

HeuristicTool, herramienta para el apoyo de evaluaciones heurísticas a sistemas interactivos por medio de ontologías

HeuristicTool, tool for supporting heuristic assessments to interactive systems through ontologies

Duván Steven Naranjo-Martínez¹
César Alberto Collazos-Ordóñez²
Royer David Estrada-Esponda³

Recibido: junio 30 de 2020

Aceptado: noviembre 12 de 2020

Resumen

Este trabajo expone una herramienta denominada HeuristicTool, mediante la cual se apoya el proceso de evaluación heurística, permitiendo a los evaluadores mejorar el proceso de análisis de los resultados y, en general, el desarrollo de dichas evaluaciones. Durante su desarrollo y permanente actualización, se incluyeron ontologías para facilitar la capitalización y transferencia de conocimiento, entre los diferentes evaluadores. Lo anterior a través de KACTUS, una metodología que propone pasos en relación con: la especificación de la aplicación, diseño preliminar con base en categorías ontológicas y un refinamiento y estructuración de la ontología. Un grupo de tres expertos calificaron 6 variables de la ontología, consolidando un promedio de 4.7 puntos sobre 5. Además, una evaluación de usabilidad a la plataforma presentó resultados satisfactorios, por medio de la batería de heurísticas de Jakob Nielsen. El principal resultado fue una aplicación que brinda a los usuarios la posibilidad de revisar evaluaciones estructuradas en una base de conocimiento, con el fin de mejorar la toma de decisiones.

Palabras clave: evaluación heurística, ontología, sistemas de recomendación, usabilidad.

Abstract

This work presents a tool called HeuristicTool, by means of which the heuristic evaluation process is supported, allowing evaluators to improve the process of analyzing the results and, in general, the development of said evaluations. During its development and permanent updating, ontologies were included to facilitate the capitalization and transfer of knowledge between the different evaluators. The above through KACTUS, a methodology that proposes steps in relation to: the specification of the application, preliminary design based on ontological categories and a refinement and structuring of the ontology. A group of three experts rated 6 ontology variables, consolidating an average of 4.7 out of 5. In addition, an evaluation of the platform's usability presented satisfactory results, using Jakob Nielsen's battery of heuristics. The main result was an application that gives users the possibility of reviewing structured evaluations in a knowledge base, in order to improve decision making.

Keywords: heuristic evaluation, ontology, recommendation systems, usability.

¹ Ingeniero de Sistemas, Universidad del Valle, Tuluá, Colombia. E-mail: naranjo.duvan@correounivalle.edu.co
ORCID: <https://orcid.org/0000-0002-4496-9740>

² Ingeniero de Sistemas, Doctor en Ciencias Mención Computación, Universidad del Cauca, Popayán, Colombia. E-mail: ccollazo@unicauca.edu.co
ORCID: <https://orcid.org/0000-0002-7099-8131>

³ Ingeniero de Sistemas, Magíster en Administración, Universidad del Valle, Tuluá, Colombia. E-mail: royer.estrada@correounivalle.edu.co
ORCID: <https://orcid.org/0000-0002-6849-1278>

1. Introducción

En la actualidad es cada vez más común el uso diario de computadores o dispositivos tecnológicos. Por ejemplo, la cantidad de horas que los jóvenes entre 18 y 35 años dedican al uso de la computadora es elevada, en algunos casos más de 8 horas diarias, ya que no solo la utilizan para el estudio o trabajo, sino también para la recreación (Fernández-Garza, 2017).

En este sentido, diversas disciplinas buscan que los sistemas sean usables, seguros y funcionales. Una de ellas es la Interacción Persona-Ordenador (IPO) o Interacción Humano-computador (HCI, por sus siglas en inglés), la cual tiene como tesis fundamental que entre más amable es el proceso de interacción con el usuario, mejor será la realización de tareas (Martínez et al., 2011). En consecuencia, la usabilidad ha cobrado significativa relevancia en el éxito de una aplicación interactiva y por ello, su evaluación se ha convertido en un importante campo de investigación. Sin embargo, aún existe confusión en el uso de modelos, métodos, técnicas y herramientas para evaluar la usabilidad (Sierra, 2013).

La usabilidad se define como el grado en el que un producto puede ser usado por usuarios específicos, para conseguir unos objetivos determinados, con: eficacia, eficiencia y satisfacción, en un contexto de uso específico (Yáñez, 2019). Para medir el nivel de usabilidad de un sistema interactivo, existen diversas técnicas y metodologías, entre las que se destacan las evaluaciones heurísticas como medio para verificar los elementos de la interfaz de un sistema interactivo, mediante una serie de principios heurísticos establecidos previamente por diversos autores (González et al., 2001).

Una evaluación heurística (EH), es una de las técnicas de evaluación de usabilidad más extendidas en el contexto de la evaluación de los sistemas interactivos. Esta consiste en la comprobación de los elementos de una lista de heurísticas que, por su naturaleza, no pueden comprobarse automáticamente y requieren del criterio de un experto (Yáñez, 2019). Este méto-

do se destaca por su bajo costo, aunque puede variar dependiendo de cuántos evaluadores se consideren. Sin embargo, el uso de EH trae resultados positivos (García et al., 2019). El análisis tiene como objetivo hallar los problemas de usabilidad en el diseño de la interfaz de usuario, con el fin de facilitar su corrección mediante el diseño iterativo (Suárez, 2011).

Sin embargo, para realizar las correcciones que requiere el sistema interactivo, se necesita información con el fin de solucionar los problemas hallados durante la evaluación (Riascos, 2018). Además, dicha información no se gestiona de forma eficiente, ya que la poca información que se encuentra se debe localizar y reconstruir. Debido a esto, los evaluadores y los desarrolladores deben consultar la escasa información disponible, lo que origina que parte del conocimiento existente se pierda o no sea utilizado eficazmente (Walz et al., 1993).

Ante el avance tecnológico de la sociedad del conocimiento, consultar cierta información que en algunas ocasiones es equivocada y en otras ocasiones es difícil de hallar, repercute en el factor temporal tanto para los evaluadores como desarrolladores de sistemas interactivos. También, omitir información importante que puede ser sustancial para el proceso de evaluación, es una posibilidad en momentos donde la información abunda, pero no está debidamente estructurada.

Asimismo, diversas herramientas que apoyan la realización de evaluaciones heurísticas, carecen de mecanismos que en primera instancia mantengan disponibles todos sus servicios y, por otro lado, faciliten la estructuración de información relacionada con los procesos de evaluación. No obstante, existen herramientas vigentes que cuentan con mecanismos que pueden mitigar la problemática antes mencionada. Una de ellas fue elaborada por Riascos (2018), y tiene como propósito servir de instrumento para aumentar la eficiencia por medio de la informatización de los procesos de planificación y puesta en marcha, propios de este tipo de evaluaciones. Esta herramienta también contribuye con la confiabilidad

de los resultados, logrando así apoyar la toma de decisiones. Sin embargo, en dicha herramienta se evidenció la necesidad de estructurar la información de modo que los datos sean interpretados más eficientemente, facilitando su consulta por agentes no humanos, como los computadores. Esto facilita la gestión del conocimiento que puede ser utilizado por diversos evaluadores, además de ofrecer la capacidad de realizar deducciones de posibles soluciones basadas en la experiencia de usuario.

Para este último componente, existen técnicas como las ontologías que permiten agrupar y estructurar la información, además de permitir la codificación del conocimiento (Taket et al., 1999). Esto permite determinar las relaciones existentes entre los conceptos de un dominio específico y gestionar el conocimiento, con el fin de evitar la pérdida o desaprovechamiento de la información (Checkland & Holwell, 2005).

El objetivo del artículo, es exponer la herramienta HeuristicTool actualmente disponible, que incluye la incorporación de ontologías ligeras y la posibilidad de asistir la corrección de errores de usabilidad de los sistemas interactivos evaluados, por medio de un modelo de recomendación inicialmente apoyado por la asignación de patrones de diseño, los cuales hacen parte de la ontología incorporada. El resto del artículo se organiza así: la sección dos contiene una revisión de trabajos sobre diversos sistemas de información que cumplen el propósito de soportar las evaluaciones heurísticas, y la metodología que guio la investigación. La sección tres expone los resultados y discusión y, finalmente, la sección cuatro presenta las conclusiones.

2. Marco teórico y metodología

2.1 Herramientas de evaluación heurística

Entre las herramientas para apoyar la evaluación heurística, está USABAIPO-GESTORHEURISTICA, que es una herramienta de escritorio basada en el diseño de un gestor de heurísticas. Se basa

en almacenar información pertinente para la experimentación, así como puntuaciones y observaciones acerca del sistema evaluado, los datos del sistema a evaluar y criterios heurísticos (Masip, 2010). Por otro lado, Kemp y Setungamudalige (2006), fueron los desarrolladores de R-IDE, que en su momento daba soporte para realizar evaluaciones heurísticas. En esta herramienta, se apreciaban diversas técnicas de evaluación, como: cuestionarios, heurísticas, recorridos cognitivos y observaciones, donde se podía evaluar el sistema interactivo (Kemp & Setungamudalige, 2006).

La aplicación Gestor Heurísticas, permite realizar evaluaciones heurísticas. En ella se observaban dos módulos, el primero que permite crear nuevos proyectos de evaluación heurística eligiendo los evaluadores deseados, y el módulo evaluador, a través del cual es posible llevar un control sobre todos los proyectos evaluados (Masip, 2010). Asimismo, SIRIUS es una aplicación web que cuenta con funciones ontológicas que permiten encontrar el nivel de usabilidad de forma cualitativa y cuantitativa. En este caso, es posible establecer criterios para mejorar el sistema interactivo y visualizar los comentarios asociados de los evaluadores ante un sistema evaluado (Suárez, 2011).

El SQUaC (Software Quality Usability and Certification), es un grupo de investigación de la Universidad Politécnica de Valencia, encargado de desarrollar el prototipo de una herramienta capaz de gestionar datos relacionados con todo el proceso de la evaluación heurística, denominado ACCUSA. A través de la herramienta, se pueden crear heurísticas, perfil de evaluadores y listados o informes con problemas detectados de usabilidad (Masip, 2010). Así mismo, ECUSI (Evaluación Colaborativa de la Usabilidad de Sistemas Interactivos) es una herramienta desarrollada para brindar métodos para realizar evaluaciones de usabilidad, como: evaluaciones heurísticas, interacción constructiva y métodos de interrogación (entrevistas). Esta herramienta permite la creación de evaluaciones, el manejo de diversos evaluadores y fácil acceso a la información (Solano et al., 2015).

La mayoría de las herramientas consultadas son poco vigentes, salvo HEURISTICTOOL, herramienta aquí utilizada, que permite la realización de evaluaciones heurísticas, a través de diversos principios heurísticos. Esta herramienta permite: la evaluación cualitativa y cuantitativa, asignar evaluadores a la evaluación, realizar una conclusión final en línea con los evaluadores, generar reportes sobre la evaluación realizada, y llevar un control de las evaluaciones realizadas por el usuario (Riascos, 2018).

En la Tabla 1, se presenta el comparativo de varias herramientas para la realización de evaluaciones heurísticas. En ella se observa que en la mayoría de los casos, las herramientas no están disponibles para los usuarios. Tampoco cuentan con un sistema que estructure la información, para facilitar el aprendizaje de los evaluadores por medio de recomendaciones.

Tabla 1. Comparación de aspectos básicos de herramientas para evaluación heurística.

| Aplicación | Gestión de heurísticas | Gestión de evaluadores | Resultados | Perfil objetivo | Implementación | Lenguaje de programación | Disponible | Criterios de mejora |
|-----------------|------------------------|------------------------|--|-----------------|----------------------|--------------------------|------------|---------------------|
| Usabaipo-Gestor | Si | No | | Evaluador | Microsoft Escritorio | | No | No |
| Heurística | | Si | | Evaluador | Web | Office | No | No |
| R-IDE | No | No | | Evaluador | Microsoft Access | | No | No |
| Gestor | Si | Si | Informe cualitativo del nivel de usabilidad | Diseñador | Escritorio | PHP / MySQL | No | No |
| Heurísticas | Si | Si | Informe con problemas detectados | Evaluador | Web | .NET | No | No |
| Sirius | No | Si | Informe con problemas detectados | Evaluador | Microsoft Escritorio | PHP | No | Si |
| Accusa | Si | Si | Informe cualitativo o cuantitativo del nivel de usabilidad | Diseñador | Office Access | | No | No |
| Ecusi | Si | Si | Informe con problemas detectados | | Web | Python / Javascript | No | No |
| HeuristicTool | | Si | Informe con problemas detectados | | | PHP / MySQL | Si | No |

2.2 Metodología

El soporte metodológico del presente trabajo, incluyó: metodologías para el desarrollo de software, SCRUM en este caso; KANBAN para la gestión de actividades en proyectos ágiles; y una metodología para la creación de ontologías. Para esta última, después de una revisión sobre enfoques metodológicos se decidió utilizar KACTUS (Guzmán et al., 2012), que se basa en la reutilización de conocimiento, para lo cual requiere un dominio preexistente. Esta metodología consta de 3 etapas, a saber:

La primera etapa, denominada Especificación de la aplicación, hace referencia a la descripción del dominio en el que se encuentra el sistema interactivo, así como también el alcance a cubrir y objetivos de interés que realizará la ontología. En el contexto de esta investigación, el dominio hace referencia a evaluaciones heurísticas por medio de la herramienta HeuristicTool.

La segunda etapa, se enfoca en el diseño preliminar basado en categorías ontológicas, que consiste en desarrollar un diseño para la ontología, así como también mapear el dominio específico en donde será implementada.

La tercera etapa corresponde al Refinamiento y estructuración de la ontología, que hace referencia a mejorar los aspectos de la ontología que aún no han sido concretados o definidos, como también a solucionar los problemas que presente para luego realizar la estructuración ontológica. Esta fase se complementó con el juicio de expertos, acudiendo al concepto de tres investigadores en el ámbito de las ontologías. Se les indagó, sobre: el cumplimiento del propósito que motivó la creación de la ontología; lo entendible y extensible de su codificación; y sobre su estructura taxonómica, el vocabulario utilizado y la redundancia de conceptos. Estas variables se evaluaron con una escala Likert de 1 a 5.

3. Resultados y discusión

Con base en la metodología KACTUS, se inició con el desarrollo de la ontología, posteriormente incluida en HeuristicTool. A continuación,

se presentan cada una de las tres etapas que constituyen dicha metodología.

3.1 Especificación de la aplicación

La primera etapa de desarrollo implicó obtener la jerarquía de clases. Para ello, se realizó la revisión y posterior análisis de la aplicación HeuristicTool, revisando principalmente la forma como están organizados los datos dentro de la misma, para luego poder representarlos. Las funcionalidades principales de HeuristicTool, permiten: configurar el ambiente de trabajo para una evaluación heurística, desarrollar dicha evaluación, analizar los resultados y generar reportes generales, ayudando al análisis y a la toma de decisiones. A partir del análisis realizado, se encuentra que el dominio que cubriría la ontología está relacionado con la información de evaluaciones heurísticas, realizadas a diferentes sistemas interactivos. Por ello, el objetivo principal se basa en proveer información a los evaluadores para que puedan realizar consultas, de manera eficiente, a una base de conocimiento estructurada.

Se observa que la clase principal es "Evaluación", ya que tiene relación con otras clases, como: i) características, cada evaluación cuenta con unas características que pueden ser similares a otras; ii) usuario creador, Cada evaluación cuenta con un único usuario creador. Es importante considerar esta clase en la jerarquía porque más adelante los usuarios serán útiles en mecanismos de recomendación, que filtren por las características propias del usuario, como rol, intereses, etc. iii) Conjunto heurístico, Las heurísticas utilizadas para resolver la evaluación; y iv) recomendación, Esta consta de los patrones de interacción seleccionados como una posible solución a los problemas encontrados.

Con base en lo anterior, se definió la jerarquía de clases o taxonomía, representada en la figura 1. Entre las características incluidas en la taxonomía, se consideraron: el tipo de implementación (web, móvil o escritorio); tipo de aplicación (E-commerce, páginas académicas, juegos serios, etc); tipo de evaluación (pública o privada); estado; fecha de creación y fecha de vencimiento.



Figura 1. Taxonomía definida.

3.2 Diseño preliminar basado en categorías ontológicas

En la segunda etapa, se usó UML para representar la ontología del sistema HeuristicTool. Dicha representación se realizó por medio de un modelo conceptual a nivel de datos, con el fin de exponer el dominio que sirvió como base para compartir el significado (o la semántica),

y el conocimiento entre los seres humanos y el sistema. Esto se aplica a todas las ontologías existentes. Las ontologías promueven la reutilización del conocimiento del dominio y mejoran la visibilidad de los supuestos del dominio, mediante especificaciones explícitas. En la Figura 2 se presenta el modelado de datos, en el cual se evidencia las relaciones de las entidades que interactúan en el proceso.

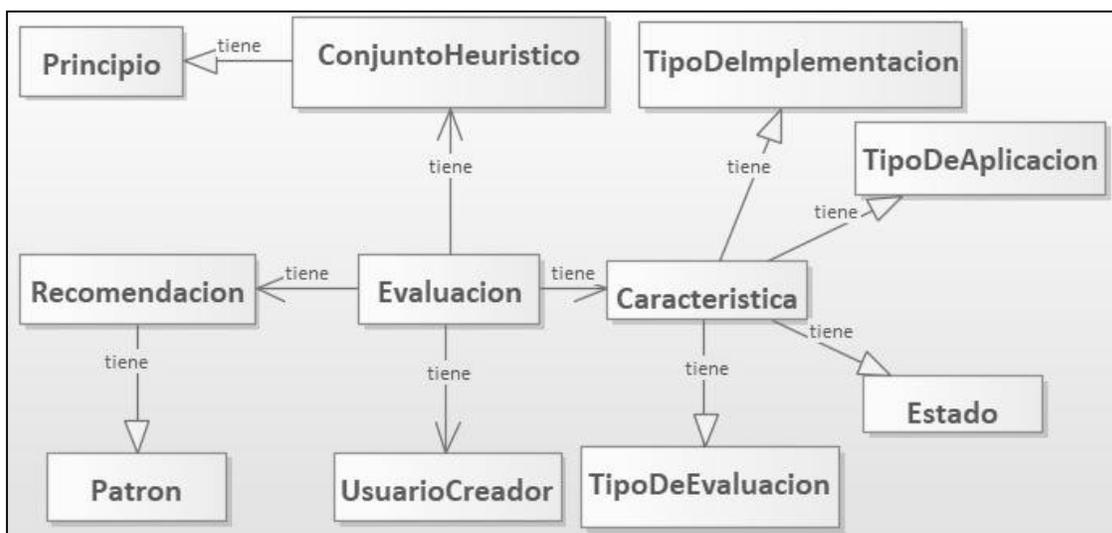


Figura 2. Modelo de datos.

3.3 Refinamiento y estructuración de la ontología

Para la construcción exitosa de la ontología, se utilizó la herramienta Protégé versión 5.5.0, basándose en la abstracción de los términos utilizados en el modelo de datos de la Figura 2.

En el proceso de definición de las clases con la herramienta Protégé, se utilizó la funcionalidad de creación de clases suministrada por dicha herramienta. Es importante destacar que la jerarquía de las clases se conservó, al realizar la transformación del modelo de datos a la ontología. Luego, para la definición de las propiedades o relaciones de las clases, se utilizó la funcionalidad Propiedad del Objeto, con la cual es posible reutilizar las propiedades para todas las clases que sean creadas dentro de un proyecto. En la definición de las restricciones asociadas a cada una de las propiedades, se crearon las relaciones entre las clases, utilizando las propiedades. Por ejemplo, se creó una propiedad llamada

tieneConjuntoHeurístico, la cual representa la relación existente entre las clases Evaluación y ConjuntoHeurístico. Lo que se hizo fue crear una propiedad que almacene una o más instancias de los ConjuntosHeuristicos, permitiendo que en el momento que se cree una instancia de la clase Evaluación y se necesite conocer los ConjuntosHeuristicos en los cuales se subdivide esa Evaluación, se pueda obtener al cargar el valor de la propiedad tieneConjuntoHeurístico.

En la creación de las instancias o individuos (Individuals), se seleccionó la ventana "Individuos Por Clase", la cual se proporciona a los editores y los navegadores para realizar la carga de la información. En la Figura 3, se muestra un ejemplo de la información cargada para la clase ConjuntoHeurístico. En la ventana "Class Hierarchy", se observa seleccionada la clase ConjuntoHeurístico, y al presionar cada una de estas instancias en la ventana "Individuals", se observan las diversas instancias asociadas a la misma, con sus respectivos valores.

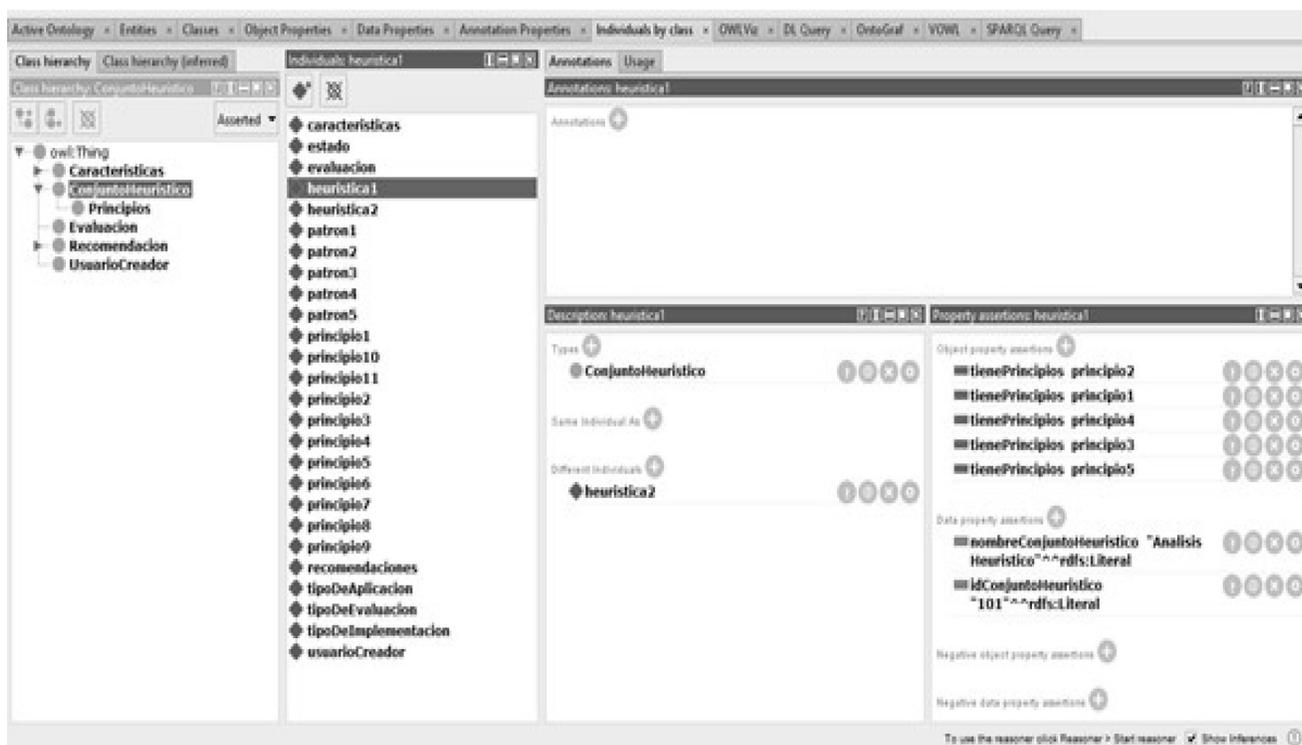


Figura 3. Instancias de la ontología presentada en Protégé.

De esta forma, la ontología con sus respectivas instancias, quedó preparada para ser consumida por algún sistema de manejo de conocimientos, el cual permita manejar y mostrar la información contenida en la misma.

3.4 Verificación de la ontología creada

La validación se realizó en Validador RDF, herramienta que permite la validación de una ontología en formato RDF, mostrando informes

de errores y comentarios detallados relativos al código fuente. En esta prueba, inicialmente se utilizó la exportación en formato *.RDF de la ontología desarrollada con Protégé. Este archivo fue introducido en la página del validador, obteniendo resultados satisfactorios. Adicionalmente, tres expertos opinaron sobre seis variables en una escala Likert, que permiten juzgar la ontología creada. En la Tabla 2, se aprecia que todas las variables fueron calificadas de forma satisfactoria.

Tabla 2. Evaluación de la ontología por expertos.

| Variable | Pregunta | Promedio |
|----------------------------------|--|----------|
| Propósito | ¿La ontología cumple el propósito para el cual fue creado? | 4.5 |
| Extensibilidad y Compresibilidad | ¿Su codificación es entendible y extensible? | 5 |
| Estructura Taxonómica | ¿Es correcta su estructura taxonómica? | 5 |
| Validez del Vocabulario | ¿El vocabulario empleado es válido? | 4.5 |
| Redundancia | ¿Presenta redundancia de conceptos? | 4 |
| Metodología | ¿Está construida de forma correcta la ontología? | 5 |

3.5 Ontología integrada a HeuristicTool

Para crear la ontología directamente desde HeuristicTool, se creó una plantilla en el lenguaje PHP (lenguaje en el cual está codificado HeuristicTool). Esta plantilla genera automáticamente la ontología de la evaluación, permitiendo mantener nutridas las ontologías de la aplicación sin intervención humana.

Inicialmente, se extrae de la base de datos relacional toda la información necesaria para la creación de la ontología. Las evaluaciones que aplican deben tener registrado al menos un patrón, deben estar en estado "Terminada" y el

tipo de evaluación debe ser "Pública". Luego, se procede a registrar el IRI de la ontología con estado activo, así como a crear el archivo vacío OWL.

Para nutrir el archivo *.OWL, se dividió el proceso en varias partes. Cada parte consta de una cadena tipo String, en formato XML, que se crea con base en los datos obtenidos de las consultas descritas anteriormente. Este proceso se hace recursivamente hasta completar la totalidad de los datos definidos en la ontología, y se concatena en el archivo OWL, con la función de PHP fwrite. De esta forma queda adaptada la ontología a la aplicación HeuristicTool, permitiendo la creación automática de una ontología pública

para el consumo de algún sistema de manejo de conocimientos, que permita manejar y mostrar la información contenida en la misma.

Paralelamente, se utilizó una aplicación externa denominada Visual Notation for OWL Ontologies (VOWL). Esta aplicación proporciona representaciones gráficas de archivos OWL, lo cual facilita la

interpretación de las ontologías por parte de los usuarios. Un ejemplo de cómo se visualiza una ontología en VOWL, se muestra en la Figura 4. Vale la pena aclarar que VOWL también se puede configurar para que trabaje en el mismo host donde se aloja HeuristicTool. Además, se elaboró la terminología de la ontología para facilitar la comprensión de la misma (Naranjo, 2019).

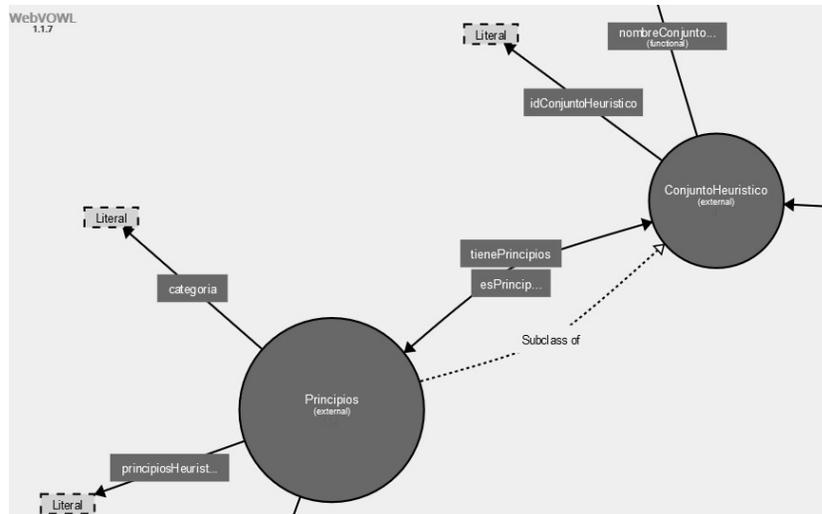


Figura 4. Fragmento de la ontología, visualizada en VOWL.

La Figura 5 refleja dos partes importantes de HeuristicTool: A) los diferentes módulos disponibles una vez se inicia sesión, y B) la interfaz para la visualización pública del grafo escogido OWL, asociado a cada evaluación disponible.

HeuristicTool

- 🏠 HOME
- 👤 USUARIOS ▾
- 👤 PERFILES ▾
- 📄 HEURÍSTICAS ▾
- 📁 EVALUACIÓN ▾
- 📄 PATRONES ▾
- 🕒 REPORTE

EVALUACIONES PÚBLICAS CON ONTOLOGÍAS DISPONIBLES

Evaluaciones Públicas B

Estas son las evaluaciones públicas desarrolladas en HeuristicTool que se encuentran terminadas

| Número | Fecha Creación | Empresa | Usuario Creador | Fecha Vencimiento | OWL Code | OWL Graph |
|--------|----------------|-----------|------------------------------|-------------------|----------|-----------|
| 76 | 2019-Aug | no aplica | kevin caicedo i | 2019-09-05 | | |
| 77 | 2019-Aug | NO APLICA | cristian felipe espinal soto | 2019-09-04 | | |
| 80 | 2019-Sep | No Aplica | cristian felipe espinal soto | 2019-09-26 | | |
| 81 | 2019-Sep | Proalba | Kevin Vargas | 2019-09-05 | | |
| 82 | 2019-Sep | | David Alejandro Gil Botina | 2019-09-12 | | |

Figura 5. Menú y listado de evaluaciones terminadas con visualización en VOWL.

También se añadió un sistema de recomendación en HeuristicTool, que está basado en patrones de interacción. Para ello, se comparan las características de la evaluación en proceso con evaluaciones ya terminadas, y que sean: del mismo tipo de implementación; del mismo tipo de aplicación; contar con al menos un conjunto heurístico igual; ser pública; y que cuente con al menos un patrón asignado. Si existen evaluaciones que cumplan dichas condiciones, se propone una posible solución, que incluirá los patrones utilizados para resolver los problemas encontrados en la evaluación usada como referencia. Además, cuenta con una ontología de dominio público, que puede ser utilizada para la representación de conceptos.

3.6 Evaluación de usabilidad a HeuristicTool

La evaluación de usabilidad fue realizada por 4 profesionales en Ingeniería de Sistemas, quienes utilizaron la batería de heurísticas de Jakob Nielsen. De las 40 calificaciones realizadas, el 50% indica que en su "totalidad no hay problemas de usabilidad", esta última una de las escalas de calificación de dicha batería. Solo el 1% de las calificaciones, apuntan a problemas mayores de usabilidad. Entre los comentarios de los evaluadores, se encuentra que la aplicación muestra un grado de usabilidad y accesibilidad apropiadas para una aplicación de este tipo. No obstante, es posible una mejora, con elementos como: un tutorial dinámico o guiado, el uso de atajos de teclado y acciones favoritas o frecuentes de los usuarios más expertos. El informe completo de la evaluación, así como los principios usados, se encuentran disponibles como archivo subsidiario de este artículo.

4. Conclusiones

El empleo de herramientas para apoyar las evaluaciones heurísticas es limitado en nuestra región. De hecho, se menciona en Estrada et al. (2018), que en el contexto colombiano solo un reducido grupo de instituciones de educación

superior forman conocimientos en esa materia, lo cual coincide con lo dicho por Granollers et al. (2008), en términos de México y Colombia. Por lo tanto, fortalecer una herramienta como HeuristicTool y mantenerla disponible, podría aumentar el interés de diferentes personas sobre la usabilidad y, en general, en algunas áreas de la Interacción Persona-Ordenador.

Se evidenció que no hay disponibilidad para realizar evaluaciones heurísticas, que incluyan ontologías para ser consumidas por algún sistema de manejo de conocimientos, que permita manejar y estructurar la información, además de facilitar la toma de decisiones. Por ello, la actualización realizada a HeuristicTool es un aporte innovador desde el punto de vista tecnológico. Además, se contribuye con la ampliación del estado del arte en temas referentes a: usabilidad, evaluaciones heurísticas y ontologías.

La aplicación HeuristicTool registra 78 evaluaciones. Por ello, como actividades futuras, se destacan la generación de una ontología pesada que se alimente de todas las ontologías que, progresivamente, se vayan generando en la aplicación; y además incorporar un mecanismo de consultas SPARQL para facilitar la consulta de la ontología, y su posterior enlazamiento y publicación en escenarios como la web de datos.

La ontología que hace parte de la aplicación HeuristicTool, fue valorada de manera satisfactoria por un grupo de 3 expertos. Esto demostró, al menos en primera instancia, que la integración de esa ontología podría ser útil para apoyar evaluaciones heurísticas a sistemas interactivos de diferente naturaleza. Además, se convierte en un criterio adicional que diferencia a HeuristicTool de las otras herramientas relacionadas en este artículo, aparte de ser la única herramienta disponible actualmente de todas las revisadas.

Paralelamente, se realizó una evaluación de usabilidad a la herramienta por medio de los 10 principios de Jakob Nielsen. Fue gratificante ver como en HeuristicTool se manifiestan a satisfacción la mayor cantidad de principios. Sin embargo, también es claro que constantemente deben

realizarse acciones de mejora para contribuir cada vez más con la usabilidad de la herramienta.

Finalmente, el mantenimiento de la aplicación HeuristicTool también incluye un mecanismo de recomendación basado en patrones de interacción, para facilitar la toma de decisiones. No obstante, previendo que el uso de la aplicación puede aumentar, se considera como trabajo futuro la incorporación de un sistema de recomendación basado en contenido o de filtro colaborativo, con el fin de hacer más eficiente el proceso de recomendación y por tanto la toma de decisiones.

Referencias

- Casado-Martínez, C., Garreta-Domingo, M., Hassan-Montero, Y., Martínez-Normand, L., & Mor-Pera, E. (2011). *Interacción persona-ordenador*. Recuperado de: [https://www.guao.org/sites/default/files/biblioteca/Interacción persona ordenador .pdf](https://www.guao.org/sites/default/files/biblioteca/Interacción%20persona%20ordenador.pdf)
- Checkland, P., & Holwell, S. (2005). Data, capta, information and knowledge. In *Introducing Information Management: The Business Approach*. <https://doi.org/10.4324/9780080458397>
- Estrada-Esponda, R. D., Riascos-Pareja, C. A., & Loaiza-Duque, Á. A. (2018). La interacción humano computador en el currículo de las instituciones de educación superior de Colombia. *Revista De Investigación, Desarrollo E Innovación*, 9 (1), 147– 162. <https://doi.org/10.19053/20278306.v9.n1.2018.7722>
- Fernández-Garza, E. (2017). *Prevención del Síndrome del Túnel Carpiano por el uso de computadoras*. <https://repositoriotec.tec.ac.cr/handle/2238/9600>
- García-Toribio, G., Polvo-Saldaña, Y., Hernández-Mora, J. J., Sánchez-Hernández, M. J., Nava-Bautista, H., Collazos-Ordóñez, C. A., & Hurtado-Alegría, J. A. (2019). Medición de la usabilidad del diseño de interfaz de usuario con el método de evaluación heurística: dos casos de estudio. *Revista Colombiana de Computación*, 20 (1), 23–40. <https://doi.org/10.29375/25392115.3605>
- González, M., Masip-Ardévol, L., Granollers, A., & Oliva, M. (2009). Advances in Engineering Software Quantitative analysis in a heuristic evaluation experiment. *Advances in Engineering Software*, 40 (12). <https://doi.org/10.1016/j.advengsoft.2009.01.027>
- González, M., Pascual, A., & Lorés, J. (2001). Evaluación Heurística.
- Granollers, T., Collazos, C. A., & González, M. P. (2008). The state of HCI in Ibero- American countries. *Journal of Universal Computer Science*, 14(16), 2599–2613. <https://doi.org/10.3217/jucs-014-16-2599>
- Guzmán-Luna, A., López-Bonilla, M., & Torres, I. (2012). Metodologías y métodos para la construcción de ontologías. 50, 133–140. <https://doi.org/http://dx.doi.org/10.22517/23447214.6693>
- Kemp, E., & Setungamudalige, D. T. (2006). A resource support toolkit (R-IDE): Supporting the DECIDE framework. *ACM International Conference Proceeding Series*, 158, 61– 65. <https://doi.org/10.1145/1152760.1152768>
- Masip-Ardévol, L. (2010). *Análisis de la viabilidad para la automatización de la evaluación heurística*. Universidad de Lleida, España. <http://hdl.handle.net/10459.1/45842>
- Naranjo-Martínez, D. (2019). *Terminología de la Ontología*. HeuristicTool. <https://www.heuristictool.com/Terminologia>
- Riascos-Pareja, C. (2018). *Aplicación web para el apoyo de evaluaciones de usabilidad de sistemas interactivos mediante múltiples conjuntos heurísticos* (recurso electrónico). Universidad del Valle. https://opac.univalle.edu.co/cgiolib/?infile=authsecsearch.glu&style=authk&nh=20&calling_page=details.glu&key=383166
- Sierra-González, J. (2013). *Métodos de Evaluación de Usabilidad para Sistemas de Información Web: una revisión*. Universidad Nacional de Colombia, 1–13. http://www.bdigital.unal.edu.co/54171/1/m_usabilidad.pdf

Solano, A., Ceron, J. C., Collazos, C. A., & Fardoun, H. M. (2015). ECUSI: Herramienta software para la evaluación colaborativa de la usabilidad de sistemas interactivos. *10th Colombian Computing Conference, 10CCC*, 157–163. <https://doi.org/10.1109/ColumbianCC.2015.7333418>

Suárez-Torrente, M. del C. (2011). *Sirius: sistema de evaluación de la usabilidad web orientado al usuario y basado en la determinación de tareas críticas*. Universidad de Oviedo. <https://dialnet.unirioja.es/servlet/tesis?codigo=22190>

Taket, A., Checkland, P., & Holwell, S. (1999). Information, Systems and Information Systems-Making Sense of the Field. *The Journal of the Operational Research Society*, 50(5). <https://doi.org/10.2307/3010006>

Walz, D. B., Elam, J. J., & Curtis, B. (1993). Inside a software design team: Knowledge acquisition, sharing, and integration. *Communications of the ACM*, 36 (10), 63–77. <https://doi.org/10.1145/163430.163447>

Yáñez-Gómez, R. (2019). *Evaluación Heurística de la usabilidad en contextos específicos*. Universidad de Sevilla. <https://idus.us.es/handle/11441/89691>