red neuronal.

Una vez finalizado el entrenamiento, podemos realizar predicciones sobre el conjunto de validación de la siguiente manera:



```
> prob_pred <- h2o.predict(classifier, newdata = as.h2o(test_set))  
===== 100%  
===== 100%  
===== 100%
```

Figura 16: Entrenamiento de la red neuronal.

Obteniendo así una matriz que nos indicará cuantos datos han sido predichos de manera correcta, dándonos las respuestas obtenidas vs las respuestas operadas, en este caso se obtuvieron 4388 respuestas correctas y 87 respuestas erradas, teniendo así un modelo que tiene una efectividad del 98.017%.

```
> cm <- table(y_test_set, y_pred)  
> cm  
      y_pred  
y_test_set 0    1  
      0 4022   43  
      1   44  366
```

Figura 17: Resultados de la comprobación del entrenamiento

CONCLUSIONES

Después de desarrollar la investigación y la experimentación de este caso de estudio se llegó a las siguientes conclusiones:

- Las redes neuronales multicapa son muy útiles para problemas de clasificación.
- El lenguaje de programación R cuenta con muchas librerías muy útiles para el procesamiento de datos.
- Todas las variables de entrada influyen en la decisión de clasificar una estrella en pulsar o no, por ende, no pueden ser eliminadas.
- La efectividad obtenida con una red neural con una sola neurona en la capa oculta es mejor que con dos neuronas.
- El error menor lo presenta la red neuronal con una sola neurona en la capa oculta.
- La red neuronal falla más con al momento de clasificar a las que si son pulsares.



REFERENCIAS

- [1] C. C. Aggarwal, "Neural networks and deep learning," Springer, vol. 10, pp. 978–3, 2018.
- [2] J. Schmidhuber, "Deep learning in neural networks: An overview," Neural networks, vol. 61, pp. 85–117, 2015.
- [3] V. Kreinovich, "From traditional neural networks to deep learning: towards mathematical foundations of empirical successes," in Recent Developments and the New Direction in Soft-Computing Foundations and Applications, Springer, 2021, pp. 387–397.
- [4] S. R. Young, D. C. Rose, T. P. Karnowski, S.-H. Lim, and R. M. Patton, "Optimizing deep learning hyper-parameters through an evolutionary algorithm," in Proceedings of the Workshop on Machine Learning in High-Performance Computing Environments, 2015, pp. 1–5.
- [5] J. Allaire, "RStudio: integrated development environment for R," Boston, MA, vol. 770, p. 394, 2012.
- [6] J. M. Elias, "Webinar sobre la docencia en línea con RStudio Cloud," IDP: revista d'Internet, dret i política, no. 31, 2020.
- [7] S. Aiello, E. Eckstrand, A. Fu, M. Landry, and P. Aboyoun, "Machine Learning with R and H2O," H2O booklet, vol. 550, 2016.



Aplicación del método Design Thinking en el área de requerimientos de software

Application of the Design Thinking method in the area of software requirements

Gerson Italo Huarcaya Zapana

Universidad Nacional de San Agustín

@ ghuarcayaz@unsa.edu.pe

Nicolas Herencia Castro

Universidad Nacional de San Agustín

@ nherencia@unsa.edu.pe

Miguel Angel Sarmiento Tico

Universidad Nacional de San Agustín

@ msarmientoti@unsa.edu.pe

Elber Chalco Choquehuanca

Universidad Nacional de San Agustín

@ echalcoc@unsa.edu.pe

Alex Daniel Ticona Bejarano

Universidad Nacional de San Agustín

@ aticonabe@unsa.edu.pe

RECIBIDO 28/11/2020 • ACEPTADO 20/02/2021 • PUBLICADO 30/03/2021

RESUMEN

El presente artículo presenta la aplicación de Design Thinking en el área de requerimientos de software para ayudar y solucionar algunos problemas que existen en el desarrollo de software. Se pretende integrar las herramientas, características y fases de Design Thinking dentro de las actividades del análisis de requerimientos de software. El principal objetivo es integrar el enfoque de Design Thinking de manera que ayude en la definición de los requerimientos de software y finalmente ver y analizar los resultados obtenidos.

Palabras claves: Design Thinking, Requerimientos de software.

ABSTRACT

This article presents the application of Design Thinking in the area of software requirements to help and solve some problems that exist in software development. It is intended to integrate the tools, features and phases of Design Thinking into the activities of the analysis of software requirements. The main objective is to integrate the Design Thinking approach in a way that helps in the software requirements and finally see and analyze the results obtained .

Keywords: Design Thinking, Software requirements.



INTRODUCCIÓN

En los diferentes proyectos de desarrollo de software, es común que surjan situaciones que comprometan el éxito o la continuación de los proyectos debido a diferentes factores. Sin embargo, ¿cómo explicamos la alta incidencia de fallos en los proyectos de software? ¿Por qué existen tantos proyectos de software víctimas de retrasos, presupuestos sobregirados y con problemas de calidad? ¿Cómo podemos tener una producción o una economía de calidad, cuando nuestras actividades diarias dependen de la calidad del sistema? Es responsabilidad de los jefes de proyecto anticiparse a estas situaciones a fin de reducir la probabilidad de ocurrencia, y minimizar el impacto en sus proyectos. Tal vez suene ilógico, pero a pesar de los avances que ha dado la tecnología, aún existen procesos de producción informales, parciales y en algunos casos no confiables.

Los Requerimientos Software cumple un papel primordial en el proceso de producción de software, ya que enfoca un área fundamental: la definición de lo que se desea producir. Su principal tarea consiste en la generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, el comportamiento del sistema; de esta manera, se pretende minimizar los problemas relacionados al desarrollo de sistemas.

En este trabajo, se propone una solución y ayuda a algunos problemas que se enfocan en el área desarrollo de software tal como son los requerimientos de software, diseñando y aplicando un marco de trabajo para desarrollo de software basado en Design Thinking, de tal forma que se puedan integrar las herramientas, características y fases de Design Thinking dentro de las actividades del análisis de requerimientos de software, como marco de trabajo. Finalmente, integrar el marco de trabajo a los requerimientos de software y ver algunos resultados obtenidos.

MARCO TEÓRICO

Ingeniería de Requerimientos

Desde la perspectiva del proceso del software, la ingeniería de requerimientos es una de las acciones importantes de la ingeniería de software que tiende un puente para el diseño y la construcción y proporciona el mecanismo apropiado para entender lo que desea el cliente, analizar las necesidades, evaluar la factibilidad, negociar una solución razonable, especificar la solución sin ambigüedades, validar la especificación y administrar los requerimientos a medida de que se transforman en un sistema funcional [1]. Un requisito del software es una característica que se debe exhibir para solucionar un cierto problema en el del mundo real [2]. Algunos de los componentes de la ingeniería de requerimientos son:

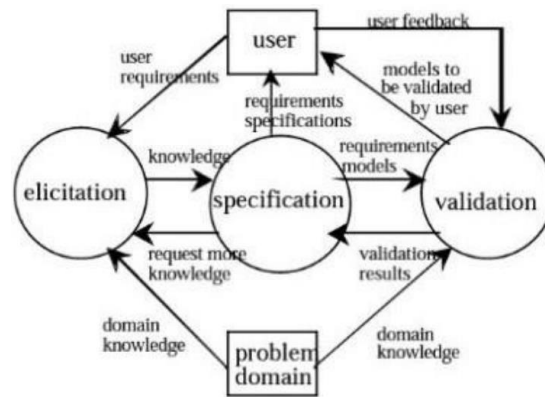


Figura 1. Componentes de la Ingeniería de Requerimientos.

Problemas del Usuario

Introduce los papeles de la gente que participa en el proceso de los requisitos. Este proceso es fundamental interdisciplinario, y el especialista de los requisitos necesita mediar entre el dominio del tenedor de apuestas y el de la tecnología de dotación lógica. Hay mucha gente implicada además del especialista de los requisitos, cada uno de ellos tiene una función en el software. Los tenedores de apuestas variarán según los proyectos, pero incluyen siempre usuarios/operadores y clientes (quienes no necesitan ser iguales) [2]. Al usarlo como método de desarrollo, también parte de la premisa de que lo que se desarrollará debe ser innovador, pero más que eso debe ser: deseable, económicamente viable y técnicamente posible [6].

Elicitación de Requerimientos

Uso de técnicas para conseguir que los stakeholders articulen sus requisitos. Es un área muy difícil en ingenieros de software, es particularmente importante entender que la captura no es una actividad pasiva, y que, ingenieros de software tienen que trabajar difícilmente para sacar la información adecuada. Existe un número de técnicas para hacer esto, como Entrevistas, Escenarios, Prototipos, Reuniones, Observaciones [2].

Especificación de Requerimientos

Se refiere típicamente a la producción de un documento, o a su equivalente electrónico, que puede estar sistemáticamente repasado, evaluado, y aprobado. Para los sistemas complejos, particularmente éstos que implican componentes no-software, se elaboran tres tipos de documentos: definición de sistema, sistema requisitos, y requisitos del software. Para sistemas simples, solamente el tercero de éstos es requerido [2].



Los documentos de los requisitos pueden estar conformes a la validación y procedimientos de verificación. Los requisitos pueden ser validados para asegurarse de que el ingeniero del software entiende los requisitos, y es también importante para verificar que un documento de requisitos se conforma con la compañía de los estándares, y éste es comprensible, constante, y finito [2].

Validación de Requerimientos

Los documentos de los requisitos pueden estar conformes a la validación y procedimientos de verificación. Los requisitos pueden ser validados para asegurarse de que el ingeniero del software entiende los requisitos, y es también importante para verificar que un documento de requisitos se conforma con la compañía de los estándares, y éste es comprensible, constante, y finito [2].

Design Thinking

El estudio de los procesos cognitivos que se manifiestan en la acción de diseñar y como la serie de actividades cognitivas específicas de diseño que los diseñadores aplican durante el proceso de diseñar, pues se centra en el proceso en lugar de en el producto o, dicho de otra manera, se enfoca en la resolución de problemas, pero no comienza con ninguna solución previa [3].

Pasos del Design Thinking

Según [4] los pasos del Design Thinking son:

- **EMPATÍA:** Empatía es la base del proceso de diseño que está centrado en las personas y los usuarios.
- **DEFINIR:** Traer claridad y enfoque al espacio de diseño en que se definen y redefinen los conceptos.
- **IDEAR:** Esta etapa se entrega los conceptos y los recursos para hacer prototipos y crear soluciones innovadoras.
- **PROTOTIPAR:** Es la generación de elementos informativos como dibujos, artefactos y objetos con la intención de responder preguntas que nos acerquen a la solución final. O sea, no necesariamente debe ser un objeto sino cualquier cosa con que se pueda interactuar.
- **EVALUAR:** Consiste en solicitar feedback y opiniones sobre los prototipos que se han creado de los mismos usuarios y colegas además de ser otra oportunidad para ganar empatía por las personas de las cuales estas diseñando de otra manera.



MATERIALES Y MÉTODOS

The Role of Design Thinking and Physical Prototyping in Social Software Engineering.

Este artículo defiende el papel de Design Thinking en Ingeniería de software social y destaca sus implicaciones para la ingeniería de software en general. Lo hace al informar sobre las contribuciones que el pensamiento de diseño, y en particular el diseño físico, ha llevado a la definición del espacio del problema, la captura de requisitos del usuario y el diseño de características del sistema de un sistema de pronóstico de energía renovable [5].

In Two Minds: How Reflections Influence Software Design Thinking

Este artículo define que la calidad del diseño del software depende en gran medida de Design Thinking y los procesos cognitivos de los diseñadores, como se muestra en muchos estudios de diseño.

Uno de estos procesos cognitivos es la capacidad de reflexionar sobre el propio diseño durante un discurso de diseño. Sin embargo, las metodologías de diseño de ingeniería de software han ignorado en gran medida dicho proceso de pensamiento de diseño [6].

IBM Design Thinking Software Development Framework

Este trabajo propone un enfoque diferente centrado en satisfacer las necesidades del usuario final empleando el desarrollo de software iterativo Design Thinking. Esta metodología se aplicó en cinco proyectos reales de desarrollo de software que se analizaron como parte de este trabajo [7].

Designerly thinking: what software methodology can learn from design theory

Este artículo define que Design Thinking es un enfoque que promueve la comprensión de las necesidades del cliente teniendo en cuenta lo que es factible técnica y económicamente. Para evaluar cómo se utiliza el enfoque Design Thinking integrado con las metodologías de desarrollo ágil de software [8].

Problemas al momento de la Indagación de Requisitos

Problemas que se encuentran cuando ocurre la indagación según [1]:

- Problemas de alcance. La frontera de los sistemas está mal definida o los clientes o usuarios finales especifican detalles técnicos innecesarios que confunden, más que clarifican, los objetivos generales del sistema.



- Problemas de entendimiento. Los clientes o usuarios no están completamente seguros de lo que se necesita, comprenden mal las capacidades y limitaciones de su ambiente de computación, no entienden todo el dominio del problema, tienen problemas para comunicar las necesidades al ingeniero de sistemas, omiten información que creen que es “obvia”, especifican requerimientos que están en conflicto con las necesidades de otros clientes o usuarios, o solicitan requerimientos ambiguos o que no pueden someterse a prueba.
- Problemas de volatilidad. Los requerimientos cambian con el tiempo.

El Design Thinking puede fomentar el desarrollo de innovación orientada a la sostenibilidad

El enfoque de Design Thinking para la formulación de problemas parece útil para establecer el alcance de innovación apropiado para Innovación Orientada a la Sostenibilidad. Su fuerte enfoque en los usuarios (por ejemplo, la elección y el uso del producto) y las partes interesadas fomenta el desarrollo de Innovación Orientada a la Sostenibilidad que satisfaga las necesidades reales de los usuarios. Con su enfoque en la experimentación iterativa, Design Thinking permite garantizar efectos positivos de sostenibilidad al tiempo que reduce el riesgo de fracaso de la innovación. En este caso, los efectos positivos de sostenibilidad previstos se neutralizan o incluso se invierten en efectos negativos. Además, nuestro marco conceptual proporciona una base diferenciada que respalda los supuestos de varios investigadores de que un enfoque basado en Design Thinking es un activo esencial, pero a menudo ignorado al abordar los desafíos de sostenibilidad [9].

Visual Design Thinking: A Collaborative Dimensions framework to profile visualisations

Desde un punto de vista teórico, las Dimensiones Colaborativas identificadas impulsan aún más el debate sobre el Design Thinking visual al sistematizar nuestra comprensión de la variedad de características visuales disponibles para los diseñadores.

En particular, los aspectos de las representaciones visuales previamente identificadas en los estudios de diseño, como la capacidad de las visualizaciones para guiar el trabajo del diseñador se agregan en un marco inclusivo. Permite unir el conocimiento académico y la práctica del

diseño, haciendo que los diseñadores y gerentes sean más alfabetizados visualmente y, por lo tanto, más competentes en la selección y uso de visualizaciones para apoyar el proceso de diseño en las organizaciones [10].



RESULTADOS Y DISCUSIÓN

Utilizar la técnica de Design Thinking en el proceso de definición de requerimientos como parte del ciclo de vida de software nos permite identificar las etapas en las cuales dicha técnica es aplicable y cómo su utilización favorece una mejora considerable en la identificación de problemas, identificación de necesidades de usuario, recopilación de nuevas perspectivas de solución, generación de opciones de solución y prototipos que permitan representar dichas ideas [11].

Aquí se muestra, cómo se inicia desde la etapa del diseño del modelo de negocio donde se analiza la viabilidad del proyecto de software, seguida de la identificación de requerimientos donde se seleccionan y definen los requerimientos que van a ser parte de las características funcionales y no funcionales que definen al producto de software, seguidamente dicha técnica es aplicable al análisis y diseño de la propuesta, la implementación y pruebas del producto de software.

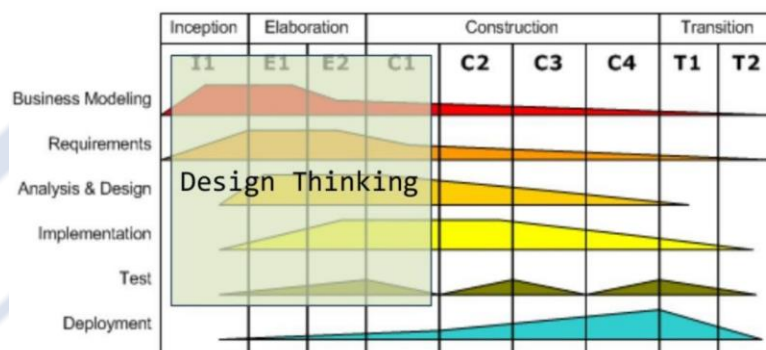


Figura 2. Utilización de Design Thinking dentro del ciclo de vida del software [11].

El siguiente cuadro resume cómo se aplica la técnica de Design Thinking al área de requerimientos de software:

Tabla 1. Aplicación de Design Thinking a la Ingeniería de Requerimientos.

Ingeniería de Requerimientos	DESIGN THINKING		
Elicitación	Empatizar	Observación	Un conjunto de actividades para ayudar a comprender a los usuarios, en el contexto de sus problemas. Observar: A través de la Técnica de Observación para la elicitación de Requerimientos.
		Entrevista	Involúcrate: Generar una conversación e interacción con los Stakeholders a través de Reuniones y Entrevistas, preguntando "¿Por qué?" ya que eso descubre nuevos significados, preguntar una y dos veces si es necesario.



		Reuniones	Mira y Escucha: Lo mejor siempre es combinar estas dos, la conversación y el <i>engagement</i> . Pídele también que te explique como hace algunas cosas y que vaya vocalizando lo que pasa por su mente cuando esté en su trabajo. Ten una conversación mientras trabaja y esté en su contexto.
	Definir	Reuniones	Enmarcar un problema con un enfoque directo, inspirando al equipo, evaluando ideas, capture emocionalmente a las personas que has estudiado y que ayude a resolver el problema imposible de desarrollar conceptos que sirven para todo y para todos.
Elicitación	Prototipar	Prototipo	Para inventar y construir para pensar en resolver el problema. Para cometer errores antes y de manera barata. Para controlar el proceso de la creación de soluciones. Ayuda a identificar distintas variables para poder descomponer grandes problemas que se puedan evaluar y arreglar de mejor forma.
Especificar	Idear	Documento de Definición del Sistema y especificación de Requisitos de Software	Define los requisitos del sistema de alto nivel desde la perspectiva del dominio. La especificación de requisitos software proporciona una base informada para transferir un producto de software a los nuevos usuarios o a las máquinas nuevas.
	Prototipar	Prototipo	Medio para validar la interpretación del ingeniero del software de los requisitos del software, así como para sacar nuevos requisitos. La ventaja de usar prototipos es que pueden hacer más fácil la interpretación del ingeniero del software y, donde lo necesite, dan la explicación útil de por qué son incorrectas
Validar	Testear	Pruebas de Aceptación.	Solicita a los usuarios comentarios sobre el prototipo creado. Las técnicas de evaluación de la experiencia del usuario (UX) podrían usarse para probar el prototipo. Las micro-pruebas son un enfoque común para evaluar prototipos en línea. El tiempo disponible para reclutar usuarios finales, realizar las pruebas y analizar e informar los resultados de las pruebas suele ser muy corto, pero proporciona respuestas rápidas para los equipos de desarrollo.



CONCLUSIONES

El diseño de software es complejo ya que involucra muchas decisiones humanas que por lo mismo no son siempre correctas, por lo que las metodologías de diseño de Ingeniería de Software no han considerado seriamente cómo estos factores de decisión influyen en el razonamiento del diseño resultante además ofrece un análisis inicial y comentarios de los usuarios, ofreciendo una mejor comprensión de qué problemas se deben resolver y cuáles son las mejores soluciones para satisfacer las necesidades del usuario por ende mejora la recopilación y redacción de los requerimientos.

El uso de Design Thinking en el proceso de definición de requerimientos como parte del ciclo de vida de software nos permite identificar las etapas en las cuales dicha técnica es aplicable y cómo su utilización favorece una mejora considerable en la identificación de problemas, identificación de necesidades de usuario, recopilación de nuevas perspectivas de solución, generación de opciones de solución y prototipos que permitan representar dichas ideas.

REFERENCIAS

- [1] R. S. Pressman, Software engineering: a practitioner's approach, Eighth edition. New York, NY: McGraw-Hill Education, 2015.
- [2] P. Bourque and R. Fairley, "Swebok," Nd: IEEE Computer society, 2004.
- [3] R. Pelta Resano, "Design Thinking. Tendencias en la teoría y la metodología del diseño, septiembre 2013," 2018.
- [4] H. Plattner, Guía del proceso creativo. Mini guía: una introducción al Design Thinking+ Bootcamp bootleg. 2018.
- [5] P. Newman, M. A. Ferrario, W. Simm, S. Forshaw, A. Friday, and J. Whittle, "The role of design thinking and physical prototyping in social software engineering," in 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, 2015, vol. 2, pp. 487–496.
- [6] M. Razavian, A. Tang, R. Capilla, and P. Lago, "In two minds: how reflections influence software design thinking," Journal of Software: Evolution and Process, vol. 28, no. 6, pp. 394–426, 2016.
- [7] P. Lucena, A. Braz, A. Chicoria, and L. Tizzei, "IBM design thinking software development framework," in Brazilian Workshop on Agile Methods, 2016, pp. 98–109.



- [8] J. C. Pereira and R. de FSM Russo, "Design thinking integrated in agile software development: A systematic literature review," *Procedia computer science*, vol. 138, pp. 775–782, 2018.
- [9] A. Buhl et al., "Design thinking for sustainability: Why and how design thinking can foster sustainability-oriented innovation development," *Journal of cleaner production*, vol. 231, pp. 1248–1257, 2019.
- [10] S. Bresciani, "Visual design thinking: A collaborative dimensions framework to profile visualisations," *Design Studies*, vol. 63, pp. 92–124, 2019.
- [11] R. Villarroel, H. Spencer, and R. Muñoz, "Aplicación de design thinking de manera interdisciplinaria en la asignatura de ingeniería de software," in *Memorias XXX Congreso SOCHEDI*, 2017, pp. 1–9.



Dispositivo de hardware Libre para Interacción en Entornos 3D

Free Hardware Device for interaction in 3D Environments

53

Jorge Enrique Rodríguez Jiménez

Universidad de las Ciencias Informáticas

@ jerodriguez@uci.cu

id <https://orcid.org/0000-0002-1910-9669>

Ivan Pérez Mallea

Universidad de las Ciencias Informáticas

@ mallea@uci.cu

id <https://orcid.org/0000-0001-8465-1086>

Arianna Rodríguez Jiménez

Universidad de las Ciencias Informáticas

@ arjimenez@uci.cu

id <https://orcid.org/0000-0002-4022-7379>

RECIBIDO 12/12/2020 • ACEPTADO 26/02/2021 • PUBLICADO 30/03/2021

RESUMEN

Los dispositivos de interacción hombre-máquina han sufrido un aumento en su velocidad de evolución. Desde la aparición de los primeros teléfonos inteligentes, hasta la aplicación de la realidad aumentada, y más recientemente, la consolidación de la realidad virtual, el modo en que el ser humano percibe el entorno ha cambiado debido a la influencia de estas nuevas tecnologías. Si bien existe una tendencia hacia el software y hardware libre, promoviendo el desarrollo sin limitaciones y la construcción de dispositivos con un bajo costo económico, Cuba no posee los recursos necesarios para garantizar la adquisición y empleo de dispositivos de realidad virtual, por lo que su desarrollo en cuanto a tecnologías 3D puede verse retrasado con respecto al del resto del mundo. En esta investigación se presentan dos modelos de diseño y construcción de un dispositivo para la simulación de los movimientos e interacción de los miembros superiores e inferiores en entornos tridimensionales (dispositivo de realidad virtual), mediante el uso de tecnologías libres. Se utiliza la plataforma de hardware Arduino como centro de adquisición de datos físicos. Se realiza la selección de sensores y actuadores, así como un análisis de factibilidad y funcionalidad de los mismos.

Palabras claves: Arduino, hardware libre, interacción, realidad virtual, sensors.

ABSTRACT

Human-machine interaction devices have experienced an increase in their speed of evolution. From the appearance of the first smartphones, to the application of augmented reality, and more recently, the consolidation of virtual reality, the way in which humans perceive the environment has changed due to the influence of these new technologies. Although there is a trend towards



free software and hardware, promoting the development without limitations and the construction of devices with a low economic cost, Cuba does not have the necessary resources to guarantee the acquisition and use of virtual reality devices, so its Development in terms of 3D technologies may be lagging behind that of the rest of the world. This research presents two models of design and construction of a device for the simulation of the movements and interaction of the upper and lower limbs in three-dimensional environments (Virtual Reality device), through the use of free technologies. The Arduino hardware platform is used as the physical data acquisition center. The selection of sensors and actuators is carried out, as well as an analysis of their feasibility and functionality.

Keywords: Arduino, free hardware, interaction, sensors, virtual reality..

INTRODUCCIÓN

La sociedad, en crecimiento exponencial y constante cambio, ha variado su forma de pensar a través de los años, apartando los instintos básicos para la supervivencia y tendiendo a un mayor interés en la comodidad y el bienestar físico-espiritual. Con el fin de satisfacer las nuevas necesidades del hombre, la tecnología se ha visto obligada a una acelerada evolución y desarrollo, trayendo consigo el surgimiento de la era de las máquinas. La metáfora de interacción hombre-máquina provista por las ventanas, el ratón, los íconos y el puntero ha sido ampliamente aceptada desde la llegada de los entornos de escritorios gráficos. Esto trajo consigo la estandarización de un conjunto de técnicas y tareas básicas para el uso de este tipo de entornos. Como resultado se desarrollaron interfaces gráficas de usuario maduras y estables, se formalizaron reglas para su manejo y se crearon nuevas funcionalidades como parte de entornos integrados de desarrollo, para potenciar este tipo de vinculación entre el usuario y la computadora. Estos entornos evolucionaron más allá de las dos dimensiones presentando programas que sentaron las bases para el desarrollo de videojuegos, simuladores, entrenadores y otros sistemas de realidad virtual. Se puede definir a la realidad virtual como un sistema informático que genera un conjunto de representaciones de la realidad, las cuales se expresan como simulaciones interactivas mediante el uso de componentes especializados. Tomando como punto de partida lo antes expuesto, es necesario señalar que existen tres elementos básicos que se encuentran presentes en todos los sistemas de realidad virtual [1,2]:

- Simulación interactiva: en un sistema de realidad virtual, cada acción y movimiento del usuario es improvisado por el mismo durante la marcha, sin necesidad de establecer un guion, el usuario decide qué hacer y qué no, afectando de forma directa el entorno y posibilitando experiencias únicas en cada ejecución de la aplicación.



- Interacción implícita: en los sistemas de realidad virtual el equipo de cómputo captura la voluntad del usuario implícita en sus movimientos naturales.
- Inmersión sensorial: se puede definir la inmersión sensorial como la desconexión de los sentidos del mundo real y la conexión al mundo digital. El usuario deja de percibir el entorno que lo rodea y pasa a estar completamente sumergido en el espacio virtual que se ejecuta desde la computadora

A la par de la evolución de los sistemas han tenido que evolucionar los dispositivos de interacción humana, que aún distan mucho de haber llegado a la madurez. Las últimas décadas han marcado un gran avance en este sentido, sin embargo, aún no se ha llegado a un consenso general en la estandarización o forma de realizar las interacciones en las interfaces que conforman estos tipos de software. Aunque la variedad de dispositivos para interactuar con estos entornos ha crecido exponencialmente, siguen siendo caros y difícil de adquirir, razón por la cual se siguen utilizando dispositivos 2D en programas que manejan 3D, limitando así la experiencia de usuario en estas interfaces gráficas. Los dispositivos de realidad virtual son equipos diseñados especialmente para servir como periféricos de entrada y salida. Estos equipos se dividen en tres categorías fundamentales según los sentidos humanos a los cuales están destinados a estimular: los encargados de la visión, la audición y la interacción [3,4].

Existen varias compañías que han desarrollado dispositivos para realidad virtual, cada uno tiene características que los diferencian de los otros, no solo en la estructura, sino también en los requerimientos para su uso y el precio de venta. Algunas de las más requeridas por los usuarios son Oculus (Rift S, Quest y Go), HTC (Vive, Vive Pro y Vive Focus) y Sony (PlayStation VR) [5-8].

Cada uno de estos equipos está constituido por varios componentes como son las gafas, los auriculares y los mandos. Los mandos, gafas y controladores más avanzados, poseen sistemas de seguimiento que les permiten rastrear la posición de los miembros del cuerpo para luego proyectarlo dentro del mundo virtual y permitir la interacción con el mismo por parte de los usuarios [3,4].

Este seguimiento generalmente se realiza mediante el uso de un sistema de localización local, donde solo se tiene en cuenta el movimiento de la parte que se desea proyectar. También es posible calcular la posición de los miembros realizando un reconocimiento a los movimientos individuales que realizan cada una de sus partes [9]. Este último método es mucho más exacto y otorga un mayor control en la adquisición de datos por parte del dispositivo, permitiendo la posible adición de más funcionalidades, aunque para implementar el mismo es necesario conocer la estructura y funcionamiento de cada uno de los miembros del cuerpo humano en su totalidad.



MATERIALES Y MÉTODOS

Se realizó un estudio fisiológico articular de cada uno de los miembros, tanto inferiores como superiores para obtener el conocimiento necesario sobre la estructura y el funcionamiento de los mismos. De ello se obtuvo los miembros superiores están compuestos por cuatro partes [10-13]:

- Cintura escapular: formada por la clavícula y la escápula.
- Brazo: el cual posee un solo hueso llamado húmero.
- Antebrazo: constituido por el radio y el cúbito.
- Mano: estructurada por tres grupos de huesos, el carpo, los metacarpianos y las falanges, todas estas partes unidas por articulaciones.

Los miembros inferiores están compuestos por: la región glútea, el muslo: el cual tiene un solo hueso llamado fémur, la pierna: compuesta por la tibia y el peroné y el pie: compuesto por tres grupos de huesos, el tarso, el metatarso y las falanges, todas las partes unidas mediante articulaciones. El estudio permitió dirigir el enfoque de acción del sistema hacia los movimientos de las partes que se deseaban simular. Para capturar las acciones realizadas por cada uno de los miembros, se exponen las variantes de construcción de un dispositivo de interacción humana de tipo dispositivo de realidad virtual.

Diseño y construcción del Modelo 1

Para lograr una experiencia de inmersión completa es necesario capturar las acciones y movimientos de cada uno de los miembros. Teniendo en cuenta que el cuerpo humano tiene dos miembros superiores y dos inferiores, además de la cabeza, el torso y la cadera, se divide el enfoque de construcción en las partes superior (cabeza, miembros superiores, torso) e inferior (cadera, miembros inferiores) del cuerpo.

Para capturar los movimientos de cada parte, es necesario el empleo de varios sensores [14-16]. Para obtener los valores de rotación se hace uso de 15 sensores BNO055, el mismo es un sensor de tipo IMU (Unidad de Movimiento Inercial) de 9 grados de libertad, contiene un acelerómetro, un magnetómetro y un giroscopio. Se utiliza además 10 sensores Flex, estos son utilizados normalmente para medir el ángulo de giro de articulaciones para aplicaciones robóticas y está diseñado para ser flexionado solo en una dirección. Para un mejor control de los BNO055 se emplean 2 multiplexores I2C TCA9548A, estos permiten manejar hasta 8 dispositivos I2C cada uno, haciendo uso de un solo puerto I2C en nuestro microcontrolador.

La distribución y ubicación de sensores en la parte superior se realiza de la siguiente forma:



- Un sensor BNO055 fijado a la cabeza y otro a la parte superior del torso para controlar los movimientos de los mismos.
- En cada miembro superior tres sensores BNO055, uno fijado en el brazo, uno en el antebrazo y otro en la mano.
- En cada dedo de la mano un sensor Flex para controlar su ángulo de flexión.
- Un multiplexor I2C TCA9548A para controlar los 8 sensores BNO055 de la parte superior

La distribución y ubicación de sensores en la parte inferior se realiza de la siguiente forma:

- Un sensor BNO055 fijado en la cadera para obtener los movimientos realizados por la misma.
- En cada miembro inferior tres sensores BNO055, uno fijado en el muslo, uno en la pierna y otro en el pie.
- Un multiplexor I2C TCA9548A para controlar los 7 sensores BNO055 de la parte inferior.

Tanto los sensores Flex como los multiplexores se conectan a un Arduino Mega 2560, el cual fue seleccionado como centro de adquisición de datos debido a que posee una mayor cantidad de puertos que el clásico Arduino Uno [17-20]. La conexión se realiza de la siguiente forma:

Tabla 1. Conexión de sensores Flex.

Sensor Flex	Arduino
Pin 1	Resistencia de 47K y GND
Pin 1	Pin A (1-10)
Pin 2	5V

Tabla 2. Conexión de Multiplexor I2C TCA9548A.

Multiplexor I2C TCA9548A	Arduino
VIN	5V
GND	GND
SDA	SDA (D 20)
SCL	SCL (D 21)

Tabla 3. Conexión de BNO055.

BNO055	Multiplexor I2C TCA9548A
VIN	5V (Arduino)
GND	GND (Arduino)
SDA	SD (0 - 7)
SCL	SC (0 - 7)



A continuación, se muestra en la Figura 1 el diagrama de conexión para cada componente:

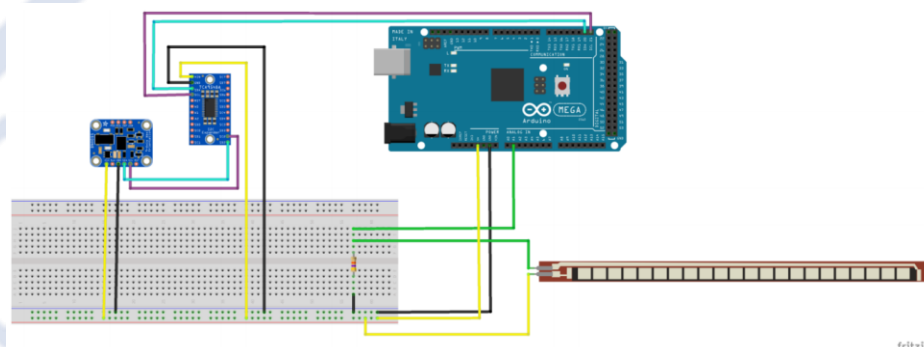


Figura 1. Diagrama de conexión de componentes del Modelo 1.

Con la construcción de este modelo se garantiza una completa proyección de los movimientos del cuerpo al espacio virtual, permitiendo una completa libertad de acción y desplazamiento por parte del usuario. Para lograr este resultado se hace obligatorio el uso de gran cantidad de componentes, por este motivo el costo de producción del dispositivo siguiendo este modelo es alto, a continuación, se presenta la relación del costo de cada componente:

Tabla 4. Análisis de costo del Modelo 1.

Componente	Cantidad	Costo unidad	Costo total
Arduino Mega 2560	1	\$ 14.00	\$14.00
Sensor Flex	10	\$ 13.37	\$ 133.7
BNO055	15	\$ 34.95	\$ 524.25
Multiplexor I2C TCA9548A	2	\$ 7.90	\$ 15.8
Total General			\$ 687.75

Diseño y Construcción del Modelo 2

Con la construcción del Modelo 1 se consigue una simulación total en el espacio virtual de los movimientos realizados por cada una de las partes del cuerpo. Sin embargo, el costo de construcción del mismo se eleva con relación a otros dispositivos que pudieran garantizar las mismas características funcionales.

El Modelo 2 está diseñado para sacrificar algunas de las funcionalidades con el objetivo de reducir el costo de construcción y aumentar la accesibilidad al dispositivo por parte de mayor cantidad de usuarios.

Este modelo se enfoca en la adquisición de los datos de la parte superior del cuerpo, permitiendo la rotación del cuerpo, pero imposibilitando la capacidad de desplazamiento del mismo en el



espacio virtual. Para capturar los datos de rotación se hace uso de 4 sensores BNO055 y de 2 sensor MPU 6050, este último es un sensor de 6 grados de libertad, está compuesto por un acelerómetro y un giroscopio. Para un mejor control de los sensores BNO055 y MPU 6050, se emplean 1 multiplexor I2C TCA9548A. Para capturar las acciones relacionadas con las manos (tomar y soltar objetos) se hace uso de dos pulsadores (botones).

La ubicación y distribución de los sensores se realiza de la siguiente forma:

- Un sensor BNO055 fijado a la cabeza y otro a la parte superior del torso para controlar los movimientos de los mismos.
- En cada miembro superior un sensor MPU 6050 fijado al brazo y un sensor BNO055 fijado al antebrazo.
- Un pulsador fijado a la parte interna de cada mano.
- Un multiplexor I2C TCA9548A para controlar los 4 sensores BNO055 y los 2 sensores MPU 6050.

Al igual que en el Modelo 1, los sensores de la cabeza y el torso capturan los movimientos de los mismos de forma directa e individual. La diferencia está dada en el modo de obtención de los datos de rotación relacionados a los movimientos de los miembros superiores, para esto se realiza un estudio fisiológico articular del miembro superior y se determina que es posible garantizar las principales funcionalidades del mismo haciendo uso de los sensores especificados. Dejando de lado los movimientos de flexión / extensión y desviación radial / cubital de las manos, los grados de rotación del miembro superior se reducen a 5, por lo que tomando los 3 grados de rotación del BNO055 y los 2 del MPU 6050 es posible la obtención de los datos necesarios para la correcta simulación del mismo en el espacio tridimensional. Para las acciones de cada mano se utilizan los estados HIGH del pulsador para mano cerrada y LOW para mano abierta.

La conexión de los componentes se realiza de la siguiente forma:

Tabla 5. Conexión de Multiplexor I2C TCA9548A para el Modelo 2.

Multiplexor I2C TCA9548A	Arduino
VIN	5V
GND	GND
SDA	SDA (D 20)
SCL	SCL (D 21)

Tabla 6. Conexión de sensor BNO055 para Modelo 2.

BNO055	Multiplexor I2C TCA9548A
VIN	5V (Arduino)
GND	GND (Arduino)
SDA	SD (0 - 7)
SCL	SC (0 - 7)



Tabla 7. Conexión de sensor MPU 6050.

MPU 6050	Multiplexor I2C TCA9548A
VCC	3.3V (Arduino)
GND	GND (Arduino)
SDA	SD (0 - 7)
SCL	SC (0 - 7)

A continuación, se muestra en la Figura 2 el diagrama de conexión para cada componente:

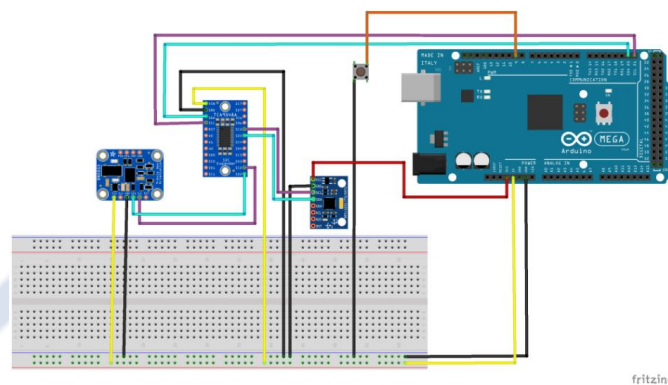


Figura 2. Diagrama de conexión de componentes del Modelo 2.

Con la construcción de este modelo se garantizan las principales funcionalidades de la parte superior del cuerpo permitiendo, aunque no de forma tan completa como el Modelo 1, la inmersión por parte del usuario en el espacio virtual. Debido a que para la construcción de este modelo se emplean pocos componentes, el precio de construcción del dispositivo siguiendo este modelo es bajo, a continuación, se muestra la relación del costo de cada componente:

Tabla 9. Análisis de Costo del Modelo 2.

Componente	Cantidad	Costo Unidad	Costo Total
Arduino Mega 2560	1	\$ 14.00	\$ 14.00
BNO055	4	\$ 34.95	\$ 139.8
MPU 6050	2	\$ 23.90	\$ 47.8
Multiplexor I2C TCA9548A	1	\$ 7.90	\$ 7.90
Pulsador	2	\$ 0.20	\$ 0.40
Total General			\$ 209,9



RESULTADOS Y DISCUSIÓN

Es más económicamente factible la construcción del Modelo 2 debido a que disminuye en gran medida el costo de construcción del dispositivo en relación al Modelo 1, sin afectar a grandes rasgos la eficiencia y funcionalidades necesarias para lograr una correcta interacción por parte del usuario en los entornos tridimensionales.

Como resultado de la investigación realizada para el diseño del Modelo 2, se determinó que los movimientos del miembro superior están dados por la combinación de los movimientos de rotación de cada una de las articulaciones, expresando el vector de la forma: $X_i (\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7)$, donde i es la parte en cuestión, X_i es el movimiento de la misma, y los θ son los valores angulares correspondientes a las articulaciones representadas en la Tabla 1, por lo que pueden ser simulados en una computadora mediante la utilización de sensores capaces de capturar estos ángulos.

Tabla 10. Relación movimiento - articulación del miembro superior

Θ_i	Movimientos del miembro superior	Articulaciones
Θ_1	Rotación Interna / Externa	Hombro
Θ_2	Abducción / Aducción	
Θ_3	Flexión / Extensión	
Θ_4	Flexión / Extensión	Codo
Θ_5	Desviación Radial / Cubital	Muñeca
Θ_6	Pronación / Supinación	Codo
Θ_7	Flexión / Extensión	Muñeca

En la construcción de prototipos de hardware en los que no se requiera un excesivo procesamiento de datos externos, Arduino constituye una opción más acertada que otras plataformas debido a que el mismo es barato y permite la conexión y el control en tiempo real de varios tipos de sensores. Utilizándolo como plataforma de control se obtuvo un dispositivo capaz de capturar los movimientos de los miembros superiores e inferiores del cuerpo mediante el uso de diferentes sensores.

El sistema es capaz de adquirir los datos para cada articulación sin restringir u obstaculizar los movimientos de los miembros, permitiendo la simulación interactiva y la interacción implícita por parte del usuario en el espacio tridimensional, cumpliendo con las pautas objetivas de los dispositivos de realidad virtual.

Aunque los modelos presentados están diseñados para realizar una conexión mediante cable, se puede convertir la comunicación entre el dispositivo y la computadora en inalámbrica mediante



el uso de un módulo de WIFI, Bluetooth o radio frecuencia y realizar la expansión del equipo en relación a la adición de un visor y un sistema de audio para garantizar la inmersión sensorial del usuario y completar la construcción del dispositivo como sistema integral de realidad virtual.

CONCLUSIONES

El estudio fisiológico-articular del miembro superior, demostró que es posible simular en una computadora los principales movimientos y acciones desarrolladas por el mismo, mediante la utilización de sensores especializados en la obtención de ángulos de rotación, permitiendo la optimización en cuanto al uso de componentes para la construcción del dispositivo.

En un proyecto de construcción de un dispositivo de hardware en el cual no se requiera una gran capacidad de procesamiento de datos, la utilización de Arduino constituye una opción más viable que el empleo de otras plataformas similares. La construcción de ambos modelos validó la posibilidad de construir este tipo de dispositivos mediante el uso de tecnologías libres.

REFERENCIAS

- [1] A. L. García, "Realidad virtual," Universidad Complutense de Madrid, Servicio de Publicaciones, 2004.
- [2] M. Kim, C. Jeon and J. Kim, "A study on immersion and presence of a portable hand haptic system for immersive virtual reality," *Sensors*, vol. 17, no. 5, p. 1141, 2017.
- [3] O. Bamodu and X. M. Ye, "Virtual reality and virtual reality system components," *Advanced materials research*, 2013, vol. 765, pp. 1169-1172.
- [4] M. Edwards, "Virtual reality system including smart objects," Google Patents, 2013.
- [5] V. Oculus, LLC, Oculus Gear VR.
- [6] M. Borges, A. Symington, B. Coltin, T. Smith and R. Ventura, "Htc vive: Analysis and accuracy improvement," 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018, pp. 2610-2615.
- [7] M. G. Solid, "Sony Playstation," Tokyo, Japan: Konami Computer Entertainment, Inc, 1998.
- [8] C. Hillmann, "Comparing the Gear VR, Oculus Go, and Oculus Quest," *Unreal for Mobile and Standalone VR*, Springer, 2019, pp. 141-167.
- [9] V. Bharath and R. Patil, "Solid modelling interaction with sensors for virtual reality welding," *MATEC Web of Conferences*, 2018, vol. 144, p. 1008.



- [10] M. Latarjet and L. Testut, "Compendio de anatomía descriptiva," Masson, 1997.
- [11] J. Sobotta, "Atlas de anatomía humana: Cabeza, cuello, miembro superior," Ed. Médica Panamericana, vol. 1, 2006.
- [12] N. A. DFreitas, "Cinematica Articular," Revista de la Sociedad Venezolana de Ciencias Morfológicas, vol. 18, no. 1, pp. 15-20, 2012.
- [13] J. Martín, "Cinética articular del hombro. Revisión de una década de investigaciones," Fisioterapia, vol. 23, pp. 9- 14, 2001.
- [14] U. G. Engine, "Unity game engine-official site," [Online] Available: <http://unity3d.com>, pp. 1534-4320, 2008.
- [15] B. Sensortec, "Intelligent 9-axis absolute orientation sensor," BNO055 datasheet, 2014.
- [16] D. Fedorov, A. Y. Ivoilov, V. Zhmud and V. Trubin, "Using of measuring system MPU6050 for the determination of the angular velocities and linear accelerations," Automatics & Software Enginery, vol. 11, no. 1, pp. 75-80, 2015.
- [17] S. A. Arduino, "Arduino," Arduino LLC, 2015.
- [18] A. Kurniawan, "Introduction to Arduino Boards and Development," en Arduino Programming with. NET and Sketch, Springer, 2017, pp. 1-19.
- [19] D. Lastra Lamarca and others, "Modelo analógico y digital en SystemC-AMS de la placa Arduino Mega 2560," 2015.
- [20] C. Arduino, "Arduino Software," [Online], Available: <https://www.arduino.cc/en/Main/Software>



Proyecto de gestión de redes en BioCen

Network Management Project in BioCen

64

Ihosvany Rodriguez Gonzalez

Centro Nacional de Biopreparados, Cuba

@ ihosvany@biocen.cu

id <https://orcid.org/0000-0003-0212-9556>

Mónica Peña Casanova

Universidad de las Ciencias Informáticas

@ monica@uci.cu

id <https://orcid.org/0000-0003-2500-4510>

Anié Bermudez Peña

Universidad de las Ciencias Informáticas

@ abp@uci.cu

id <https://orcid.org/0000-0002-1387-7472>

Ariel O. Famadas García

Centro Nacional de Biopreparados, Cuba

@ fams@biocen.cu

id <https://orcid.org/0000-0002-4231-5831>

Alicia C. Blanco Domínguez

Centro Nacional de Biopreparados, Cuba

@ alicia@biocen.cu

id <https://orcid.org/0000-0002-1713-1117>

Ludibel Mauri Sedeño

Centro Nacional de Biopreparados, Cuba

@ ludi@biocen.cu

id <https://orcid.org/0000-0001-5899-3859>

Arley Pérez Almario

Centro Nacional de Biopreparados, Cuba

@ arley@biocen.cu

id <https://orcid.org/0000-0003-0588-0396>

RECIBIDO 28/12/2020 • ACEPTADO 05/03/2021 • PUBLICADO 30/03/2021

RESUMEN

BioCen es una institución científica integrada por varias unidades productivas que se dedican a diferentes líneas en el universo biotecnológico y farmacéutico cubano. Para garantizar la seguridad y gestión de la información que se genera en la institución y minimizar el impacto de los riesgos asociados a fallas en la infraestructura, se cuenta con un conjunto de herramientas, procedimientos y estrategias, recogidas en un proyecto de gestión de redes que se actualiza todos los años. El proyecto recoge lo relacionado a la configuración, estructura, control y monitoreo, tanto de la red de datos, la de proceso, como de la wifi; además de la organización de los servicios, servidores, y los sistemas que los conforman. En este trabajo se presentan las herramientas de gestión utilizadas en BioCen para el control y monitoreo de los servicios que se prestan en la red, el tráfico interno y externo de la misma; así como la detección de incidentes. Se muestran ejemplos de monitoreo y control con las herramientas de gestión que se aplican en algunas áreas funcionales, así como el diseño de la red local de BioCen. Finalmente se abordan



las pruebas realizadas al sistema de red, las auditorías internas, el análisis estadístico de los servicios y la respuesta a incidentes.

Palabras claves: Gestión, biotecnología, infraestructuras tecnológicas, monitoreo y control, seguridad y red de datos.

ABSTRACT

BioCen is a scientific institution made up of several production units that are dedicated to different lines in the Cuban biotechnology and pharmaceutical universe. To guarantee the security and management of the information generated in the institution, and to reduce the impact of IT risk, it has a set of tools, procedures and strategies, collected in a network management project that is updated every year. The project includes everything related to the configuration, structure, control and monitoring, both of the data network, the process network, and the Wi-Fi network; in addition to the organization of services, servers, and the systems that comprise them. This paper presents the management tools used in BioCen for the control and monitoring of the services provided on the network, its internal and external traffic; as well as the detection of incidents. Examples of monitoring and control are shown with the management tools that are applied in some functional areas, as well as the design of the local BioCen network. Finally, the tests carried out on the network system, internal audits, statistical analysis of services and response to incidents are addressed.

Keywords: Biotechnology, management, monitoring and control, security and data networks, technological infrastructure.

INTRODUCCIÓN

BioCen es una institución científica integrada por varias unidades productivas que se dedican a diferentes líneas en el universo biotecnológico y farmacéutico cubano: medios de cultivos y bases nutritivas, anti-anémicos, ingredientes farmacéuticos activos y productos parenterales. Además, dispone de otras áreas de servicios, entre ellas las de ingeniería, compras, aseguramiento de la calidad y control de la calidad. Para garantizar la seguridad y gestión de la información que en él se genera se cuenta con un conjunto de herramientas, procedimientos y estrategias que tienen como objetivo, garantizar su integridad, disponibilidad y confidencialidad, recogidas en un proyecto de gestión de redes que se actualiza todos los años, cuyo objetivo central es lograr la máxima disponibilidad de los servicios de la red, con los enfoques de negocio y técnico. Ello se realiza teniendo en cuenta las diferentes funciones al organizar la gestión como son: operación, administración, análisis y planificación. Elaborado a partir de las legislaciones establecidas en



Cuba y cumpliendo las regulaciones internacionales establecidas para la Industria Biofarmacéutica.

El Proyecto recoge lo relacionado a la Red en BioCen, específicamente: la configuración, estructura, control y monitoreo, tanto de la red de datos, la de proceso, como de la wifi; además de la organización de los servicios, servidores y los sistemas que los conforman. Relacionando siempre cada servicio con las áreas funcionales de la gestión de redes y barriendo los componentes de un sistema integrado de gestión de redes y servicios.

El Departamento de Informática y Redes se encuentra subordinado a la dirección general, dando servicios y desarrollando productos para cumplir con los objetivos y metas de la empresa.

MATERIALES Y MÉTODOS

BioCen cuenta con un sistema de gestión como se muestra en la Figura 1, el cual muestra una estructura por niveles y la estrecha relación que tienen para lograr una operación eficiente de la red y sus servicios.

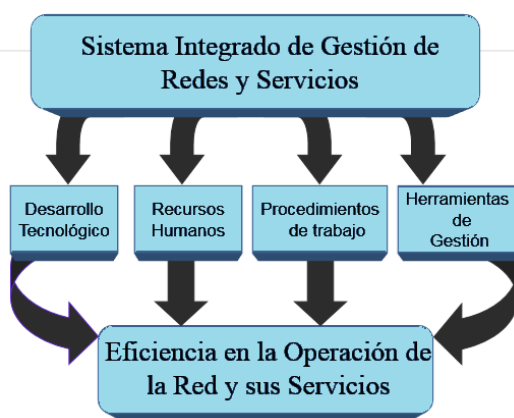


Figura 1. Sistema Integrado de Gestión de Redes y Servicios en BioCen.

En el desarrollo tecnológico se cuenta con la virtualización de varios servidores y *Cloud Computing* para el intercambio de información entre las empresas de BioCubaFarma, tenemos el servicio *VoD*, utilizando *Trueconf* para las videoconferencias, estamos comenzando un proyecto con España para elevar el estándar de BioCen a industria 4.0 y usar *IoT* en las plantas productivas. Además, solicitamos el incremento de ancho de banda de transmisión con el objetivo de mejorar las comunicaciones con cierta frecuencia y tecnología móvil 4G, entre otras.



Respecto a los recursos humanos se cuenta con un personal altamente calificado que siempre está superándose en las disciplinas y competencias que se necesitan para el buen desempeño de sus funciones, en especial los administradores de red y especialista de seguridad informática.

Se cuenta con un manual de procedimientos de las operaciones básicas para la gestión de la red y con un Sistema Certificado de Gestión de la Calidad a nivel de empresa, que rige el seguimiento y modificaciones de los mismos, para su actualización y mejoras.

Herramientas de seguridad en la red de BioCen

BioCen utiliza nueve herramientas de gestión (ver Figura 2) para el control y monitoreo de los servicios que se prestan en la red, el tráfico interno y externo de la misma; así como la detección de incidentes. La caracterización de cada sistema, su empleo y la frecuencia de monitoreo se reflejan en los informes de análisis diarios.



Figura 2. Herramientas de seguridad utilizadas en la red de BioCen.

Los integrantes del grupo de redes son responsables de verificar los accesos de escritura y borrado a los subdirectorios de trabajos y carpetas, a través de programas que se ejecutan de forma automática y que informan de cualquier anomalía a los administradores de las aplicaciones.

Se activan los medios de alarmas y avisos del sistema, que permiten a los operadores o administradores de los mismos, conocer cuándo están ocurriendo hechos relevantes según las reglas declaradas en cada caso.

Con el objetivo de implementar el cumplimiento de la Resolución 126/2019 del Ministerio de Comunicaciones "Medidas de control y los tipos de herramientas de seguridad que se



implementan en las redes privadas de datos” se establece el uso de las siguientes herramientas en la red de computadoras de BioCen:

1. Herramientas que muestran el estado actualizado de los servicios implementados en cada servidor.
 - PRTG Network Monitor (versión 17). Su función es alertar sobre problemas en la red. Permite realizar chequeos intermitentes en los equipos y en los servicios que se especifiquen con el uso de complementos externos, los que devuelven información al sistema informático. Proporciona gran versatilidad para consultar cualquier parámetro de interés de un sistema y genera alertas que se reciben por medio de correos electrónicos y SMS, cuando estos parámetros exceden los márgenes definidos por el administrados de la red. Las alertas son recibidas y procesadas, según se presentes, por el administrador de red [1].
 - ManageEngine AD Manager Plus (versión 6) [2]. Permite a los administradores de redes administrar objetos de Active Directory (AD) fácilmente y general informes instantáneos. Se emplea por los administradores de la red en la gestión de usuarios y grupos del AD, según necesidad. Se aprovecha las facilidades de generación de informe para elaborar estadísticas de los servicios brindados o cualquier otra información solicitada por los administrativos o la especialista de seguridad informática.
 - Kerio Control (versión 9) [3]. Es un firewall de gestión unificada de amenazas que cuenta con la prevención de intrusiones, filtrado de contenido, brinda informes de actividades y permite la gestión del ancho de banda. Es empleado por el administrador de la red en la gestión del servicio de internet diariamente. También es utilizado por la especialista de seguridad informática para la confección de informes estadísticos mensuales de la utilización de este servicio e investigación de incidentes.
2. Herramientas para supervisar la carga y disponibilidad de los servidores.
 - PRTG Network Monitor y ManageEngine AD Audit Plus (versión 6) [4]. Esta última es una herramienta que permite la monitorización de redes, vigilar los equipos y servicios que se especifiquen, así como alertar cuando su comportamiento no es el adecuado. Pueden realizar el monitoreo de los servicios de red (SMTP, POP3, HTTP, SNMP) y de los recursos de sistemas de hardware (carga del procesador, uso de los discos, memoria, estado de los puertos). El monitoreo es realizado semanalmente por el administrado de red.
3. Sistemas de detección y prevención de intrusos (IDS/IPS en inglés).



- Open Source Security Information Management (OSSIM) (versión 5.8.6) [5]. Es capaz de generar análisis de tráfico en tiempo real y registro oficial de eventos (logs) de paquetes en redes IP. Protege los sistemas computacionales de ataques tanto internos como externos, de manera proactiva; con el uso de tecnologías de detección basada en firmas, en políticas, en anomalías o por medio de sensores. Los sistemas IDS/IPS son usados en conjunto. El monitoreo es realizado diariamente por el administrador de red.
4. Herramientas para monitorear el comportamiento del tráfico de la red, análisis de protocolos y detección de anomalías.
- ManageEngine Netflow Analyzer (versión 12.4) [6]. Su función es monitorear el comportamiento del tráfico de la red, brinda gráficos de tráfico multi-enrutadores (MRTG en inglés), que permite monitorear la progresión del tráfico entrante o saliente de las interfaces del enrutador y el estado de la red, a partir del protocolo simple de Administración de red (SNMP). Contiene archivos de imágenes y proporciona una información visual en línea del tráfico de los dispositivos. El monitoreo es realizado diariamente por el administrador de red
5. Herramientas para dar seguimiento a las trazas.
- Kiwi Syslog Server (versión 9.6) [7], ManageEngine EventLog Analyzer (versión 12) [8], Kerio Control y Sawmill Enterprise (versión 8.7). Estas herramientas realizan análisis de logs y generan los reportes asociados a los servicios media, e-mail, registros de seguridad, redes y de aplicaciones. Son herramientas que permiten a los administradores de sistemas informáticos ver de una manera sencilla y amigable qué sitios de Internet se visitan. Generan listados diarios, semanales, mensuales y personalizados con los sitios de Internet que visita cada usuario, cuánto consumió en megabytes, entre otras informaciones. Son empleadas por el administrador de red y la especialista en seguridad informática para monitorear diariamente, elaborar las estadísticas mensuales del uso de estos servicios y realizar la investigación correspondiente ante un incidente de seguridad detectado.
6. Herramientas para la detección de posibles vulnerabilidades en la red.
- Open Source Security Information Management (OSSIM). Es empleada para explotar, administrar y auditar la seguridad de redes. Detecta hosts online, sus puertos abiertos, servicios y aplicaciones que corren en ellos, su sistema operativo, así como qué cortafuegos corren en una red. Realiza un chequeo exhaustivo de potenciales problemas en el servidor, existencia de archivos y aplicación peligrosas. El monitoreo es realizado diariamente por el administrador de red y semanalmente por la especialista en seguridad informática.



7. Herramientas para el control centralizado del inventario de hardware y del software.

- LanSweeper (versión 7.2) [9]. Es empleada para el control remoto por los administradores de redes del inventario de hardware y del software. Permite escanear una dirección IP o una subred con el objetivo de obtener información detallada de equipos no inventariados. Es empleada, según necesidad para el desempeño de sus funciones, por la especialista del grupo de informática que atiende el control del equipamiento, los técnicos de mantenimiento de hardware y software, así como por la especialista en seguridad informática para el monitoreo mensual e investigación de incidentes.

8. Gestión de actualizaciones de seguridad.

- Windows Server Update Services (WSUS, 2019) [10]. La actualización las herramientas de seguridad es muy necesaria para eliminar los problemas de seguridad, lo cual permite mantener la eficiencia operativa y la estabilidad en la infraestructura de los sistemas. La supervisión es realizada semanalmente por el administrador de red.

9. Sistemas de correlación de eventos.

- Open Source Security Information Management (OSSIM). Sistema que combine toda la información de las diferentes herramientas de seguridad para mostrar por medio de la correlación de eventos lo que sucede en la red en tiempo real. Esto le permite al supervisor tomar medidas con carácter proactivo ante un evento o incidente de seguridad. El monitoreo es realizado diariamente por el administrador de red y semanalmente por la especialista en seguridad informática.

El diseño de la red de los centros de datos se basa en un modelo estructurado de tres capas, lo cual permite incrementar la escalabilidad, el rendimiento, la disponibilidad y el mantenimiento de forma transparente y continua (ver Figura 3). Cada una de estas capas requiere una funcionalidad diferente:

- Núcleo: conmutación en capa 3 del tráfico que entra y sale del centro de datos. Brinda conectividad a varios módulos de agregación. Se encarga del enrutamiento entre el centro de datos y la red externa.
- Agregación: brinda un punto de consolidación en el cual los switches de la capa de acceso se interconectan dando extensión a las VLANs entre servidores que estén en diferentes racks. Provee una frontera entre los enlaces capa 3 y los dominios de broadcast capa 2. Se encarga del procesamiento del STP, puertas de enlace redundantes. Brinda servicios como cortafuegos, detección de intrusos, balance de



carga. Es la capa fundamental de un centro de datos.

- Acceso: conformado por los switches que les brindan conectividad a los servidores. Permiten mantener el tráfico de servidor a servidor dentro de una misma VLAN de forma local lo que reduce la necesidad de procesamiento en las capas superiores.

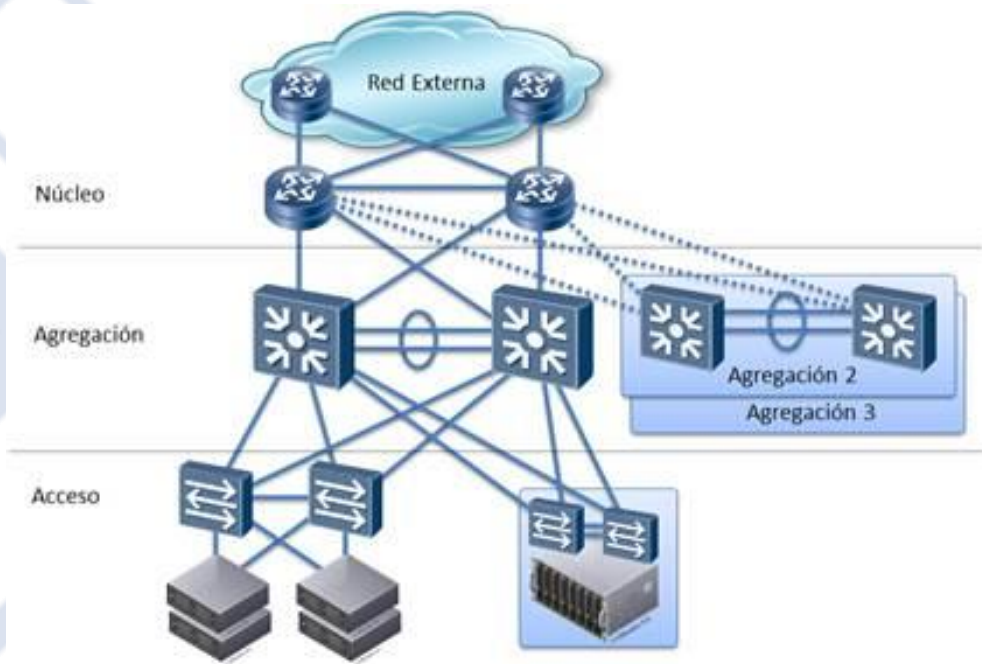


Figura 3. Diseño de la red de los centros de datos basado en tres capas.

Ejemplos de monitoreo y control

Ejemplos de monitoreo y control con tres herramientas de gestión que se aplican en algunas áreas funcionales, para su evaluación y análisis.

Gestión de contabilidad

La Figura 4 muestra una captura de pantalla con la herramienta PRTG sobre la recolección de datos estadísticos y utilización de los recursos del servidor ATT.

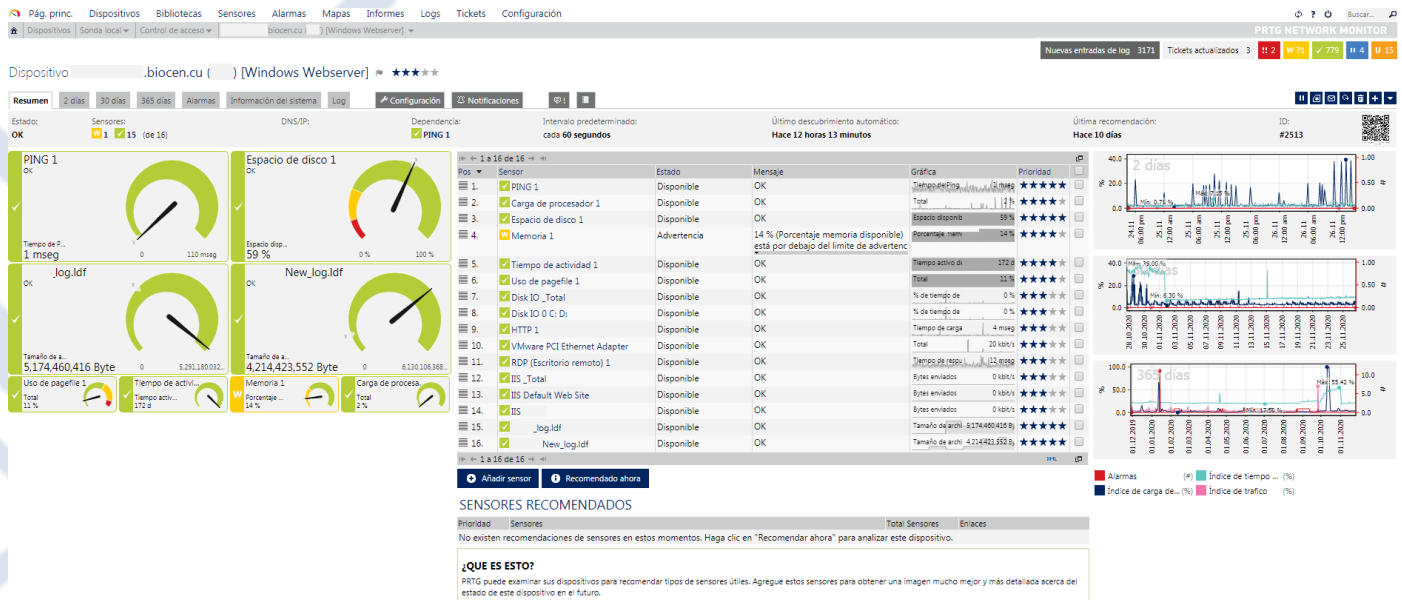


Figura 4. Ejemplo de recolección de datos estadísticos con la herramienta PRTG en BioCen.

La Figura 5 muestra una captura de pantalla con la herramienta PRTG sobre el monitoreo de la disponibilidad de los servicios y recursos de la red.

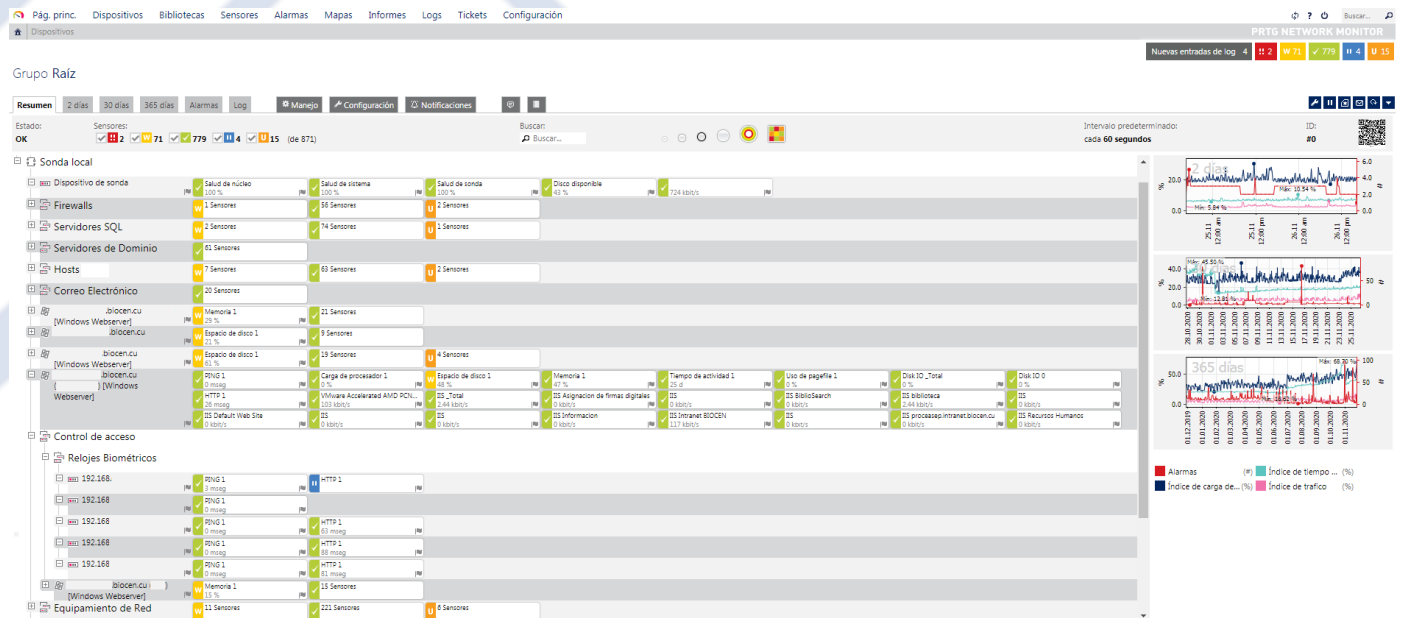


Figura 5. Ejemplo de monitoreo con la herramienta PRTG en BioCen.



La Figura 6 muestra una captura de pantalla con la herramienta Kerio Control, sobre el monitoreo de ancho de banda de Internet por direcciones en la empresa.

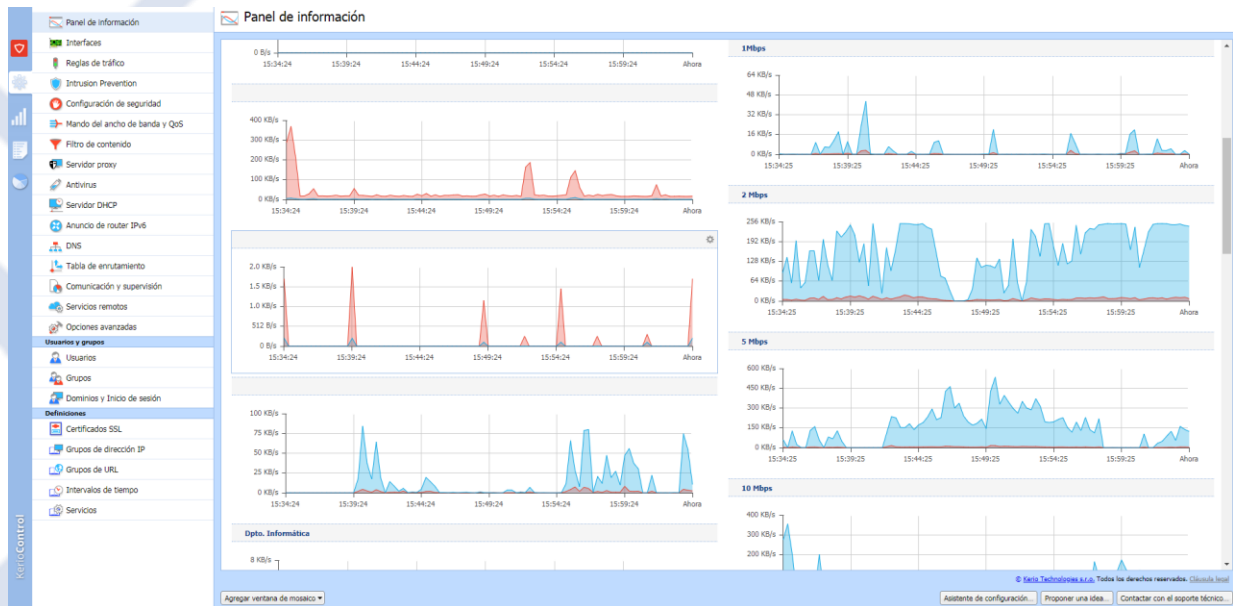


Figura 6. Ejemplo de monitoreo con la herramienta Kerio Control en BioCen.

Gestión de rendimiento

La Figura 7 muestra una captura de pantalla con la herramienta PRTG sobre el ancho de banda del canal de la VPN de BioCubaFarma.

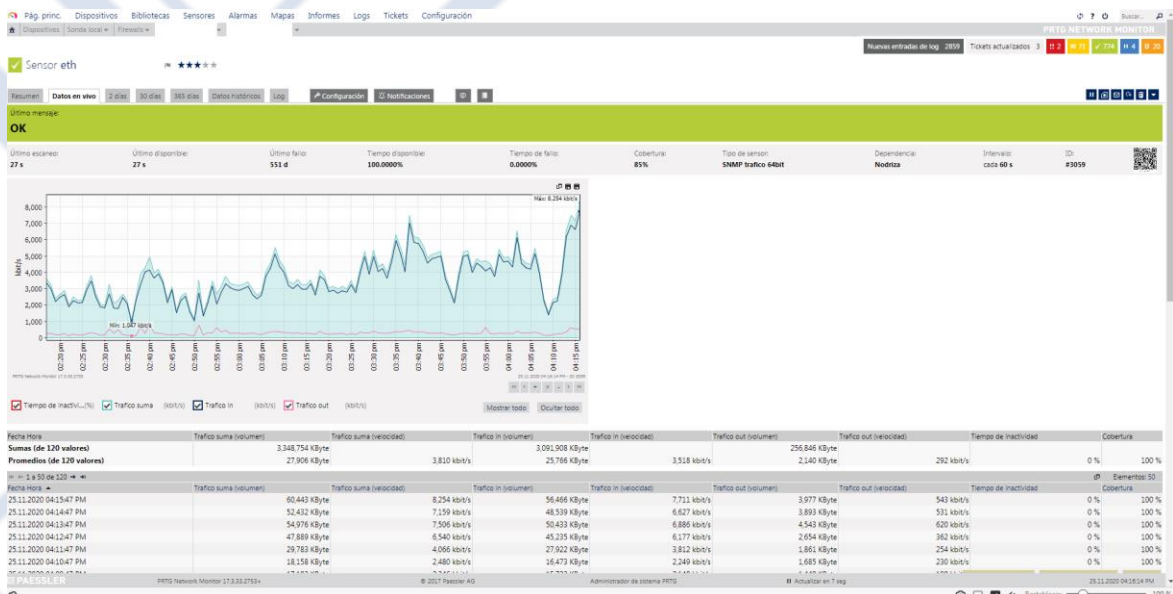




Figura 7. Ejemplo de gestión de rendimiento con la herramienta PRTG en BioCubaFarma.

Gestión de fallos

La Figura 8 muestra una captura de pantalla con la herramienta PRTG sobre las reglas para detectar fallos de umbral.

The screenshot displays the PRTG Network Monitor web interface. The top navigation bar includes links for 'Pág. princ.', 'Dispositivos', 'Bibliotecas', 'Sensores', 'Alarmas', 'Mapas', 'Informes', 'Logs', 'Tickets', and 'Configuración'. The main content area is titled 'Sensor Espacio de disco 1' and shows a 'Configuración' tab. Below this, there are sections for 'DESENCADENADORES QUE PUEDEN SER HEREDADOS DE OBJETOS SUPERIORES' and 'DESENCADENADORES DEFINIDOS EN LOS OBJETOS DE BIBLIOTECA'. The 'DESENCADENADORES DE OBJETO' section is expanded, showing a table of rules for disk space thresholds.

Tipo	Notificaciones	Acciones
Estado de desencadenador	Quando el estado del sensor es Fallo durante al menos 600 segundos, realizar Correo electrónico y notificación distribuida al administrador	Editar Eliminar
Estado de desencadenador	Quando el estado del sensor es Fallo durante al menos 900 segundos, realizar Sin notificación y repetir cada 0 minutos	Editar Eliminar
Estado de desencadenador	Quando se borre la condición tras el lanzamiento de una notificación, realizar Correo electrónico y notificación distribuida al administrador	Editar Eliminar
Desencadenador de umbral	Si Espacio disponible D: (%) channel es Debajo 25 por lo menos por 30 segundos, active Correo a cuenta de todos los miembros del grupo Grupo de usuarios PRTG	Editar Eliminar
Estado de desencadenador	Quando el estado del sensor es Advertencia durante al menos 60 segundos, realizar Correo a cuenta de todos los miembros del grupo Grupo de usuarios PRTG	Editar Eliminar
Estado de desencadenador	Quando el estado del sensor es Advertencia durante al menos 300 segundos, realizar Correo a cuenta de todos los miembros del grupo Grupo de usuarios PRTG y repetir cada 15 minutos	Editar Eliminar
Estado de desencadenador	Quando se borre la condición tras el lanzamiento de una notificación, realizar Correo a cuenta de todos los miembros del grupo Grupo de usuarios PRTG	Editar Eliminar
Estado de desencadenador	Quando el estado del sensor es Fallo durante al menos 15 segundos, realizar Correo a cuenta de todos los miembros del grupo Grupo de usuarios PRTG	Editar Eliminar
Estado de desencadenador	Quando el estado del sensor es Fallo durante al menos 300 segundos, realizar Correo a cuenta de todos los miembros del grupo Grupo de usuarios PRTG y repetir cada 5 minutos	Editar Eliminar
Estado de desencadenador	Quando se borre la condición tras el lanzamiento de una notificación, realizar Correo a cuenta de todos los miembros del grupo Grupo de usuarios PRTG	Editar Eliminar

Figura 8. Ejemplo de gestión de fallos con la herramienta PRTG en BioCen.

Gestión de seguridad

La Figura 9 muestra una captura de pantalla con la herramienta OSSIM para el análisis de vulnerabilidades.

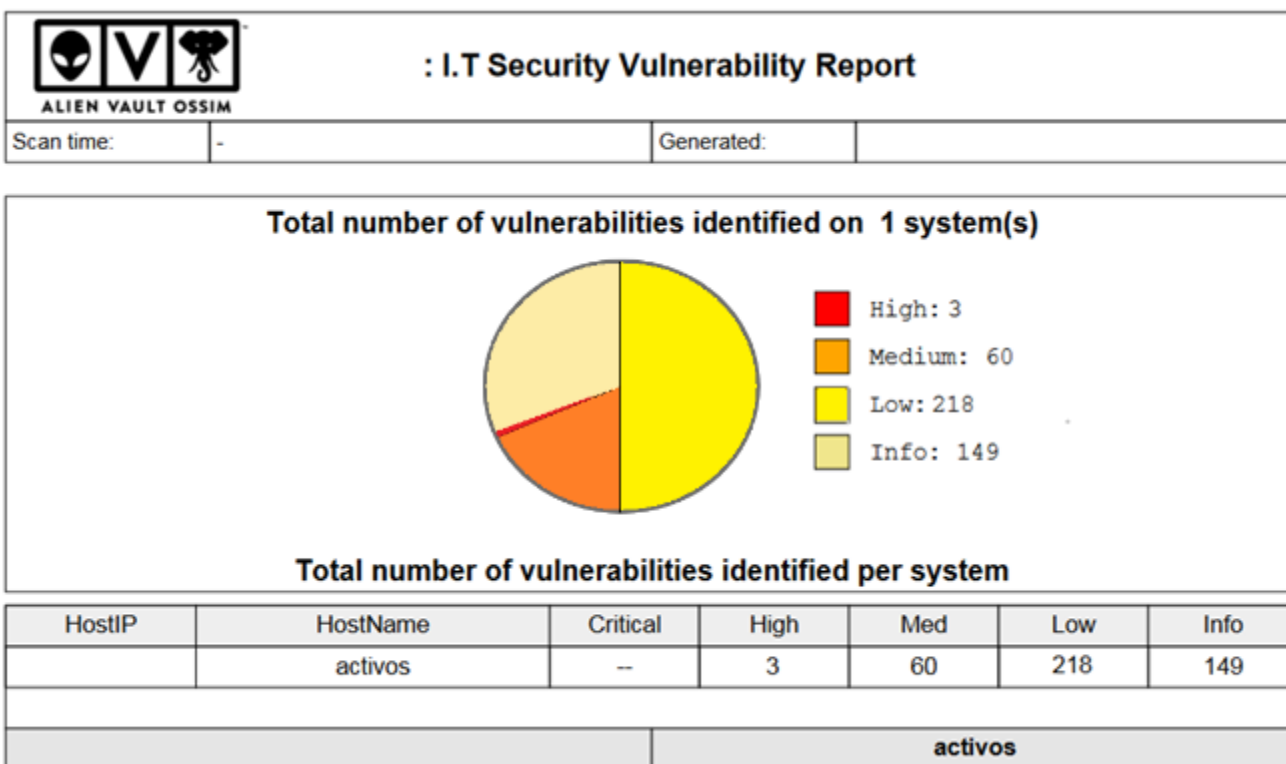


Figura 9. Ejemplo de gestión de seguridad con la herramienta OSSIM en BioCen.

GESTIÓN DE REDES EN BIOCEN

A continuación, se muestra la configuración de redes, la seguridad de servicios, auditoría, incidentes y algunos servicios disponibles para los usuarios de BioCen.

Configuración de la red, servicios y distribución del cabling

La Figura 10 muestra el diseño de la red local donde se aprecian algunas características. Posee un cableado central de fibra óptica para los exteriores y que enlaza a cada una de las edificaciones y cableado par trenzado para el interior de las instalaciones. Todos los servicios que se brindan están soportados en los servidores físicos y virtuales.

La entidad proveedora del servicio de conectividad para BioCen es la Empresa de Tecnología de Información (ETI), con la cual se ha establecido el contrato correspondiente, esta nos brinda los



servicios de conectividad con la Red BioCubaFarma, así como, la navegación por Internet, servicios de bases de datos de información científica, acceso a la Nube, etc.

La conexión contratada es protegida por un cortafuego administrado por nuestra entidad.

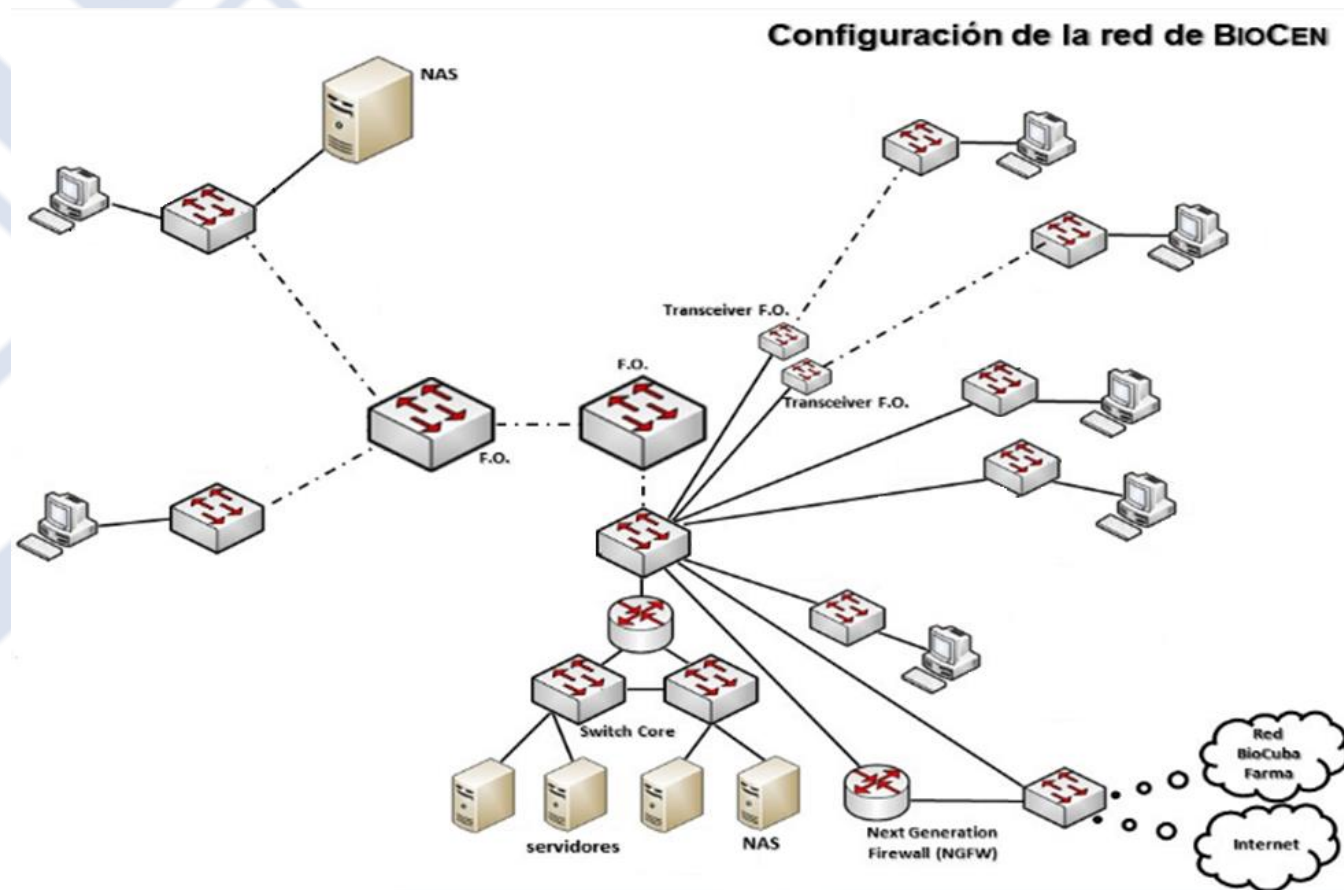


Figura 10. Diseño de la red local en BioCen.

Seguridad implementada en la sala de servidores

Todos los servidores están protegidos contra el ataque de virus informáticos con la instalación de antivirus propios para servidores, que se actualiza periódicamente de forma automática. Algunos servidores instalados con LINUX y otros con Windows Server.

Solo un grupo muy reducido de usuarios cuentan con privilegios especiales dentro de la red, la habilitación y control de este servicio es realizado a través de las Políticas de Acceso del Servidor Proxy utilizando el administrador del mismo. Se realizan salvas periódicas de la información



contenida en los servidores, la cual se programa en dependencia del grado de sensibilidad para la Institución y la frecuencia de actualización de la misma, separándose en diaria, semanal o mensual. Los servidores cuentan con sistemas de protección eléctrica (UPS) que garantizan la continuidad del proceso informativo en casos de fallos de corriente eléctrica.

En situaciones de desastre existe un local climatizado en un edificio alejado del nodo central, donde tenemos equipamiento con las réplicas de todos los elementos crítico que tenemos en los servidores centrales.

Servicio de correo

BioCen cuenta con un dominio propio biocen.cu, registrado en el Centro Cubano de Información de Red (CUBANIC), CITMATEL, según contrato que se renueva anualmente; por lo que todas las direcciones de correo definidas en nuestro servidor salen con el sufijo @biocen.cu. Sólo tienen acceso al correo electrónico internacional los usuarios que han sido debidamente autorizados por la Dirección General del Centro. Esto es habilitado y controlado por el Administrador de Red a través de las facilidades que brindan las herramientas de administración del software que brinda este servicio.

Servicio de Internet

La conexión a redes externas se brinda a través de un router, que trabaja con una línea dedicada conectada a un equipo que realiza la transmisión de datos a través de fibra óptica dedicados para internet y conectividad nacional. La conexión a la red interna se realiza a través de un proxy con un corta fuego, que se mantienen actualizados. El control de la conexión se realiza por la coincidencia del nombre de usuario, dirección IP y número MAC correspondiente al hardware de cada estación de trabajo y en horarios establecidos para cada uno de ellos. Se registran todos los accesos a Internet y se monitorean periódicamente.

Los servicios son filtrados hacia el interior de la red aprovechando las facilidades de filtrado de paquetes con que cuentan los servidores Proxy. El servidor destinado a esta tarea está configurado de manera tal que:

- Se bloquean del exterior al interior todos los servicios que no se utilicen y los puertos correspondientes.
- No se permite la entrada de paquetes que contengan alguna de nuestras direcciones interiores como dirección de origen.
- Sólo se permite la salida hacia INTERNET de las estaciones de trabajo de los departamentos



que hayan sido previamente autorizados a conectarse a INTERNET y verificando que coincida el número IP del ordenador con el usuario autorizado a utilizar este servicio desde él.

Servicio de acceso remoto

El servicio de acceso remoto se brinda vía MODEM que están dedicados al enlace desde el exterior de la Institución a la red como una terminal más. Este servicio está instalado en las estaciones de trabajo personales en los hogares de directivos y especialistas principales de las direcciones, según necesidad Institucional y con la autorización del Director General.

Servicio de Acceso a la red mediante enlace Wifi

El servicio de acceso a la red mediante el enlace WIFI de equipos móviles, es autorizado solo por la máxima dirección de la entidad, la Directora General. En cada equipo se aplican las siguientes medidas de seguridad:

- Anclaje de IP a la MAC del móvil para conexión a la WIFI de BioCen.
- Acceso a Internet y Correo Electrónico empresarial por WorldClient mediante conexión segura.
- Configuración que Brinda la ETI para la conexión a la VPN de BioCubaFarma.
- Instalación de Antivirus.
- Instalación y configuración de Corta Fuego.

Auditoria al sistema de red

En BioCen se establecen diferentes tipos de inspecciones o auditorias, de forma tal que se abarcan todos los elementos que intervienen en la seguridad del sistema, las cuales están dirigidas a:

- La inspección se realiza en conjunto con la Especialista de la Oficina para el Control de la Información Clasificada (OCIC) de la Institución.
- Anualmente se realizan inspecciones internas a la seguridad implementada en los puestos de trabajo habilitados para el procesamiento de la IOC.
- Trimestralmente se realizan inspecciones a los controles y servicios implementados en la Red de BioCen.



- Se establece que anualmente los Activistas de Seguridad Informática de las áreas realizarán autocontroles para chequear el cumplimiento de las medidas definidas en el plan de seguridad informática.

Análisis estadístico de los servicios

Mensualmente, utilizando las posibilidades de generación de estadísticas que poseen las herramientas para el monitoreo y control de los servicios de redes, se elaboran reportes de los servicios de Internet, correo electrónico y acceso remoto; las cuales incluyen los siguientes datos:

- Para Internet: Identificación de los usuarios, relación de los sitios más accedidos en el período en que se está monitoreando y cantidad de accesos realizados. Resumen del comportamiento de las categorías que pudieran indicar alguna incidencia o indisciplina, así como el análisis de las que se detecten.
- Para Correo Electrónico: Identificación de la cuenta, área a la cual pertenece, tipo de contenido (laboral u otros), cantidad de correos recibidos o enviados, así como los usuarios que más se comunican hacia el exterior. Se realiza un análisis de la racionalidad en el uso de este servicio y las cuentas implicadas en cualquier tipo de indisciplina detectada.
- Para el Servicio de Acceso Remoto: Se analizan los usuarios que utilizaron este servicio para detectar accesos no autorizados. Las estadísticas de los servicios son analizadas y procesadas, elaborándose un informe con los resultados del monitoreo.

Respuesta a incidentes

En caso de detectarse alguna anomalía en el uso de alguno de estos servicios que se brindan en BioCen, el mismo es interrumpido inmediatamente por el Administrador de la Red y se informa al Director del área a la que pertenece el usuario responsable del mismo, profundizando en el análisis de los elementos que causaron la misma. En caso de detectarse algún incidente de Seguridad Informática o violación de lo establecido en el PSI, por cualquiera de las vías de auditoría o monitoreo, se procede según procedimiento establecido para esto.

RESULTADOS Y DISCUSIÓN

La importancia de una buena gestión de redes de datos



La gestión de redes de datos es un concepto amplio, que abarca su administración desde un enfoque completo. Dentro de ella se engloban políticas y procedimientos que intervienen en su planteamiento y configuración, así como el control y monitoreo de cara a evitar fallos y reforzar la seguridad, con el fin de asegurar la calidad de los servicios esperados.

La administración de redes, por lo tanto, es la suma de actividades orientadas a mantener una red eficiente, que tenga una alta disponibilidad. De ahí la importancia de llevar a cabo una buena gestión de redes de datos pues, en última instancia, un funcionamiento idóneo es un gran aliado para el buen funcionamiento de la empresa.

A mayor tamaño y complejidad de la red, más necesario será contar con un sistema de administración adaptado a sus necesidades, en el que se incluyan todos los aspectos relacionados con un adecuado funcionamiento. Solo así será posible prevenir y detectar problemas, incluyendo aspectos de seguridad, con la mayor anticipación posible, buscando minimizar errores.

Estándares y protocolos, un aspecto clave

Una buena gestión de redes de datos requiere, fundamentalmente, basarla en un modelo con tareas bien definidas sujetas a estándares y protocolos para facilitar tanto su implementación como su actualización.

En este sentido, cobran importancia los sistemas de gestión de redes de datos, al tiempo que se suele centralizar la gestión para vigilar el funcionamiento de la red o redes de la empresa. Una diversidad que viene acompañada de protocolos y estándares. Su aplicación, por lo tanto, buscan ese control eficaz de la red para que ésta pueda responder en todo momento.

En este contexto, existen distintos protocolos de gestión de red, algunos de ellos modelos estándar, como el SNMP que lo aplicamos en BioCen.

CONCLUSIONES

Cada vez es más difícil llevar a cabo una adecuada gestión, capaz de dar acceso a los servicios que proporcionan las redes de datos de forma eficiente, pues estos también crecen en número y complejidad. Se trata de un reto que aumenta su dificultad conforme ganan en complejidad los sistemas de redes y, junto con ello, lógicamente se disparan las expectativas de funcionamiento. Afortunadamente, también la gestión de red ha ido evolucionando de forma paralela, a medida que han ido aumentando los diferentes servicios.



Como consecuencia de ello, este creciente desarrollo de servicios ha marcado la necesidad de gestionarlos convenientemente para que los usuarios puedan satisfacer sus necesidades sin interrupciones. A tal efecto, se precisan soluciones avanzadas, que monitoricen y automaticen procesos, así como buenas prácticas que ayuden a mejorar las capacidades necesarias en la entrega de servicios.

Actualmente nos encontramos ante un panorama diverso, si en sus inicios la gestión de red se basó sobre todo en la monitorización del tráfico de red, la detección de errores y el establecimiento de la calidad de servicio (QoS), en la actualidad disponemos de sistemas heterogéneos, lo que ha supuesto la proliferación de diferentes sistemas de gestión de red; donde se destacan los que emplean las técnicas de correlación de eventos que permiten en breve tiempo detectar fallas en el sistema e interrelacionar datos que cada día se vuelven más complejos. En BioCen se continúa mejorando nuestro sistema acorde como avance la tecnología, con el apoyo de la dirección de la empresa y la creación de nuevos proyectos sobre la gestión de redes.

REFERENCIAS

- [1] T. Anttila, "WOIS Automation System Monitoring", 2018.
- [2] M. A. García Domínguez, "Intranet en centro de educación secundaria", B.S. thesis, 2013.
- [3] Z. Abidin, "LKP: Manajemen Proxy Server Berbasis Kerio Control pada Kantor Wilayah BPN Provinsi Jawa Timur", PhD Thesis, Institut Bisnis dan Informatika Stikom Surabaya, 2018.
- [4] S. Kamal, I. M. Helal, S. A. Mazen and S. Elhennawy, "Computer-Assisted Audit Tools for IS Auditing", Internet of Things-Applications and Future, Springer, pp. 139-155, 2020.
- [5] D. Karg, "Open source security information management (OSSIM)", OSSIM, Standard Document, 2003.
- [6] M. Engine, NetFlow Analyzer. Network Traffic Analysis with NetFlow. Available: <https://www.manageengine>, 2014.
- [7] M. Ammar, M. Rizk, A. Abdel-Hamid and A. K. Aboul-Seoud, "A framework for security enhancement in SDN-based datacenters", 8th IFIP international conference on new technologies, Mobility and security, pp. 1-4, 2016.
- [8] F. Ö. Sönmez and B. Günel, "Evaluation of Security Information and Event Management Systems for Custom Security Visualization Generation", International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism, pp. 38-44, 2018.



- [9] M. A. Hafsaoui and H. Mansour, "Développement d'une application de gestion du parc informatique", PhD Thesis, Université Virtuelle de Tunis, 2019.
- [10] D. Dimov, "Adaptive Patching Strategy", Constanta Maritime University Annals, vol. 27, no. 223, 2019



LaSalle
Universidad
Perú