

## Evaluación del Atributo de Calidad del Software ‘Aprendizaje’ empleando un Método Multicriterio

José L. Andrada<sup>1</sup>, Ana Funes<sup>2</sup>, Aristides Dasso<sup>2</sup>

1 Universidad Nacional de La Rioja, La Rioja, Argentina  
joseluisandrada2002@gmail.com

2 SEG / Departamento de Informática / Facultad de Ciencias Físico-Matemáticas y Naturales / Universidad Nacional de San Luis  
Ejército de los Andes 950, D5700HHW San Luis, Argentina  
+54 (0) 266 4520300, ext. 2126  
{afunes, arisdas}@unsl.edu.ar

**Resumen.** La motivación de este trabajo es proponer la adopción de un modelo para la evaluación del atributo de calidad ‘Aprendizaje’ en el marco de aplicaciones web, tanto simples como complejas, permitiendo conocer de antemano cuán fácil y eficientemente podrán los usuarios realizar sus tareas habituales empleando una aplicación como soporte. Para la creación del modelo propuesto, se siguen los pasos indicados por el método LSP (Logic Scoring of Preference), un método multicriterio que permite la creación de modelos cuantitativos para la evaluación, optimización, comparación y selección de sistemas complejos. LSP es un método de decisión multicriterio y multiatributo que respalda la elaboración de modelos que pueden dar importancia diversa a los distintos atributos considerados, en este caso elementos en un árbol de requisitos de adecuación al atributo de calidad ‘Aprendizaje’, permitiendo elegir qué elementos tendrán una importancia más destacada en función de las necesidades del proyecto considerado. Por otro lado, se propone seguir directrices de normas reconocidas, como la International Organization for Standardization (ISO), como referencia para la creación del modelo, por ejemplo, los lineamientos dados en la norma ISO 9241-110: Ergonomics of Human-system Intereaction, la cual sirve de guía a organizaciones y desarrolladores para, entre otras cosas, formular e implementar estrategias de desarrollo a lo que dan en llamar ‘Learnability’ del software. En particular, en el presente trabajo, estas normas han servido de guía para seleccionar características y sub-características que hacen al modelo de evaluación del atributo ‘Aprendizaje’. Estas características han sido aumentadas con otras propias, enriqueciendo así, y en base a otras propuestas analizadas, los requisitos considerados para la evaluación.

**Palabras claves.** Facilidad de Aprendizaje. Aprendizaje (Learnability) de sistemas web. Usabilidad. Atributos de Calidad del Software. Métodos de Evaluación. Logic Score of Preference (LSP). ISO/IEC 25000.

## 1 Introducción

Tanto en un proyecto de desarrollo de un nuevo producto así como cuando, en una organización, se desea adquirir un nuevo producto de software, es imperativo contar con herramientas que permitan medir la calidad de los distintos atributos de ese producto. Una de las características que hacen a la calidad del software es el atributo ‘Aprendizaje’. De acuerdo con la norma ISO/IEC 25010 [19], y como puede verse en la Fig. 1, que muestra cada uno de los componentes de la calidad de software, el atributo ‘Aprendizaje’ es uno de los seis componentes de calidad de la ‘Usabilidad’ (o Capacidad de Uso); otros son la ‘Inteligibilidad’ (Appropriateness Recognizability), ‘Operatividad’ (Operability), ‘Protección frente a errores de usuarios’ (User Error Protection), ‘Estética de la interfaz de usuario’ (User Interface Aesthetics) y la ‘Accesibilidad’ (Accessibility). En este sentido, contar con un modelo que permita conocer qué tan fácil es para los usuarios aprender a usar un producto de software, conocer y controlar los atributos que contribuyen a hacer el producto amigable, fácil de aprender y usar, es de vital importancia.



Fig. 1. Atributos de la calidad del software según ISO/IEC 25010

La construcción de modelos para evaluar sistemas complejos, incluidos los sistemas de software, si bien es una necesidad importante, no es una tarea sencilla. En particular, en el caso de la evaluación del atributo ‘Aprendizaje’, se deben tener en cuenta varios aspectos, por ejemplo, aspectos funcionales del sistema, aspectos relacionados con la facilidad de entendimiento como la legibilidad, la carga de trabajo, la orientación al usuario, la previsibilidad, entre otros. Cada una de estas propiedades caracterizará la rapidez del aprendizaje con la que los clientes podrán utilizar cada uno de las funcionalidades provistas, lo más rápidamente y con la mejor profundidad posible.

En consecuencia, es importante disponer, en una organización que se preocupa por la calidad de sus productos de software, de instrumentos adecuados para evaluar el grado de adecuación a estos principios y, en particular, conocer y controlar el atributo ‘Aprendizaje’ de un sistema de software, independientemente de que el software en cuestión se encuentre bajo desarrollo o esté ya desarrollado.

En este sentido, existen en la literatura y en las web múltiples propuestas. En [21] se elabora un Modelo de Atributos de Aprendizaje, basado en lo indicado en la ISO/IEC 25010: 2011 [19]. Se descompone el atributo de calidad ‘Aprendizaje’ en sub ítems. En el mismo artículo se presenta, además, un caso de estudio.

Manuela Unsöld, en su tesis de posgrado [23], discute varias definiciones de ‘Aprendizaje’ así como métodos de evaluación de este atributo de calidad del software; finalmente, construye un framework para evaluar y clasificar los distintos métodos presentados.

Robillard [22] realiza una encuesta con el objetivo de encontrar las causas que hacen que una API sea difícil de aprender a usarla. Identifica obstáculos que agrupa en cinco categorías: Recursos; Estructura; Antecedentes; Medioambiente Técnico; Proceso.

Asimismo, Grossman et al. [13] presentan un estudio de las definiciones, mediciones y metodologías de evaluación de la capacidad de aprendizaje del software. Muestran un sistema de clasificación de problemas de aprendizaje y demuestran cómo estas categorías pueden conducir a pautas para abordar los desafíos asociados.

## **2 Estándares y Otros Modelos Considerados en la Creación del Modelo de Evaluación del Atributo ‘Aprendizaje’**

El modelo propuesto en este trabajo para evaluar el atributo ‘Aprendizaje’ toma como base la propuesta de Abrahão, E. Insfran Pelozo, A. Fernandez Martinez [2] para la evaluación de la Usabilidad en productos software orientados a la web. Dicha propuesta se ajusta a la normativa ISO 25000 SQuaRE [18] y contempla las dos vistas del modelo de calidad de la norma: una perspectiva centrada en el producto software y otra perspectiva centrada en la calidad en uso. Dado que el trabajo aquí presentado se focaliza en el atributo ‘Aprendizaje’, hemos considerado sólo aquellos atributos del modelo mencionado [2] que tienen que ver en forma directa con aprender a usar un producto software o que de alguna manera afectan ese aprendizaje.

La ISO 9241-110 [20] es una norma que también ha sido considerada para el desarrollo del modelo propuesto, ya que en una de sus secciones, la sección 4.6 ‘Suitability for learning’, establece las características que tienen que tener los diálogos en un software para decir que es adecuado para aprender a usarlo. En sus 7 ítems encontramos ejemplos de cómo debe ser el software para ser fácil de aprender o tenga condiciones para que se aprenda.

La norma ISO 25010 [19] menciona la ‘Facilidad de entendimiento’ (Understandability) como la capacidad de un producto de ser entendido por nuevos usuarios en términos de su propósito y cómo puede ser usado en tareas específicas. Si bien en dicha norma, la ‘Facilidad de Entendimiento’ no está considerada como una sub-característica de ‘Aprendizaje’ sino de la ‘Usabilidad’, consideramos que tiene una relación directa en el aprendizaje, motivo por el cual forma parte del modelo propuesto. Por otro lado, en la ISO 25000 (SQuaRE) [18] se incorpora una nueva sub-característica, la ‘Facilidad de Ayuda’, que ya se mencionaba en las métricas de la norma ISO 9126-2 [17] como una de las métricas relacionadas con el ‘Aprendizaje’ y que hace mención a la relevancia con la que el producto ofrece ayuda a los usuarios cuando la necesitan. Estas dos características, junto con la que hemos llamado ‘Facilidad de Aprendizaje’, que comprende otro conjunto de sub características de ‘Apre-

dizaje’, han sido consideradas en nuestro modelo. Sin embargo, dado que estas tres características identificadas son demasiado abstractas, las mismas han sido descompuestas en otras sub características de menor granularidad, pasibles de ser medidas. Para ello, hemos adoptado algunas de las sub características del modelo propuesto en [2], tal como se menciona anteriormente.

En lo que nos concierne, además, este modelo ha sido ajustado teniendo en cuenta las reglas ergonómicas propuestas por Bastien y Scapin [4], algunas métricas, procedimientos y reglas propuestas por M. Unsöld [23], la definición y aprobación de medidas para los modelos de navegación por Abrahão, Condori-Fernández, Olsina y Pastor [1], y las mostradas en los principios ISO 25010 SQuaRE [19] ] referenciados anteriormente, a la vez que se han seguido las pautas del método LSP [8].

### 3 El Método LSP

En el desarrollo del modelo para la evaluación del cumplimiento de los requisitos de adecuación al Aprendizaje, proponemos la adopción del método LSP (Logic Scoring of Preference) [8]. Se espera que el modelo resultante sea de ayuda para evaluar el cumplimiento del atributo ‘Aprendizaje’ en una aplicación web dada.

LSP es un método para la realización de funciones de criterio complejas y su aplicación en la evaluación, optimización, comparación y selección de sistemas complejos en general. Como establece su creador, “la metodología se basa en el trabajo en soft computing, sistemas difusos, multicriterio y toma de decisiones multiatributo”. [8, pág. xvii]. Se ha utilizado en varios dominios para evaluar y medir diversos sistemas en amplias áreas de diferentes organizaciones, no solo en la industria del software, como se puede ver en [5], [6], [7], [10], [11], [12].

Al ser un método de evaluación general, se puede aplicar a diferentes dominios, en particular para los procesos de evaluación involucrados en el atributo ‘Aprendizaje’, en este caso a partir de las hojas de un árbol de requisitos de adecuación al Aprendizaje. Como expresa J. Dujmovic en [8, pág. 368], no es fácil encontrar problemas de decisión multicriterio donde todos los criterios individuales tengan exactamente la misma importancia; por lo tanto, el método LSP es especialmente útil cuando hay decisiones complejas en la evaluación del tipo ‘y/o’, donde los métodos simples de puntuación aditiva no resultan suficientes.

Al igual que con muchos otros métodos, los primeros pasos en LSP implican aclarar cuáles son los requisitos del usuario. Los requisitos vienen en forma de un conjunto de características, que contiene todas las entradas cuantificables que afectan la idoneidad general del objeto evaluado. Las mismas deben agruparse en categorías y subcategorías coherentes, todas ellas organizadas en un árbol; el árbol resultante se denomina Árbol de Requisitos (AR). Las hojas del AR son las características del sistema a medir. Estas características se denominan variables de rendimiento. Como muestra el esquema de la Fig. 2, cada una de estas variables de rendimiento es transformada, durante el proceso de evaluación, en una preferencia elemental por medio de la aplicación de una función de criterio elemental; estos criterios elementales son

definidos previamente, durante el desarrollo del modelo, para cada una de las variables de rendimiento.

La Fig. 2 muestra los principales artefactos que forman parte de un modelo LSP, los que se crean durante las fases prescritas por el método y que se utilizan para la evaluación de un sistema determinado, a saber: el AR, cuyas hojas son las entradas al modelo (variables de rendimiento), las funciones de criterios elementales, las preferencias elementales y la función criterio LSP o estructura de agregación.

Las variables de rendimiento, que corresponden a las características identificadas del sistema y que son las hojas del AR, se instancian, durante la evaluación, con los valores reales observados del sistema bajo evaluación.

Los criterios elementales son funciones que durante la evaluación, dado un valor de una variable de rendimiento (hoja del AR), devuelven una preferencia elemental, la cual no es otra cosa que un valor en el intervalo cerrado  $[0,100]$ . Este valor representa el porcentaje de cumplimiento de un requisito determinado (variable de rendimiento) en el AR, donde 0 significa que el requisito no se ha cumplido en absoluto y 100 que se ha cumplido por completo.

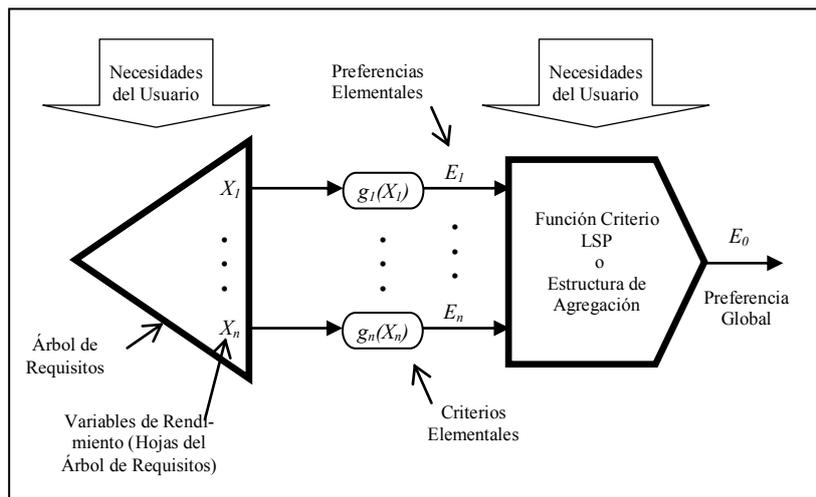


Fig. 2. Descripción general del método LSP.

Las preferencias elementales, obtenidas de la transformación de cada variable de desempeño a través de sus correspondientes criterios elementales son, como se muestra en la Fig. 2, las entradas a la función criterio LSP; las mismas se encuentran agrupadas bajo la forma de una estructura de agregación, la cual, al momento de la evaluación, devolverá un valor entre 0 y 100, que representará el porcentaje de satisfacción con respecto a todos los atributos del AR.

El proceso de agregación es la fase más compleja del desarrollo del modelo y requiere de varios intentos hasta que la estructura de agregación final esté debidamente calibrada. Se construye agregando preferencias elementales mediante un conjunto de operadores. Agregar preferencias significa reemplazar un grupo de preferencias de

entrada por una única preferencia de salida de un operador dado, y denota el grado de satisfacción del evaluador con respecto a todo el grupo de preferencias de entrada. Las preferencias de salida deben agregarse nuevamente hasta que se pueda obtener una única preferencia global. Por tanto, la Función Criterio LSP es el resultado de la combinación de preferencias, teniendo en cuenta tanto la importancia relativa de cada preferencia como la relación lógica entre ellas. Durante la evaluación, la función de criterio LSP produce un único indicador global del grado de cumplimiento de todos los requisitos del sistema, así como un conjunto de indicadores parciales (uno para cada una de las categorías agregadas). En esta fase, es necesario prestar especial atención no solo a las necesidades de los usuarios finales, sino también a la relación entre las preferencias agregadas.

Una vez finalizada la calibración de la función de criterio LSP, el modelo LSP está listo y la evaluación puede realizarse; en el caso presentado aquí, se podrá comenzar con el proceso de evaluación del cumplimiento de adecuación al Aprendizaje de los elementos en el AR. Esto significará recolectar los valores correspondientes a cada una de las variables de desempeño (hojas del AR) y proporcionarlos como entrada a los criterios elementales definidos, cuyas salidas servirán de entradas a la Función de Criterio LSP, para así finalmente obtener un indicador de desempeño global del cumplimiento de adecuación al Aprendizaje con respecto a todos los elementos en el AR, junto a un conjunto de indicadores parciales para cada una de las categorías y sub-categorías identificadas en el mismo. Estos indicadores parciales también son muy importantes ya que muestran claramente cuáles de las categorías y sub-categorías identificadas en el AR presentan fallas de en su cumplimiento.

Para agregar  $n$  preferencias elementales  $E_1, \dots, E_n$  en una sola preferencia  $E_0$ , la preferencia resultante  $E_0$  – interpretada como el porcentaje de satisfacción de los  $n$  requisitos – se expresa mediante una función que cumple con las siguientes propiedades:

$$\min(E_1, \dots, E_n) \leq E_0 \leq \max(E_1, \dots, E_n). \quad (1)$$

La importancia relativa de cada preferencia elemental  $E_i$  ( $i = 1 \dots n$ ) se expresa mediante un peso  $W_i$ ,

Para realizar la agregación de  $n$  preferencias se emplea un conjunto de funciones de Conjunción-Disyunción Generalizada (GCD) que surgen a partir de la instancia-ción de la media de potencia ponderada (Ecuación 2):

$$E(r) = (W_1 E_1^r + W_2 E_2^r + \dots + W_n E_n^r)^{1/r}, \quad (2)$$

donde

$$0 \leq W_i \leq 100, \quad 0 \leq E_i \leq 100, \quad i = 1, \dots, n, \quad W_1 + \dots + W_n = 1, \quad -\infty \leq r \leq +\infty \quad (3)$$

La elección de  $r$  determina la ubicación de  $E(r)$  entre el valor mínimo  $E_{\min} = \min(E_1, \dots, E_n)$  y el valor máximo  $E_{\max} = \max(E_1, \dots, E_n)$ . Para  $r = -\infty$  la media de potencia ponderada se reduce a la conjunción pura (la función mínima C) y para  $r = +\infty$  a la disyunción pura (la función máxima D), dando lugar a una Lógica de Preferencia Continua (CPL), como se muestra en la Fig. 3.

El rango entre la conjunción pura y la disyunción pura está cubierto por una secuencia de operadores CPL, que son funciones de Conjunción-Disyunción Generalizada (GCD). En la Tabla 1 se muestra un conjunto de estos operadores; en ella se han identificado cinco funciones GCD básicas: conjunción (C), quasi-conjunción media (CA), media aritmética (A), quasi-disyunción media (DA), y disyunción (D). Así también, existen cuatro funciones intermedias que interpolan entre cada una de las funciones de agregación anteriores, ellas son: quasi-conjunción fuerte (C+), quasi-conjunción débil (C-), quasi-disyunción débil (D-) y quasi-disyunción fuerte (D+). Este conjunto no es único, pudiéndose identificar un mayor número de operadores, en función de las elecciones de  $r$ ; sin embargo, a los efectos de este trabajo el conjunto de operadores mostrados en la Tabla 1 es suficiente. Para obtener una descripción más detallada, véase [8].

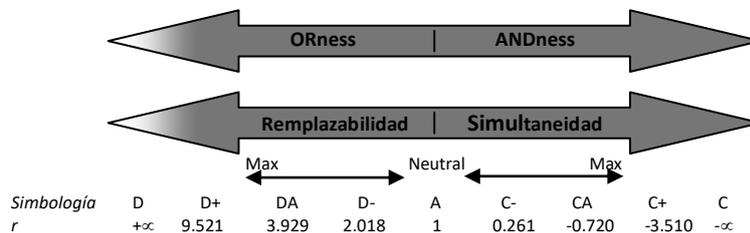


Fig. 3. Niveles de simultaneidad y reemplazabilidad de los operadores lógicos CPL

Tabla 1. Funciones de Conjunción-Disyunción Generalizada

| Req<br>Mand | Nombre de<br>la Operación | Símbolo | Grado de<br>Conj. | Valor de r |           |           |           |
|-------------|---------------------------|---------|-------------------|------------|-----------|-----------|-----------|
|             |                           |         |                   | n = 2      | n = 3     | n = 4     | n = 5     |
| NO          | Disyunción                | D       | 0.000             | $+\infty$  | $+\infty$ | $+\infty$ | $+\infty$ |
| NO          | Disy. Fuerte              | D+      | 0.125             | 9.52       | 11.09     | 12.28     | 13.16     |
| NO          | Disy. Media               | DA      | 0.250             | 3.83       | 4.45      | 4.82      | 5.09      |
| NO          | Disy. Débil               | D-      | 0.375             | 2.02       | 2.19      | 2.30      | 2.38      |
| NO          | Media Arit.               | A       | 0.500             | 1.00       | 1.00      | 1.00      | 1.00      |
| NO          | Conj. Débil               | C-      | 0.625             | 0.26       | 0.20      | 0.17      | 0.16      |
| SI          | Conj. Media               | CA      | 0.750             | -0.72      | -0.73     | -0.71     | -0.67     |
| SI          | Conj. Fuerte              | C+      | 0.850             | -3.51      | -3.51     | -2.18     | -2.61     |
| SI          | Conjunción                | C       | 1.000             | $-\infty$  | $-\infty$ | $-\infty$ | $-\infty$ |

Los pesos ( $W_i$ ) asociados a cada preferencia elemental son asignados por el usuario de acuerdo con la importancia que tiene cada preferencia elemental en el modelo que se está construyendo.

Todos los artefactos mencionados, que forman parte del modelo LSP, se explican con mayor detalle en la sección 4; todos ellos se ilustran a través de ejemplos, donde se muestran estructuras de agregación construidas usando las categorías de Aprendizaje de más alto nivel en el AR así como sub categorías de una de las categorías de primer nivel. También, se da un ejemplo de la definición de un criterio elemental y el AR es presentado.

## 4 Modelo para la Evaluación del Atributo ‘Aprendizaje’ en Aplicaciones Web

En esta sección, se presenta el modelo propuesto para estimar el atributo de calidad ‘Aprendizaje’ en aplicaciones web. Diversas instancias de los diferentes artefactos que forman parte del modelo son mostradas con el objeto de brindar una idea general del mismo.

Las siguientes sub secciones se introducen en un orden similar al con el cual se han creado los distintos artefactos del modelo. Tal y como avala el método LSP, se ha comenzado con la elaboración del Árbol de Requisitos (AR), el cual se presenta en la sub sección 4.1. La siguiente sub sección muestra la definición de uno de los criterios elementales, el cual fue definido para evaluar una de las variables de desempeño del AR. Por último, en la sub sección 4.3, se presentan a modo ilustrativo porciones de la estructura de agregación final.

### 4.1 Árbol de Requisitos

Para empezar, se identificaron las características relevantes para la evaluación del atributo ‘Aprendizaje’ y se organizaron en una estructura de varios niveles, concretamente en lo que el método da en llamar Árbol de Requisitos (AR). El AR contiene las distintas características relevantes del sistema, las cuales se van descomponiendo en varios niveles hasta llegar a atributos que son directamente mensurables.

En el caso de este trabajo, el objetivo es medir el atributo de calidad ‘Aprendizaje’ de una aplicación web. De acuerdo a esto, surge el AR, que aparece en la Fig. 4; el mismo contiene categorías que han surgido a partir de diversas fuentes. Por un lado, las categorías vinculadas al aprendizaje, presentes en el modelo propuesto en [2], y las que surgen de los ejes principales ya mencionados en la Sección 2, la ISO 9126 y la ISO 25000.

Por otro lado, de la norma 9241-110 se ha tomado como guía el punto 4.6.2, el cual se encuentra reflejado en el ítem 3.4 ‘Guía de asesoramiento en procesos poco frecuentes’ del AR; el punto 4.6.3 reflejado en el ítem 1.3 ‘Familiaridad’ y el punto 4.6.4 en el ítem 2.3 ‘Retroalimentación inmediata’ del Árbol de Requisitos.

Las sub categorías 1.1 ‘Legibilidad visual con las hojas’ y 1.1.1 ‘Adecuación de la fuente’ han sido incorporadas de acuerdo al estándar W3C [25]; 1.1.2.1 ‘Proporción de palabras con contraste adecuado’, según norma WCAG 2.0 [26]; 1.1.2.2 ‘Proporción de palabras indicativas e informativas’, de la norma ISO 214-1976 [15]; 1.1.2.3 ‘Cantidad de palabras enfatizadas’, se adecua a normas APA [24] y el ítem 1.1.3 ‘Cantidad de scrolls horizontales’ según norma WCAG 2.0. Las sub-categorías de 2.4 ‘Complejidad de tareas’, 2.5 ‘Comandos’, 3.5 ‘Uso de la documentación de la propuesta’ y 1.4.4 ‘Organización de la información’ según lo descrito en [25], la sub-categoría 1.2 ‘Carga de trabajo’ del modelo propuesto en [4] y la sub-categoría 1.2.2 ‘Densidad de información’ de [1]. Asimismo, las sub características de 1.5.4 ‘Buenos ejemplos’, 3.6 ‘Complejidad de la documentación’ y 1.4.4 ‘La organización de la

información' han sido consideradas de acuerdo a las observaciones dadas en [21] sobre la documentación.

1. Facilidad de entendimiento
  - 1.1 Legibilidad visual
    - 1.1.1 Adecuación de fuente
    - 1.1.2 Adecuación de la visualización textual
      - 1.1.2.1 Proporción de palabras con contraste adecuado
      - 1.1.2.2 Proporción de palabras indicativas e informativas
      - 1.1.2.3 Cantidad de palabras enfatizadas
    - 1.1.3 Cantidad de scrolls horizontales
  - 1.2 Carga de trabajo
    - 1.2.1 Brevedad o agrupación cohesiva de la información
    - 1.2.2 Densidad de información
  - 1.3 Familiaridad
    - 1.3.1 Consistencia de formato
    - 1.3.2 Internacionalización
    - 1.3.3 Metáfora
  - 1.4 Ahorro de esfuerzo
    - 1.4.1 Acciones mínimas
    - 1.4.2 Auto-descripción
    - 1.4.3 Complejidad de la información
    - 1.4.4 Organización de la información
  - 1.5 Orientación al usuario
    - 1.5.1 Mensajes de actualización significativos
    - 1.5.2 Mensajes de aviso significativos
    - 1.5.3 Retroalimentación inmediata de los controles
    - 1.5.4 Buenos ejemplos en la documentación
  - 1.6 Navegabilidad
    - 1.6.1 Soporte a búsqueda interna
    - 1.6.2 Enlaces distinguibles
    - 1.6.3 Interconectividad
    - 1.6.4 Alcanzabilidad
      - 1.6.4.1 Amplitud de la navegación
      - 1.6.4.2 Profundidad de la navegación
      - 1.6.4.3 Densidad de la navegación
      - 1.6.4.4 Número de enlaces rotos
      - 1.6.4.5 Número de nodos huérfanos
2. Facilidad de aprendizaje
  - 2.1 Predictibilidad
    - 2.1.2 Nombres de enlaces significativos
    - 2.1.3 Etiquetas significativas
    - 2.1.4 Controles significativos
    - 2.1.5 Contenido multimedia significativo
  - 2.2 Potencialidad
    - 2.2.2 Determinación de acciones posibles
    - 2.2.3 Determinación de acciones prometedoras
  - 2.3 Retroalimentación inmediata
    - 2.3.2 Progreso explícito de las transacciones
    - 2.3.3 Contexto explícito del usuario
  - 2.4 Completitud de tareas
    - 2.4.1 Capacidad de los usuarios para completar tareas
      - 2.4.1.1 Porcentaje de usuarios que completan una tarea de manera óptima
      - 2.4.1.2 Porcentaje de usuarios que completan una tarea sin ayuda
      - 2.4.1.3 Capacidad para completar la tarea de manera óptima en un cierto tiempo
      - 2.4.1.4 Tiempo para que el usuario complete una tarea con éxito
      - 2.4.1.5 Tiempo para que el usuario complete un conjunto de tareas

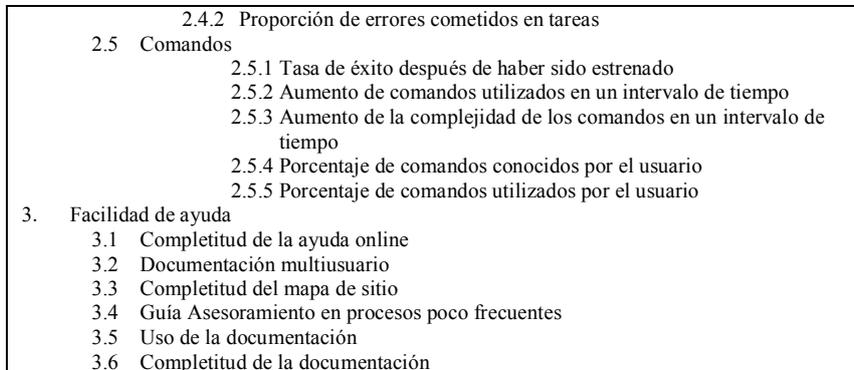


Fig. 4. Árbol de Requisitos para el atributo de calidad 'Aprendizaje' de aplicaciones web

#### 4.2 Criterios Elementales para la Evaluación de las Variables de Rendimiento

Cada criterio elemental  $g_i$  mostrado en la Fig. 2 se encuentra definido como una función que transforma la variable de desempeño  $X_i$  en una preferencia elemental  $E_i$  (valor en una escala normalizada de  $[0,100]$ ). Cada  $E_i$  expresa el grado de satisfacción o cumplimiento del parámetro  $X_i$ .

Los valores que pueblan, durante la evaluación, las variables de rendimiento o desempeño pueden obtenerse por diferentes métodos de acuerdo con el tipo de atributo a medir o calcular. Un proceso de medición permite asignar valores a las variables de desempeño de forma directa, mientras que un proceso de cálculo, mediante una métrica, permite también asignar valores aunque, en este caso, de forma indirecta. Por lo tanto, algunas variables de rendimiento podrían no necesitar una definición sofisticada de criterio elemental, es decir, en aquellas situaciones donde los valores de los atributos se pueden obtener a partir de una medición directa en el intervalo  $[0,100]$ , las preferencias elementales coincidirán con las mediciones para las variables de performance.

Por ejemplo, teniendo en cuenta lo que establecen los autores Abrahão, Condori-Fernandez, Olsina y Pastor en [1], "los modelos de navegación constituyen la columna vertebral del diseño de aplicaciones web, su calidad tiene un gran impacto en la calidad del producto final que realmente se implementa y entrega", se ha elegido mostrar en la Fig. 5 la definición del criterio elemental para la variable de rendimiento 1.6.4.3 'Densidad de la Navegación' (de la subcategoría 1.6.4. 'Alcanzabilidad' de la característica 1.6. 'Navegabilidad', del AR).

Como se puede observar en la Fig. 5, este criterio elemental, que llamamos  $DN$ , provee un indicador del volumen de información y funcionalidad que representa la navegación. Cuanto mayor sea  $DN$ , más tiempo ha de invertir un usuario en realizar cada acción y en consecuencia su capacidad de aprendizaje se ve comprometida debido a la complejidad asociada a la navegación.

Si bien  $DN$  está basado en los conceptos vertidos en [2], lo hemos definido en forma diferente. En [2], los autores lo calculan a partir de lo que llaman NAD (Diagrama

de Acceso Navegacional), construido para un tipo específico de usuario; es decir, en términos MDA, a partir de modelos PIM. En nuestro caso, en cambio, si hablamos en términos MDA, el cálculo es realizado a nivel PSM, es decir, a nivel de la aplicación misma. Más específicamente, *DN* se calcula como el promedio de los cocientes entre el número de enlaces navegacionales y nodos navegacionales. Cada cociente es calculado para cada funcionalidad provista (caso de uso) por la aplicación. Por lo tanto, *DN* se puede obtener analizando las páginas y los hipervínculos entre ellas, incluidos en el código fuente de la aplicación web.

|  |  |
|--|--|
| <b>Tipo de métrica:</b> indirecta  |  |
| <b>Escala:</b> valor real mayor a cero.  |  |
| <b>Objetivo:</b> indicador del volumen de información y funcionalidad que representa la navegación de la aplicación                |  |
| <b>Método de Cálculo :</b>   |  |
| $DN = \begin{cases} 100 & \text{si } 1 \leq PD \leq 3 \\ 50 & \text{si } 3 \leq PD \leq 5 \\ 0 & \text{si } 5 \leq PD \end{cases}$ |  |
| <b>Nombre de métricas auxiliares</b>   | <b>Definición de métrica</b>   |
| Promedio de Densidades ( <i>PD</i> )   | $PD = (\sum_{i=1}^n DNCU_i) / n$ con <i>n</i> = número de casos de uso |
| Densidad de Navegación de un Caso de Uso ( <i>DNCU</i> )   | $DNCU = \frac{NEN}{NNN}$   |
| Número de nodos de navegación ( <i>NNN</i> )   | Número total de nodos de navegación (páginas) para un caso de uso.     |
| Número de enlaces de navegación ( <i>NEN</i> )   | Número total de enlaces de navegación para un caso de uso.             |

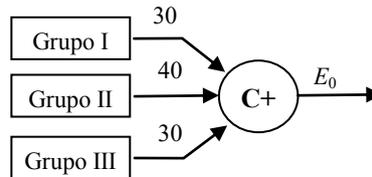
Fig. 5. Definición de criterio elemental para la variable de rendimiento 1.6.4.3 ‘Densidad de la Navegación’

### 4.3 Estructura de Agregación para el Árbol de Requisitos de Adecuación al Aprendizaje

Para decidir qué funciones se deben utilizar en una estructura de agregación es necesario establecer cuál es la relación básica entre las preferencias elementales a agregar, es decir, decidir si la relación es de reemplazabilidad (disyunción), simultaneidad (conjunción) o neutralidad (media aritmética). Por ejemplo, si se pretende establecer una condición de simultaneidad entre dos factores (*X1* y *X2*), se puede elegir el operador *C-* para modelar una simultaneidad débil y que no presenta requisitos obligatorios, o bien puede seleccionar los operadores *CA* o *C+* que modelan una relación de conjunción más estricta.

La Fig. 6 muestra una posible estructura de agregación teniendo en cuenta que las clasificaciones que se dieron para las tres características de alto nivel del AR (Fig. 4) son de simultaneidad fuerte. En la misma, aparecen agregados, por medio de un operador *C+*, los tres grupos de preferencias de nivel superior del AR, a saber: el grupo I

de preferencias correspondiente a la categoría de nivel superior ‘Facilidad de entendimiento’, que a su vez comprende las preferencias de las sub-categorías 1.1 a 1.6; el grupo II para ‘Facilidad de aprendizaje’, que comprende las preferencias agrupadas para las sub-categorías 2.1 a 2.5, y el grupo III para ‘Facilidad de Ayuda’, que abarca las preferencias para las categorías 3.1 a 3.6.



**Fig. 6.** Estructura de agregación para los tres grupos de preferencias correspondientes a las categorías de nivel superior del AR.

En esta estructura de agregación de nivel superior, los tres grupos de preferencias han recibido pesos muy similares; sólo el Grupo II tiene un peso ligeramente mayor, ya que se ha considerado relativamente más importante que los otros dos. Se ha elegido el operador CPL C+ para agregar los tres grupos de elementos. El operador C+ corresponde a una conjunción fuerte (ver Tabla 1), lo que significa que se espera un alto grado de simultaneidad en los elementos obligatorios agregados. Si hubiera presente al menos una entrada completamente insatisfecha, entonces todo el criterio ( $E_0$ ) estaría completamente insatisfecho; por lo tanto, todas las preferencias de entrada a C+ deben satisfacerse al menos parcialmente, de lo contrario, toda la preferencia  $E_0$  se reduce a cero. Además, se puede verificar que, cuando se aplica este operador, la presencia de valores de entrada bajos afecta la salida mucho más que la presencia de valores de entrada altos.

La Fig. 7 muestra otra subestructura de agregación o parte de la función de criterio LSP que se corresponde con las preferencias del conjunto de sub-características de 1.1. ‘Legibilidad visual’ del AR. Por un lado, se ha utilizado el operador CPL C-, una conjunción suave, para agregar las preferencias elementales surgidas de la aplicación de los criterios correspondientes a las variables de rendimiento 1.1.2.1 ‘Proporción palabras con contraste adecuado’, 1.1.2.2 ‘Proporción de palabras indicativas e informativas’ y 1.1.2.3. ‘Cantidad de Palabras enfatizadas’. La consideración, en este caso, fue que los tres componentes se desea tenerlos al mismo tiempo, sin embargo con expectativas no muy altas.

Para la agregación de la preferencia parcial resultante del operador C- con la preferencia elemental 1.1.1 ‘Adecuación de fuentes’ hemos usado el operador C+ donde las dos entradas son simultáneamente necesarias y deben ser satisfechas durante la evaluación, es decir, son componentes obligatorios que dan como resultado una conjunción parcial fuerte.

El agregado siguiente es una estructura particular llamada Absorción Parcial Conjuntiva (CPA), que puede interpretarse como una conjunción asimétrica en la que la parte interesada desea tener simultáneamente las dos entradas, una de ellas obligatoria y la otra opcional. Esta estructura tiene las siguientes características: (a) Si la entrada obligatoria, en este caso proveniente de C+, es cero, entonces la salida  $E_{1,1}$  también

será cero, independientemente del valor de la entrada opcional. (b) Si el requisito obligatorio es parcialmente satisfecho y el opcional es cero, la preferencia resultante  $E_{1.1}$  sufrirá una penalización  $P$ , pero no será cero. (c) Si el requisito obligatorio es parcialmente satisfecho y el opcional es completamente satisfecho, la preferencia resultante sufrirá un incremento o recompensa  $R$  debido a la perfecta satisfacción de la entrada opcional.

Los valores de penalización  $P$  y recompensa  $R$  son los que determinan el efecto que tendrá el operador CPA. Por lo tanto, para modelar una relación de obligatoriedad y opcionalidad, el primer paso es decidir el valor que estos tendrán. En la estructura mostrada en la Fig. 7, se ha aplicado una penalización del 50.3% y una recompensa del 28.5%; la elección de  $P$  y  $R$  determina el valor de los pesos, los cuales pueden ser obtenidos a partir de las tablas P/R aportadas por Dujmovic en [9]. El requisito 1.1.3 es considerado opcional, con respecto a los requisitos 1.1.1 y 1.1.2, teniendo en cuenta que en su utilización lo significativo es la visualización del texto, más allá de que este se encuentre dentro de un scroll horizontal.

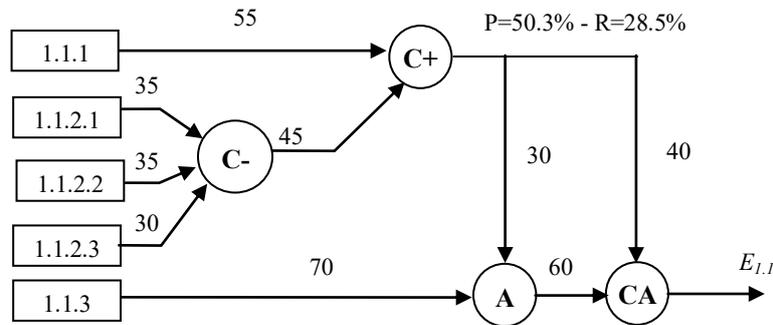


Fig. 7. Estructura de agregación para la preferencia parcial  $E_{1.1}$  correspondiente a la característica 1.1 'Legibilidad visual' del AR.

En cuanto a la asignación de los pesos, se ha adoptado una de las técnicas propuestas por Dujmovic: la asignación directa de pesos [8]. Esta técnica es el enfoque más simple, en la que los evaluadores asignan directamente pesos según los niveles de importancia deseados. La única condición es que la suma de pesos sea igual a 100. Para determinar la importancia de cada requisito con respecto a otro, nos hemos basado en nuestra experiencia como desarrolladores.

## 5 Conclusiones

Hemos mostrado, a través de ejemplos, y siguiendo los lineamientos prescriptos por el método multicriterio LSP, cómo es posible desarrollar un modelo para medir el grado de adecuación de una aplicación web con respecto al atributo de calidad 'Aprendizaje', a partir de un conjunto de requisitos provistos en la forma de árbol. El modelo se ha podido construir y calibrar utilizando un método basado en una lógica graduada, que es un método flexible, adaptable a las necesidades y requisitos de los usuarios finales, y lo suficientemente potente como para resolver no solo problemas

multicriterio, sino también problemas donde decisiones del tipo ‘y/o’ son necesarias y donde un simple método aditivo no alcanza.

El modelo propuesto es lo suficientemente dúctil como para adecuarse a necesidades y requisitos específicos, brindando la posibilidad de adaptar sus artefactos, haciéndolos tan completos como sea necesario; por lo tanto, diferentes categorías o incluso porciones del Árbol de Requisitos de adecuación al Aprendizaje, podrían tomarse en consideración según sea necesario, de acuerdo con la tecnología y procesos involucrados y las necesidades particulares.

Como un resultado extra, el conjunto de categorías identificadas en el Árbol de Requisitos de adecuación al Aprendizaje, junto con sus respectivas descripciones, puede resultar de utilidad como guía en las distintas etapas de desarrollo, ya que brinda una lista comprensible de los requisitos de Aprendizaje en las aplicaciones.

A su vez, la Función Criterio LSP junto con el Árbol de Requisitos puede brindar, en conjunto, una herramienta más precisa para ayudar en la evaluación no sólo del producto terminado sino de los diversos incrementos y versiones que se vayan produciendo de la aplicación, permitiendo así controlar un aspecto que hace a la calidad del producto durante el proyecto de desarrollo, ayudando de esta manera a detectar y fortalecer aquellos aspectos que se encuentran por debajo de los niveles de aceptación.

Finalmente, cabe aclarar que no se trata de un modelo cerrado sino, por el contrario, se pretende continuar perfeccionándolo por medio del análisis de nuevos trabajos de investigación relacionados y el estudio de otras características y métricas, siempre con el objetivo final de mejorar la precisión de los resultados obtenidos.

## Referencias

1. S. Abrahão, N. Condori-Fernandez, L. Olsina, and O. Pastor. Defining and Validating Metrics for Navigational Models. Conference Paper, October 2003. <https://www.researchgate.net/publication/4035117>.
2. S. Abrahão, E. Insfran Pelozo, A. Fernandez Martinez “WUEP: Un Proceso de Evaluación de Usabilidad Web Integrado en el Desarrollo de Software Dirigido por Modelos”, Diciembre 2009.
3. H. Bai, Evaluation and comparison of search engines using the LSP Method. San Francisco State University, Computer Science Dept., Report SFSU-CS-CE-07.12, 2007.
4. C. J. M. Bastien and D. L. Scapin, Ergonomic Criteria for the Evaluation of Human-Computer Interfaces, Institut National de Recherche en Informatique et en Automatique INRIA, 1993.
5. A. Dasso, A. Funes, A Model for Choosing the Right ERP System, SII 2013, 42 JAIIO, 16 al 20 de septiembre de 2013, Córdoba, Argentina.
6. A. Dasso, A. Funes, Web Applications Security Testing Evaluation, SIIO 2020, 49 JAIIO, 19 al 30 de Octubre de 2020.
7. S. J. Dragičević and R. M. Dujmović, Modeling urban land-use suitability with soft computing: the GIS-LSP method. In: Thill, J-C, Dragičević, S. (Eds.) GeoComputational analysis of regional systems. Springer, 2018, pp. 257–275.

8. J. Dujmović, *Soft Computing Evaluation Logic. The LSP Decision Method and Its Applications*. Wiley, IEEE Press. Hoboken, NJ : John Wiley & Sons, 2018.
9. Dujmovic, J. 1979. Partial Absortion Function. Univ. Belgrade, EE Dept., Ser. Mathematics and Physics, 659, 156-163.
10. J. Dujmović, J. W. Ralph, and L. J. Dorfman, Evaluation of disease severity and patient disability using the LSP method. In L. Magdalena, M. Ojeda-Aciego, J. L. Verdegay (Eds.), *Proceedings of the 12th Information Processing and Management of Uncertainty international conference (IPMU 2008)*, pp. 139.
11. J. Dujmović, G. De Tré, and S. Dragičević, Comparison of multicriteria methods for land-use suitability assessment. *Proceeding of the 13th IFSA World Congress and the 6th EUSFLAT Conference, July 20–24, 2009, in Lisbon, Portugal*, pp. 1404–1409, ISBN: 978-989-95079-6-8, 2009.
12. J. Dujmović, G. De Tré, and N. Van de Weghe, Suitability maps based on the LSP method. *Proceedings of the 5th MDAI conference (Modeling Decisions for Artificial Intelligence)*, Sabadell (Barcelona), Catalonia, Spain, October 30–31, 2008. *Lecture Notes in Computer Science*, Vol. 5285, pp. 15–25, Spring.
13. Tovi Grossman, George Fitzmaurice, Ramtin Attar. *A Survey of Software Learnability Metrics, Methodologies and Guidelines*. April 7th, 2009, Boston, MA, USA.
14. Hatch K., Dragičević S., Dujmović J. (2014) Logic Scoring of Preference and Spatial Multicriteria Evaluation for Urban Residential Land Use Analysis. In: Duckham M., Pebesma E., Stewart K., Frank A.U. (eds) *Geographic Information Science. GIScience 2014*. *Lecture Notes in Computer Science*, vol 8728. Springer, Cham
15. ISO 214:1976 - Documentation — Abstracts for publications and documentation - <https://www.iso.org/standard/4084.html>
16. ISO/IEC 9126-1:2001, Software engineering — Product quality — Part 1: Quality model. <https://www.iso.org/standard/22749.html>
17. ISO/IEC TR 9126-2:2003 Software engineering — Product quality — Part 2: External metrics. <https://www.iso.org/standard/22750.html>
18. ISO/IEC 25000:2014, Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE <https://www.iso.org/standard/64764.html>
19. ISO/IEC 25010:2011 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) —System and software quality models. <https://iso25000.com/index.php/normas-iso-25000/iso-25010>
20. ISO 9241-110, International Standard. 2006-04-01. Ergonomics od Human-system Intereaction. Part 110: Dialogue Principles. <https://www.iso.org/obp/ui/#iso:std:iso:9241:-110:ed-2:v1:en>
21. I. Rafique, W. Jingnong, W. Yunhong, M. Qanber Abbasi, P. Lew, and X. Wang, Evaluating Software Learnability. A Learnability Attributes Model. 2012 International Conference on Systems and Informatics (ICSAI 2012).
22. M. P. Robillard, What Makes APIs Hard to Learn? Answers from Developers. November/December 2009 IEEE Software.
23. M. Unsöld, Measuring Learnability in Human-Computer Interaction. Master's thesis at Universität Ulm. Faculty of Engineering, Computer Science and Psychology, Databases and Information Systems Department. Version from September 3, 2018.
24. Normas APA - <https://normasapa.in/>
25. W3C, Recomendaciones sobre técnicas CSS para la mejora de la accesibilidad de los contenidos HTML. <https://www.w3.org/TR/WCAG10-CSS-TECHS/>.
26. W3C, Web Content Accessibility Guidelines (WCAG) 2.0 - <https://www.w3.org/TR/WCAG20/>