big data and cognitive computing

MDPI

Article

Towards a Refined Heuristic Evaluation: Incorporating Hierarchical Analysis for Weighted Usability Assessment

Leonardo Talero-Sarmiento [1,*,†], Marc Gonzalez-Capdevila [2,†], Antoni Granollers [3,†], Henry Lamos-Diaz [4,†] and Karine Pistili-Rodrigues [2,†]

[1] Ingeniería Industrial, Universidad Autónoma de Bucaramanga, Bucaramanga 680003, Colombia
[2] Programa Institucional de Bolsas de Iniciação em Desenvolvimento Tecnológico e Inovação Facens (PIBITIF), Centro Universitario Facens, Sao Paulo 18087-125, Brazil; marc.capdevila@facens.br (M.G.-C.); karine.rodrigues@facens.br (K.P.-R.)
[3] Enginyeria Informática i Disseny Digital, Polytechnic School, Universitat de Lleida, 25002 Lleida, Spain; toni.granollers@udl.cat
[4] Ingeniería Industrial, Universidad Industrial de Santander, Bucaramanga 680002, Colombia; hlamos@uis.edu.co
* Correspondence: ltalero@unab.edu.co; Tel.: +57-607-643-6111 (ext. 309)
† These authors contributed equally to this work.

Abstract: This study explores the implementation of the analytic hierarchy process in usability evaluations, specifically focusing on user interface assessment during software development phases. Addressing the challenge of diverse and unstandardized evaluation methodologies, our research develops and applies a tailored algorithm that simplifies heuristic prioritization. This novel method combines the analytic hierarchy process framework with a bespoke algorithm that leverages transitive properties for efficient pairwise comparisons, significantly reducing the evaluative workload. The algorithm is designed to facilitate the estimation of heuristic relevance regardless of the number of items per heuristic or the item scale, thereby streamlining the evaluation process. Rigorous simulation testing of this tailored algorithm is complemented by its empirical application, where seven usability experts evaluate a web interface. This practical implementation demonstrates our method's ability to decrease the necessary comparisons and simplify the complexity and workload associated with the traditional prioritization process. Additionally, it improves the accuracy and relevance of the user interface usability heuristic testing results. By prioritizing heuristics based on their importance as determined by the Usability Testing Leader—rather than merely depending on the number of items, scale, or heuristics—our approach ensures that evaluations focus on the most critical usability aspects from the start. The findings from this study highlight the importance of expert-driven evaluations for gaining a thorough understanding of heuristic UI assessment, offering a wider perspective than user-perception-based methods like the questionnaire approach. Our research contributes to advancing UI evaluation methodologies, offering an organized and effective framework for future usability testing endeavors.

Keywords: heuristic evaluation; usability testing; analytic hierarchy process; usability; algorithm efficiency; expert evaluation; human–computer interaction; heuristic evaluation; user interface

Citation: Talero-Sarmiento, L.; Gonzalez-Capdevila, M.; Granollers, A.; Lamos-Diaz, H.; Pistili-Rodrigues, K. Towards a Refined Heuristic Evaluation: Incorporating Hierarchical Analysis for Weighted Usability Assessment. *Big Data Cogn. Comput.* **2024**, *8*, 69. https://doi.org/10.3390/bdcc8060069

Academic Editors: Paweł Weichbroth, Jolanta Kowal and Mieczysław Lech Owoc

Received: 9 April 2024
Revised: 6 June 2024
Accepted: 7 June 2024
Published: 13 June 2024

1. Introduction

Usability is critical in the design and development of technology and software. It refers to the ease with which users can effectively, efficiently, and satisfactorily interact with a system or product to achieve their goals [1]. This concept is paramount in determining the success or failure of software applications and technological products. Its importance is emphasized by Giacomin, who states that user-centric designs lead to higher productivity, reduced errors, and enhanced user engagement [2]. A focus on usability ensures that products are intuitive and accessible, meeting the diverse needs of users [3] and enhancing

human performance [4]. Thus, integrating usability in technology fosters a positive user experience [5] and significantly influences user adoption and satisfaction [6,7].

Experts often try to quantify the usability of systems despite these challenges [8]; they implement heuristic evaluations and quantification approaches like the System Usability Scale (SUS) for usability testing [9], the Questionnaire for User Interaction Satisfaction (QUIS) for human–computer interfaces [10], the Software Usability Measurement Inventory (SUMI) for testing computer usability satisfaction [11], or the Post-study System Usability Questionnaire (PSSUQ) [12], among others. Heuristic evaluations involve experts examining an interface against established usability principles [13]. This qualitative method relies on the expertise of evaluators to identify usability issues. In contrast, mixed-method approaches quantify perceived ease of use from the end user's perspective [14]. They provide a numerical score to measure a product's usability. While heuristic evaluations offer in-depth, expert analysis, approaches such as SUS capture user feedback quantitatively [15], collectively benefiting a comprehensive understanding of usability [16].

Developing a unified usability strategy that combines qualitative (also known as inquiry) and quantitative or so-called testing-based approaches presents several challenges [17]. Technology developers need to balance expert-driven insights from heuristic evaluations with user-centric data from tools like usability measurement estimations, requiring an intricate understanding of both approaches [18]. Qualitative methods, rich in contextual information, can be subjective. At the same time, quantitative approaches, though objective, might not capture nuanced user experiences [19]. It is necessary to devise a sophisticated strategy that integrates these methodologies to produce a comprehensive usability score. This strategy should respect the strengths and limitations of each method, ensuring that the usability score reflects the expert analysis and user experience [20].

Usability benefits extend beyond merely evaluating technology adoption; they are instrumental for identifying characteristics that require improvement in software engineering and task prioritization within software development [21]. However, evaluating usability based on expert assessment is challenging due to the numerous components and potential for order effects, which can introduce bias [22]. These challenges highlight the need for methodologies that prioritize the order of evaluation to mitigate bias and enhance accuracy. Existing approaches often quantify usability based on expert heuristics evaluations. However, these methods can vary significantly in terms of scales, components, and the number of heuristics [23], making quantification dependent on these characteristics. This work proposes using the analytical hierarchical process [24] as a strategy to evaluate or estimate the relevant weights of the heuristics that do not depend on the instrument's characteristics. Our approach modifies the traditional application, focusing not on comparing alternatives but on identifying the relevance of each component. Furthermore, considering the extensive number of items per instrument during heuristic UI assessment, we propose a tailored algorithm for decreasing the number of comparisons required by leveraging the transitivity property. We designed this alternative for experts who find the number of comparisons cumbersome and prone to bias due to their exhaustiveness.

Given the dynamic nature of usability, which is strongly influenced by the specific lifecycle stage of software and the version in use, along with the complex and diverse strategies for fostering user–machine interactions [25], it is crucial to delineate the scope of this research, especially when there are ambiguous definitions and applications of usability testing. Considering the taxonomy proposed by Alonso-Ríos et al. [26], this document examines the need for an easily implemented and objectively assessed methodology for evaluating user interfaces (UIs) during these critical phases. Section 2 delves into the evolution and implications of expert approaches for quantifying UI usability. The methodology detailed in Section 3 incorporates the analytic hierarchy process to estimate weightings and explores a simulation approach for a practical application with actual data. We applied these methods to adapt to the evolving nature of software, ensuring that the heuristic UI assessment is both relevant and precise for each stage. The findings from this approach are aggregated and interpreted in Section 4, with a discussion on their practical and theoretical

significance provided in Section 5. Ultimately, Section 6 summarizes the principal findings and underscores the contributions of this research, which aims to refine the discourse in HCI by presenting a methodologically rigorous and empirically validated approach to UI assessment.

2. Background

The term usability was defined by ISO/CD 9241-11, where the usability of an interaction system is primarily evaluated based on its capacity to aid the user with accomplishing a task, thereby improving effectiveness, simplicity, and enjoyment, which are the three fundamental usability aspects [27]. Heuristic evaluation was first introduced by J. Nielsen and R. Molich in 1990 in their seminal work "Heuristic Evaluation of User Interfaces" [13], which is a method for assessing usability in user interfaces. It involves expert evaluators scrutinizing an interactive system's UI to gauge its quality of use. This assessment measures the extent to which the interface adheres to a predetermined set of usability guidelines or heuristics—hence, the name. This technique relies on a curated list of guidelines drawn from the collective expertise of the evaluators. Their experience enables them to identify usability issues or areas for improvement effectively. The method typically entails the following steps: evaluators individually complete questionnaires, documenting encountered problems; subsequently, they convene to discuss and consolidate their findings into a cohesive list. Throughout these discussions, evaluators prioritize the identified problems based on severity, frequency, and criticality.

Nielsen and Molich proposed ten heuristics to guide evaluations in their original work. This structured approach ensures a systematic and thorough assessment of the interface's usability, leading to actionable insights for optimization. Their findings about that method were various, and they opened multiple research lines in the following years. Among their conclusions, they highlighted the following:

1. Heuristic set: The heuristic set originally contained nine heuristics extracted from the work by Molich and Nielsen [13]. It serves as a way to categorize usability problems. However, it does not provide information about how to solve them.
2. Number of evaluators: The authors noticed that the number of evaluators was a critical factor, and they determined that an optimal number of them might be around three to five, and that more than ten evaluators might be unnecessary.
3. Evaluator biases: The answers from the evaluators are subject to their expertise, previous experience, and own judgment, providing potential limitations and biases in the results.

Several studies emerged to update and propose different sets of heuristics regarding the heuristic set. Nielsen introduced another heuristic to the initial set of nine, providing what is commonly known as "The 10 Nielsen's Heuristics". However, authors consider the initial set of heuristics too general [23], opening up new studies to improve the initial set to be more specific to the desired object of study. This led to the proposal of new sets of principles, such as Shneiderman's Eight Golden Rules [28], which emphasize usability guidelines for user interface design, Norman's Seven Principles [29], focusing on cognitive aspects of design, or Tognazzini's First Principles of Interaction Design [30], providing foundational principles for crafting engaging user experiences.

Performing heuristic evaluations offers numerous advantages beyond enhancing technology acceptance [31], including cost-effectiveness, as it requires minimal time and fewer users compared to traditional user testing [32,33]. Additionally, it demands less extensive planning, involves fewer people, and entails a streamlined analysis process. Moreover, heuristic evaluation is versatile and applicable across various stages of software development, including the planning, development, and post-release phases [34]. However, several disadvantages exist. Firstly, finding evaluators with sufficient expertise to provide high-quality feedback can be challenging [35,36]. Secondly, depending on the project stage, evaluators may struggle to grasp the full range of tasks applicable to the software [37,38]. Thirdly, the evaluation results may lack actionable suggestions for resolving identified

usability issues, potentially necessitating additional data collection [39]. Finally, the original scoring system introduced by Nielsen and Molich [13] has often perplexed evaluators due to its differentiation among severity, frequency, and criticality attributes; thus, heuristic methods without a rigid framework poorly support problem discovery [40]. This confusion has led many experts to predominantly focus on one attribute, typically criticality, highlighting the need for subsequent proposals to refine this aspect [41,42].

Numerous authors have contributed to the evolution of usability heuristics to explore aspects beyond user interfaces by proposing tailored sets specific to their use cases [43], prompting further inquiries and the development of methodologies to compile such data. Quiñones et al. [44] conducted a comprehensive systematic review of various usability heuristic proposals in the literature, elucidating diverse approaches to their creation. Their findings reveal a spectrum of methodologies, including considering existing heuristics, literature reviews, analyses of usability problems, incorporating design recommendations, interviews, and theoretical frameworks. From the analysis of over 70 papers, two main clusters emerged in heuristic research: one focused on developing domain-specific usability heuristics, and the other focused on processes and methodologies for their creation. In exploring the process of heuristic development, researchers have investigated the existence of a consensus on the most effective approach [45–50]. Hermawati and Lawson did another approach [51] where they analyzed more than 90 articles that used heuristic evaluation, and their findings were that less than 10% were acceptable and robust. They justified this as most of the studies did not perform validation, did not conduct a comparative justification between heuristics, did not quantitatively analyze the comparison results, and relied only on detailed textual descriptions.

Extracting actionable insights from qualitative feedback without numeric measurements necessitates effort from researchers to quantify or estimate usability attributes such as effectiveness, efficiency, and satisfaction [52]. Mitta proposed a methodology for quantifying expert usability using a linear multivariate function, with user perceptions and performance as independent variables [53]. This approach, illustrated with a practical example, yielded a usability score through linear normalization of experimental data. Delice and Güngör explored the quantification of attributes like severity, as defined by Nielsen and Molich [13], using the analytic hierarchy process in combination with heuristic evaluation for heuristic prioritization [54]. Their method involved expert website usability evaluations and ranking and identified problems using pairwise comparisons based on Saaty's scale [24]. Granollers investigated a combined approach for heuristic UI assessment, integrating a 15-principle heuristic set with specific questions and a 4-option rating scale for quantification, resulting in a usability percentage (UP) [55]. This method garnered attention as a notable proposal in the field [56]. Paz et al. proposed a specialist-oriented inspection technique for usability quantification, employing a 64-item checklist validated by Bonastre and Granollers [57,58]. Usability was quantified by averaging evaluators' responses, establishing the reliability of the assessment methodology.

In terms of enhancing the checklist approach for heuristic assessment, Kemp et al. [59] introduced a detailed checklist for each heuristic, with specific questions to assess the system, utilizing a 0 to 4 rating scale. This refinement aimed to improve the precision of assessments. However, numerous sub-heuristics or topics raised concerns about evaluator fatigue, a challenge initially addressed by Nielsen [32] and later by Granollers [55] through limiting the question quantity. Furthermore, the diversity in checklist formats—ranging from PDF and DOC to XLS—underscores the variation in support mechanisms employed across studies [60,61], reflecting the adaptability of heuristic evaluation methods to different technological contexts and evaluator preferences. While beneficial for tailored assessments, this adaptability necessitates careful consideration to maintain evaluator engagement and ensure the reliability of usability insights. Despite the diversity in instruments designed to quantify system usability, the discipline of psychometrics covers all approaches [62]. This connection underscores the necessity of grounding heuristic evaluation and survey design within robust psychological principles in HCI.

However, ensuring the best instrument for measuring concepts or getting the "gold standard" in psychometric research (understood as the highest level of methodological quality for survey design) is a paramount activity [63]. Different strategies arise in the literature to enhance the instrument's consistency and effectiveness [64]. Maintaining a uniform scale, measurement level, or end-point scale across all analyzed items or constructs helps to unveil the similarities between latent variables [65], especially for multivariate and correlational models [66]. Using the same number of items ensures that different test versions are consistent and can be compared across different administrations or populations, helping the generalizability of the results [67]. Face validity indicates the extent to which a test appears effective in terms of its stated aims [68]. Mathematically, aligning the number of items per construct enhances reliability [69] and reflects principles from item-response theory, emphasizing the significance of each question's contribution to the overall construct [70].

Some research does not follow a uniform scale, and experts have designed proper weight systems to accomplish that task [71–73]. The study by Gulzar et al. [74] presents multiple criteria weights used in the past, like mathematical programming, analytic network processes, linear weighting, and analytic hierarchy processes. Kamaldeep et al. [71] followed the "CRiteria Importance Through the Inter-criteria Correlation" (henceforth CRITIC) methodology to establish and find objective weights related to the criteria of their evaluation. Although they do not use a heuristic evaluation, they compared the relationships between properties of the individual criteria. In the same way, Muhammad et al. [72] used the "Fuzzy analytic hierarchy process" (henceforth FAHP) to create their methodology to compute global weights for usability factors. In that study, they do not specify that they followed a heuristic evaluation, but the methodology used refers to the range of usability factors that come from an expert evaluation. Iryanti et al. used a similar approach [73] with a fuzzy preference programming method known as "Inverse Trigonometric Fuzzy Preference Programming" (henceforth ITFPP) to evaluate a specific domain like e-learning through the arc-sine function.

3. Materials and Methods

This research introduces significant modifications to the Granollers heuristic UI assessment method to enhance its adaptability and relevance to diverse HCI contexts, including flexibility and adaptability to different tests for usability assessment. Traditionally, Granollers' method evaluates usability across 15 items with uniform significance, irrespective of the interface, user, or technology involved [55]. Our approach redefines the assessment process by introducing variable weights to these items, recognizing the original limitation. We propose this modification predicated on the understanding that certain heuristics may bear more significance than others during specific software development phases depending on the specific objectives of the Usability Testing Leader: for example, prioritizing those aspects more relevant in the software to be improved or organizing the heuristic UI assessment process based on prioritizing the most relevant aspect at the beginning of the evaluation to avoid bias due to exhaustiveness.

Considering all aspects or characteristics of UI assessment, particularly in heuristic evaluation, it becomes evident that a multicriteria decision making (MCDM) approach is indispensable. This necessity arises due to the complex and varied elements involved, such as group decision making, sensitivity to changes in initial matrix values, and integrating qualitative evaluation with quantitative assessment criteria. Despite the diversity of MCDM methodologies, selecting an appropriate method hinges significantly on researchers' and practitioners' specific requirements and assumptions. Munier highlights that a guideline for choosing a multicriteria method can be significantly beneficial [75]. This guideline assesses compatibility based on 54 potential characteristics, ensuring that the selected method aligns closely with the unique demands of a project, thereby optimizing the assessment of user interface heuristics pragmatically and efficiently. Table 1 shows the characteristics related to heuristic UI assessment. We identified 16 characteristics to consider, with the

analytic hierarchy process (AHP) and the technique for order of preference by similarity to ideal solution (TOPSIS) being the most suitable methodologies for this research scope. Considering the relevance of decomposing complex problems into a set of simple sub-problems [76], we employed AHP to systematically assign weights to the heuristic elements. We chose AHP and not TOPSIS because the Usability Testing Leader does not have a reference value, so we do not try to classify alternatives with an ideal solution (indeed, we do not have alternatives). Additionally, the researchers are familiar with AHP, and the present research modifies the traditional AHP to make it more suitable for the research aims (this section describes all these assumptions and modifications).

Table 1. Multicriteria method comparison for heuristic UI assessment based on Munier's approach [75].

		Method									
ID	**Characteristic**	**SAW**	**AHP**	**TOP**	**VIK**	**PRO**	**MOO**	**ELE**	**ANP**	**LPr**	**SIM**
1	Simple scenario	1	1	1	1	1	1	1	1	1	1
8	Several DMs (group decision making)	0	1	0	0	0	0	0	0	0	1
9	Ease of changing the initial matrix values	1	0	1	1	1	1	1	0	1	1
10	Large project involving consultations with people	1	0	1	1	1	1	1	0	1	1
11	Linguistic initial matrix	0	1	0	0	0	0	0	1	0	0
12	Qualitative criteria	1	1	1	1	1	1	1	1	1	1
14	Using a particular normalization procedure	0	1	1	0	0	1	0	1	0	0
16	Independent alternatives	0	1	0	0	0	0	0	1	0	0
19	Many criteria	1	0	1	1	1	1	1	0	1	1
20	Independent criteria (compensatory methods)	1	1	1	0	0	0	0	1	0	0
22	Necessity of knowing criteria's validity range	0	1	1	0	1	0	1	1	1	1
46	Necessity to evaluate criteria's relative importance	1	1	1	1	1	1	1	1	0	1
47	Want to use subjective weights	1	1	1	1	1	1	1	1	0	0
50	Sensitivity analysis (SA) with weights	1	1	1	1	1	1	1	1	0	0
54	Not theoretically complex	1	1	1	0	1	0	0	0	0	0
	Requirement	16	16	16	16	16	16	16	16	16	16
	Match	10	12	12	8	10	9	9	10	6	8
	Best (minimum gap)	6	4	4	8	6	7	7	6	10	8

SAW: Simple Additive Weighting, **AHP**: Analytic Hierarchy Process, **TOP**: Technique for Order of Preference by Similarity to Ideal Solution, **VIK**: VlseKriterijumska Optimizacija I Kompromisno Resenje (VIKOR), **PRO**: Preference Ranking Organization Method for Enrichment Evaluations (PROMETHEE), **MOO**: Multi-Objective Optimization, **ELE**: ELECTRE (Elimination Et Choice Translating Reality), **ANP**: Analytic Network Process, **LPr**: Linear Programming, and **SIM**: Simulation

AHP, a structured technique for organizing and analyzing complex decisions, is based on mathematics and psychology. It involves decomposing a problem into a sub-problem hierarchy, which is then analyzed independently. A correct AHP implementation is a challenge due to the ambiguous formulation of research questions, potentially resulting in varied treatment hierarchies that can undermine the robustness of its outcomes [77]. Additionally, the method's reliance on hierarchical structuring of objectives can inadvertently skew the distribution of weights, affecting the overall balance and fairness of the analysis [78]. The selection of units for analysis and the strategies employed in modeling within AHP frameworks are critical, as they substantially influence the conclusions' validity [79] as well as the emergent relationships between criteria, subcriteria, and alternatives, which

generate dependence and feedback between components [80]. Thus, when implementing AHP, it is important to consider pairwise comparisons, judgment scales, derivation methods, consistency indices, incomplete matrices, synthesis of the weights, sensitivity analysis, and group decisions [81].

In this study, we applied AHP to quantify the relative importance of each heuristic item. This methodology involved generating pairwise comparisons and deriving weights through standardized equations, ensuring a robust and replicable approach. Our AHP implementation requires assumptions such as the independence of criteria, treating each as a distinct usability aspect for evaluation. The questions in our test functioned as guidelines, facilitating both the quantification of value and the collection of expert feedback. Unlike typical applications of AHP, which focus on evaluating alternatives, our approach targets the assessment of system usability and aims to establish a baseline indicator for a UI's usability component improvements. Furthermore, the method for generating weights enhances the sensitivity of the heuristic UI quantification process. It does not depend on the number of components or the scale of components within each heuristic under consideration. This adaptation ensures that our heuristic UI assessment is comprehensive and finely attuned to subtle variations in system interaction, providing valuable insights for enhancing user experience.

3.1. Heuristic Instrument

The heuristic evaluation framework by Toni Granollers is an extension and adaptation of principles from pioneers like Nielsen and Tognazzini [55] and aims to provide a comprehensive toolkit for usability assessment in HCI. The framework comprises 15 heuristics, each with specific questions designed to probe various aspects of user interaction and interface design. These heuristics cover areas from the visibility of system states and error management to aesthetic design and efficiency of use. The questions are quantified, and their answers are categorized to reflect the degree of the system's alignment with these heuristics, ranging from full compliance to non-applicability. The number of questions per heuristic varies, reflecting the depth of investigation into each area. The usability value derived from these evaluations is a calculation expressed as a percentage, standardized between 0% to 100%, indicating the extent of adherence to usability standards. A color-coding system is employed to communicate this value visually: green represents high usability, yellow indicates moderate usability, red suggests poor usability, and white denotes non-applicability or non-issues. The full list of heuristics and the number of associated questions and descriptions can be meticulously detailed, ensuring a robust and well-rounded evaluation instrument.

1. Visibility and system state (five questions): Focuses on ensuring that users are always aware of what the system is doing and their position within it.
2. Connection with the real world (four questions): Prioritizes using familiar language, metaphors, and concepts, aligning the system with real-world analogs.
3. User control and freedom (three questions): Emphasizes allowing users to navigate and undo actions easily.
4. Consistency and standards (six questions): Ensures uniformity in the interface, with consistent actions and standards across different elements.
5. Recognition rather than memory (five questions): Aims to design systems that minimize the need for remembering information, enhancing user learning and anticipation.
6. Flexibility and efficiency (six questions): Focuses on providing shortcuts and efficient paths for experienced users while remaining accessible to novices.
7. Help users recognize, diagnose, and recover from errors (four questions): Focuses on designing systems that provide clear, understandable error messages, aiding users in recognizing and rectifying issues efficiently.
8. Error prevention (three questions): Involves designing systems to prevent errors before they occur.

9. Aesthetic and minimalist design (four questions): Encourages visually appealing designs and minimal unnecessary elements.
10. Help and documentation (five questions): Stresses the importance of accessible, clear help and documentation for users.
11. Save the state and protect the work (three questions): Addresses the need to save user progress and protect against data loss.
12. Color and readability (four questions): Ensures that text is readable with appropriate color contrast and size.
13. Autonomy (three questions): Allows users to make personal choices and customizations in the system.
14. Defaults (three questions): Focuses on providing sensible default settings while allowing users to revert to these defaults when needed.
15. Latency reduction (two questions): Aims to minimize delays and provide feedback during processes that require time.

Granollers' heuristic framework analysis extends beyond the traditional limits of user interface review, incorporating a broader and integrated approach. This method thoroughly evaluates the three fundamental pillars of usability as outlined by ISO 9241-11: effectiveness, efficiency, and satisfaction. This multidimensional assessment guarantees a comprehensive examination of usability, considering not only the precision and completeness with which users can achieve their intended goals but also analyzing the use of resources and the overall subjective experience of the user. The first pillar, effectiveness, addresses how accurately and completely users can perform their tasks and is influenced heavily by heuristics such as system status visibility, real-world connections, and error prevention. These guidelines help clarify the system's operations for the user, enabling a more straightforward and accurate interaction with the system.

Efficiency, the second pillar, concentrates on minimizing the resources required to achieve goals. Key heuristics are vital here, including user control and freedom, flexibility, and efficiency. By enabling users to correct errors or navigate efficiently, these heuristics significantly reduce the time and effort needed for task completion, thus enhancing the overall system efficiency. Lastly, the pillar of satisfaction focuses on the comfort and positive experiences users derive from interacting with the system. Heuristics like aesthetic and minimalist design, effective help and documentation, and user autonomy are crucial. They ensure the interface is functional, enjoyable, and intuitive, promoting increased user engagement and satisfaction. Table 2 illustrates the relationship between each heuristic and the usability components defined by the ISO standards.

Table 2. Granollers' heuristics mapped to ISO 9241-11 usability components.

Heuristic	Effectiveness	Efficiency	Satisfaction
Visibility and system state	Enhances task accuracy by informing users of system status.	Reduces time spent understanding system state.	Increases user confidence by providing clarity.
Connection with the real world	Improves task understanding through familiar concepts.	Speeds up task performance by reducing cognitive load.	Enhances comfort using familiar metaphors.
User control and freedom	Improves task accuracy by allowing error correction.	Reduces effort through efficient task recovery.	Enhances user satisfaction by providing control options.
Consistency and standards	Maintains task accuracy through uniform interactions.	Increases efficiency by reducing learning time.	Provides a predictable and satisfying user experience.

Table 2. *Cont.*

Heuristic	Effectiveness	Efficiency	Satisfaction
Recognition rather than memory	Minimizes reliance on memory, improving task accuracy.	Enhances efficiency by providing cues and reminders.	Reduces user frustration, enhancing satisfaction.
Flexibility and efficiency	Supports accurate task performance for all user levels.	Provides experienced users with shortcuts to enhance efficiency.	Accommodates diverse user skills, increasing satisfaction.
Help users recognize, diagnose, and recover from errors	Improves accuracy through effective error management.	Decreases time and effort in error recovery.	Provides a safety net that enhances user satisfaction.
Error prevention	Prevents potential errors, improving accuracy.	Enhances efficiency by reducing error correction needs.	Improves user experience by minimizing disruptions.
Aesthetic and minimalist design	Focuses user attention on essential tasks.	Streamlines interactions, improving efficiency.	Delivers an appealing environment that enhances satisfaction.
Help and documentation	Assists users with completing tasks correctly.	Decreases time spent seeking assistance, boosting efficiency.	Increases user satisfaction with accessible support.
Save the state and protect the work	Preserves user progress, enhancing accuracy.	Reduces time redoing tasks, improving efficiency.	Protects user effort, increasing satisfaction.
Color and readability	Improves accuracy by enhancing content clarity.	Lowers effort required for content comprehension.	Provides a visually accessible interface that increases satisfaction.
Autonomy	Allows personalization, improving task relevance and accuracy.	Enhances efficiency by tailoring the system to user preferences.	Enhances satisfaction through personalized interaction.
Defaults	Provides reliable starting points, enhancing task accuracy.	Reduces initial setup time, improving efficiency.	Increases satisfaction with dependable system behaviors.
Latency reduction	Provides immediate feedback, ensuring task accuracy.	Reduces waiting times, significantly improving efficiency.	Boosts satisfaction with a responsive system.

The methodology we adopted in this study introduces a weighted heuristic evaluation approach for usability assessment. This approach refines the original method proposed by Granollers. The Granollers approach estimates a usability score or usability percentage (UP) considering Equation (1). Here, n_i is the number of questions in the ith heuristic, and value$_{ij}$ is the value assigned to the jth question of the ith heuristic. The values are assigned based on a predefined response scale: "Yes" = 1.0, "Neither Yes, nor No" = 0.5, and "No" = 0.0. NA_i, NP_i, and WR_i are the counts of non-quantitative responses "Not Applicable", "Not a Problem", and "Impossible to Check" for the ith heuristic, respectively.

$$UP = \frac{\sum_{i=1}^{15} \sum_{j=1}^{n_i} \text{value}_{ij}}{\sum_{i=1}^{15} n_i - (NA_i + NP_i + WR_i)} \quad (1)$$

We now calculate the usability percentage using Formula (2) to relate the usability percentage calculation with each heuristic's relevance. In this case, we add the parameter w_i as the weight or relevance of each heuristic during the assessment (3). Originally, this parameter represented the proportion of questions per heuristic to the total available questions, meaning that regardless of the systems or intention for assessment, all the heuristics had a predefined relevance based on the number of questions. In addition, analyzing each heuristic's component alone requires guaranteeing at least one value in each heuristic (see Equation (4)). In this sense, the original approach encounters challenges: it may not align with the relevance of heuristics in specific systems. For example, Table 3 shows the relative relevance of each heuristic based on the number of questions or analysis over the general software evaluation; the w_i in this table corresponds with the number of questions in each heuristic divided by the total number of questions. Moreover, this approach can lead to indeterminate calculations when the number of non-applicable responses equals the total questions in a heuristic.

We propose a revised approach whereby w_i is determined based on the heuristic's relevance to the specific system under evaluation rather than the mere quantity of questions to address these issues. This adjustment ensures a more accurate usability score, recognizing the varied importance of heuristics and preventing misleading evaluations. For instance, a low value in a low-weighted heuristic does not generate an alarm. Still, a low value in a high-weighted heuristic encourages the software development team to prioritize improvements. This more granular approach facilitates the identification of critical usability issues, enabling developers to allocate resources effectively during software improvement sessions. It allows prioritization of aspects that significantly impact user satisfaction and operational efficiency, optimizing the development process and enhancing end-user engagement with the software.

$$UP = \sum_{i=1}^{15} w_i Heuristic_i \tag{2}$$

$$\sum_{i=1}^{15} w_i = 1 \tag{3}$$

$$Heuristic_i = \begin{cases} \frac{\sum_{j=1}^{n_i} value_{ij}}{n_i - (NA_i + NP_i + WR_i)}, & \text{if } n_i - (NA_i + NP_i + WR_i) \neq 0 \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

Table 3. Equivalent weights in the Granollers approach.

	Heuristic														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
w_i[%]	8.33	6.67	5.00	10.00	8.33	10.00	6.67	5.00	6.67	8.33	5.00	6.67	5.00	5.00	3.33

3.2. Analytical Hierarchical Process

In the domain of heuristic UI assessment, usability experts can use the analytic hierarchy process to determine the relative importance of different usability heuristics. This method excels when experts apply it to grade and prioritize criteria based on their professional judgment due to its simplicity, efficiency, and safety [82]. AHP is widely used in multiple-criteria decision making, with applications in planning, selecting alternatives, resource allocation, conflict resolution, and optimization [83]. AHP's structured approach breaks down the evaluation into a hierarchy, from the overarching goal of obtaining a usability score to application contexts such as apps, websites, or software. Experts must construct a pairwise comparison matrix for each heuristic criterion, with each element a_{ij} indicating the relative importance of the ith heuristic over the jth using the Saaty scale [24]. This matrix is reciprocal, where $a_{ij} = \frac{1}{a_{ji}}$, generating the matrix A that relates the rela-

tive weight w_i or relevance of the ith characteristic as see in Equation (5). The principal eigenvector w corresponding to the largest eigenvalue λ_{max} is computed to calculate the weights, which determine the priorities. A consistency check ensures the reliability of these comparisons, using a consistency ratio (hereafter CR) to compare the matrix's consistency index (henceforth CI) against an ideal index derived from a random matrix (see (7)). If the CR is below 0.1, the weights are consistent; this concept is relevant during comparison processes because the utility theory requires a consistency comparison [84]. The resulting eigenvector provides the weighted priorities for the usability heuristics, reflecting the aggregated expert opinion on the importance of each usability heuristic.

$$A = \begin{bmatrix} 1 & a_{12} & \cdots & a_{1n} \\ \frac{1}{a_{12}} & 1 & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{1}{a_{1n}} & \frac{1}{a_{2n}} & \cdots & 1 \end{bmatrix} = \begin{bmatrix} 1 & \frac{w_1}{w_2} & \cdots & \frac{w_1}{w_n} \\ \frac{w_2}{w_1} & 1 & \cdots & \frac{w_2}{w_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{w_n}{w_1} & \frac{w_n}{w_2} & \cdots & 1 \end{bmatrix}, w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix} \tag{5}$$

$$A \times w = \lambda_{max} \times w \tag{6}$$

$$CR = \frac{CI}{RI}, CI = \frac{\lambda_{max} - n}{n-1}, RI = \frac{1.98 \times (n-2)}{n} \tag{7}$$

We seamlessly integrated the hierarchical analysis process into our modified heuristic evaluation method (Appendix A). The Usability Testing Leader, an expert in usability testing, determines the relevance of each heuristic through a systematic pairwise analysis. This process occurs independently of the evaluators tasked with assessing the software, thus maintaining an unbiased approach. Unaware of the Usability Testing Leader's weighting decisions, the evaluators focus solely on their usability assessment tasks. This dual-process approach ensures the evaluations are objective and reflect the software's inherent usability features. To implement this modified evaluation method, we chose Python for its versatility and robustness, particularly for the computation of the heuristic weights. The evaluators, on the other hand, conducted their assessments using tools that are universally accessible and user-friendly, such as Excel or online forms. This choice of software facilitated ease of use and widespread applicability. The last step is multiplying the evaluators' scores by the respective weights in Python. This approach allows for a nuanced analysis that considers the individual scores and the adjusted significance of each heuristic element.

- Input: Pairwise comparison matrices for criteria and alternatives.
- Output: Priority vector (weights) for criteria and alternatives.
- For each pairwise comparison matrix:
  - Normalize the matrix by column.
  - Compute the principal eigenvector to determine weights.
  - Calculate the consistency ratio.
  - If CR is less than 0.1:
    - Accept the weights.
  - Else:
    - Re-evaluate comparisons.
- Show the weights for final decision making.

3.3. Simulation

We designed a simulation framework to evaluate the performance of our modified AHP algorithm over 10,000 iterations. The aim was to analyze the algorithm's ability to enhance consistency and reduce the number of comparisons using the transitivity property. In each simulation, a virtual decision maker initiates the process with the first criterion and randomly selects a value from the Saaty scale. The decision maker then chooses a set of criteria for comparison, ranging from one to the entire remaining set.

Upon selecting multiple criteria, the algorithm applies the transitivity property to reduce the number of direct comparisons needed, thereby excluding those criteria from future selections. The simulation repeats this process until all necessary comparisons end. This methodology aims to understand the algorithm's efficiency in reducing comparisons and improving consistency in decision-making scenarios. Additionally, the experiment records the computing time, the number of comparisons made, and the consistency index for each simulation. The primary objectives are to assess the algorithm's impact on reducing the number of necessary comparisons and improving the consistency of the pairwise comparison matrix.

3.4. Data Acquisition

Data for this study were meticulously gathered through a series of structured heuristic UI assessments, this time within the controlled environment of a European usability research center in 2021. The evaluators, seven engineers who had completed doctoral studies in engineering and informatics and had extensive experience in heuristic UI assessment, had previously worked with the Granollers test. These evaluators offered their services voluntarily through an academic agreement, and we respected good research practices through a consent form. Each evaluator operated independently, without interaction or knowledge of the contributions of other evaluators. This independence ensured that the assessment remained unbiased. Additionally, the evaluators were unaware of the weights assigned to their evaluations, minimizing potential biases (i.e., blind review). Their expertise brought depth to the analysis, providing a professional perspective on the usability of a sophisticated web application under examination.

Significantly, the Usability Testing Leader, who took part in the analytical assessment, directed the process while setting weights according to their expert understanding of the software's usability needs. As a result of this pragmatic approach, the evaluations offered theoretical insights and practical implications for software development. Regarding data transparency, our research subscribes to an open-data policy, with all related materials, methodologies, and datasets available for replication and further investigation. Researchers and practitioners interested in these data can access the RUXAILAB (Remote User eXperience Artificial Intelligence LAB. https://github.com/ruxailab. Last access: 6 June 2024), a remote usability lab based on artificial intelligence to perform usability testing and experiments and host the datasets. The Usability Testing Leader and evaluators confined their activities to the typical scope of an educational environment. Since the research focused on the evaluation process rather than the participants, and the evaluators were also researchers benefiting from the activity in their corresponding research, there was no need for ethical approval, aligning with established ethical research standards when the investigation involves no human beings.

4. Results

4.1. Algorithm for Pairwise Comparison

The Python script introduces a tailored algorithm for decision making using the AHP to evaluate complex scenarios with multiple criteria, like selecting heuristics. Despite its apparent quadratic worst-case complexity, the algorithm significantly aims to streamline the decision-making process. This efficiency stems from a strategic approach to collecting user inputs for pairwise comparisons. Traditional AHP requires $N * (N-1)/2$ comparisons (with N as the number of criteria), but this algorithm cleverly reduces this number. It directs the decision maker to evaluate the relative importance of one criterion against others. For example, it might ask, "Select an option for comparisons involving Heuristic one (Cost) against the remaining criteria". When a user chooses a value from Saaty's scale, like "Equal Importance", the algorithm asks which criteria are of this importance level compared to cost.

The brilliance of this method is in its use of the transitive nature of comparisons [85]. When a user categorizes multiple criteria as equally important to a specific criterion, the

algorithm automatically applies the same level to those grouped criteria, thus increasing the matrix's consistency as an effort to avoid posterior comparison corrections. This smart approach can significantly reduce the necessary number of comparisons. In a practical scenario, for a set like Granollers' heuristics with 105 pairwise comparisons, this reduction could bring the number down to as low as one if all heuristics are considered equally important. This decrease transforms what could be a cumbersome and lengthy task into a far more manageable one, enhancing the algorithm's effectiveness in situations with a large set of criteria (a common challenge in those AHPs with a large number of criteria [75]).

We conducted a simulation analysis to evaluate how varying the group size m influences the reduction in the number of necessary comparisons when employing the pairwise comparison algorithm for decision making using the AHP. We adjusted the number of criteria N and the group size m across a broad range, from 2 to 30 for N and from 1 to N for m, to understand their impact on the efficiency of the decision-making process. Figure 1 showcases the number of comparisons the algorithm requires for each combination of N and m. In this model, the algorithm selects a random comparison value between 1 and 9 for each group to ensure a sufficient range of available numbers during the simulation. This approach circumvents limitations that might arise from the more restricted Saaty scale, which typically uses integer values and may not provide enough gradations for larger N values, such as 30 with $m = 1$, where 29 distinct values would be necessary. Figure 1 highlights how flexibility in scale values can aid with accommodating the extensive variability in group sizes and criteria numbers, potentially optimizing the process by reducing the overall number of comparisons needed for large-scale decision-making scenarios.

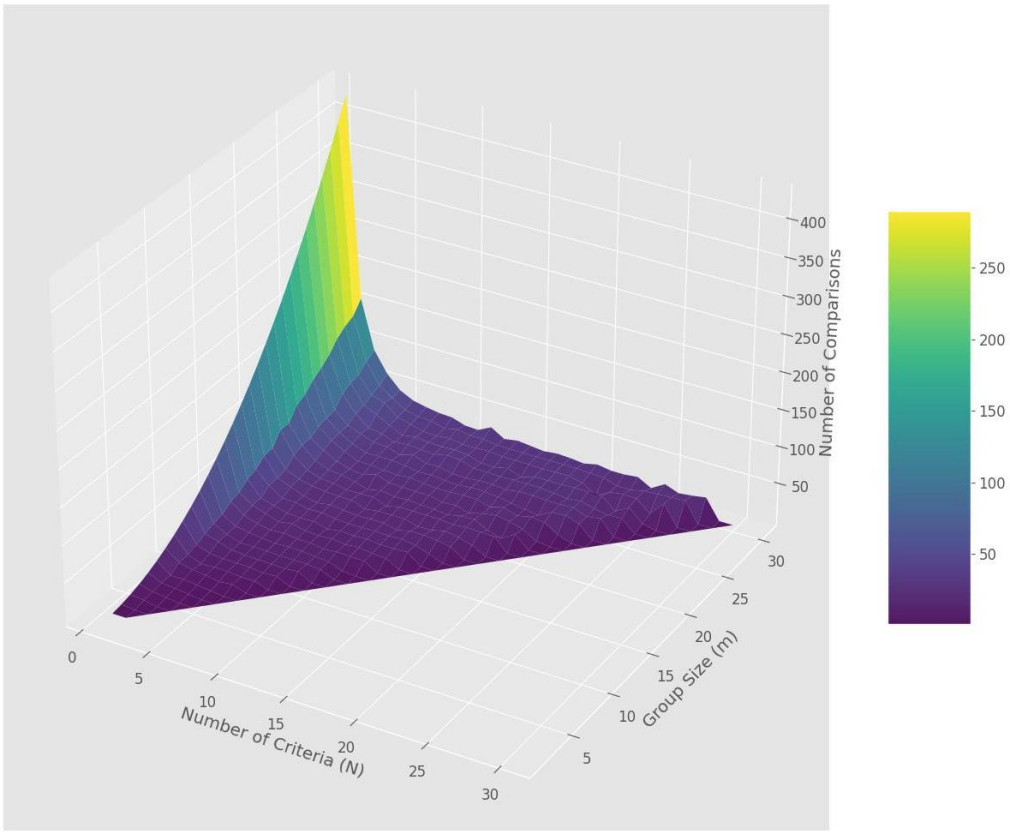

Figure 1. Meshgrid of number of comparisons by N and m.

Based on the results of Figure 1, we can assess the number of comparisons using the equation $\frac{N(N-1)}{2m}$, which estimates the expected total number of comparisons in an AHP matrix. This estimate assumes that m, the average number of grouped selections per criterion, enables efficient coverage of all necessary comparisons, distributing the decision-

making burden across fewer manual inputs. Each selection combines multiple criteria, assuming they share the same relative importance, thus amplifying the impact of each user interaction. However, in practical scenarios, m varies significantly based on the Usability Testing Leader's judgment and the specific context of the decision-making process. The Usability Testing Leader (UTL) must determine which criteria are similar enough to be grouped, which is a decision shaped by their knowledge, experience, and the nuances of the problem addressed. Consequently, the actual number of interactions—and, therefore, the efficiency of the grouping—can fluctuate. Some criteria may not align neatly with others, necessitating more distinct comparisons, while others may easily grouped, decreasing the number of necessary interactions.

Algorithm Performance

In this study, we ran 10,000 simulations using our algorithm with randomly assigned comparison values. We focused on evaluating our assignation method's consistency ratio compared to theoretical expectations of random assignments. Our results in Figure 2 reveal a notable and expected trend: the consistency ratio improves with fewer assignments. This pattern underscores the robustness of our approach, especially considering a random assignment methodology. Furthermore, our algorithm's efficiency is evident in its requirement of only a maximum of 38 comparisons: a significant reduction from the 105 comparisons necessary for a complete pairwise evaluation. This performance is also clear in computation time, with a maximum of 0.14 s, an average of 0.022 s, and a standard deviation of 0.005 s. This improved efficiency does not compromise the accuracy of the evaluations but assists the decision maker with creating a safe comparison, avoiding a long and cumbersome experience. This efficacy is evidenced by generating 858 cases that achieved satisfactory consistency (values below the threshold of 0.1).

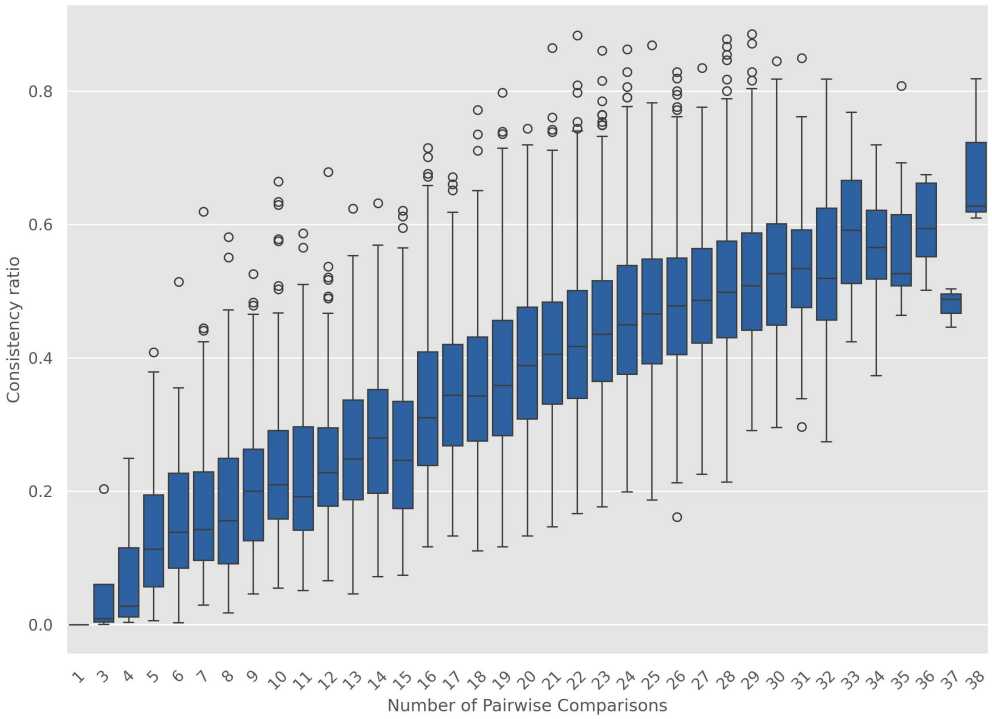

Figure 2. Consistency ratio boxplot by number of pairwise comparisons.

An intriguing aspect of our findings is the role of single comparisons in achieving higher consistency. The algorithm's effectiveness is most pronounced when it employs just one comparison, using the transitivity property to infer additional comparisons. Figure 3 depicts that instances with only one comparison account for 6.83% of all cases. Our analysis, detailed in Figure 4, also shows that the algorithm maintains consistency in

scenarios with up to 15 comparisons. Despite the lower overall consistency rate (under 9% of the 10,000 simulations), our method significantly outperforms random assignment, with a maximum consistency ratio of approximately 1.59. In comparison, our approach achieved a maximum consistency ratio of 0.88 and an average of 0.39, highlighting its reliability and potential application in heuristic UI assessment within the HCI field with a high-dimension set of criteria.

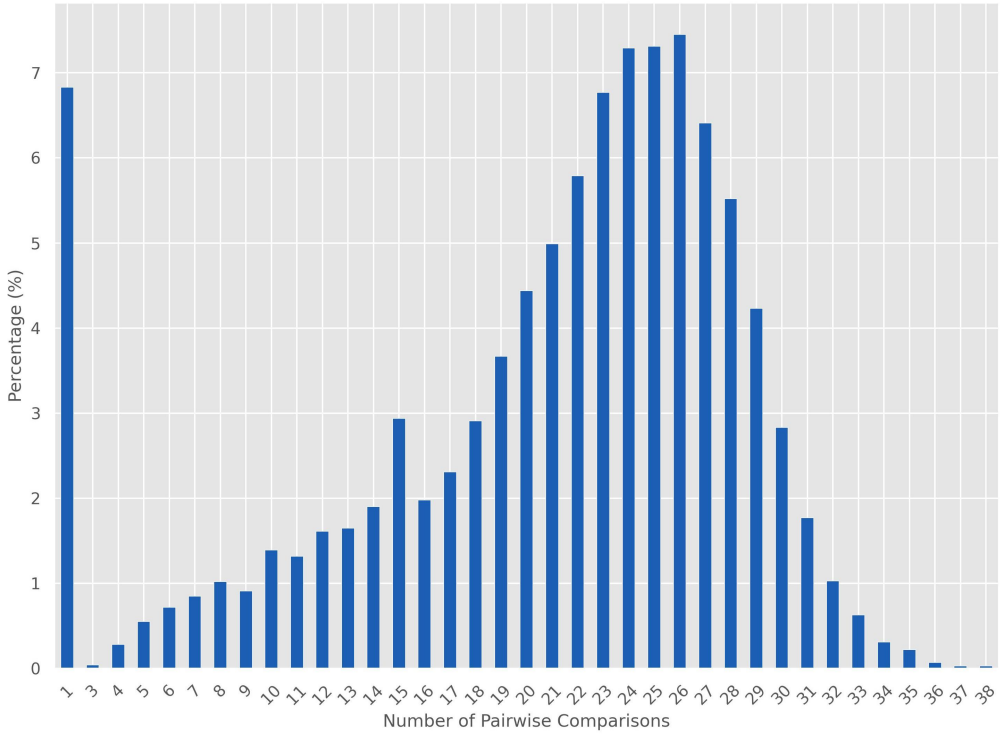

Figure 3. Frequency of pairwise comparisons.

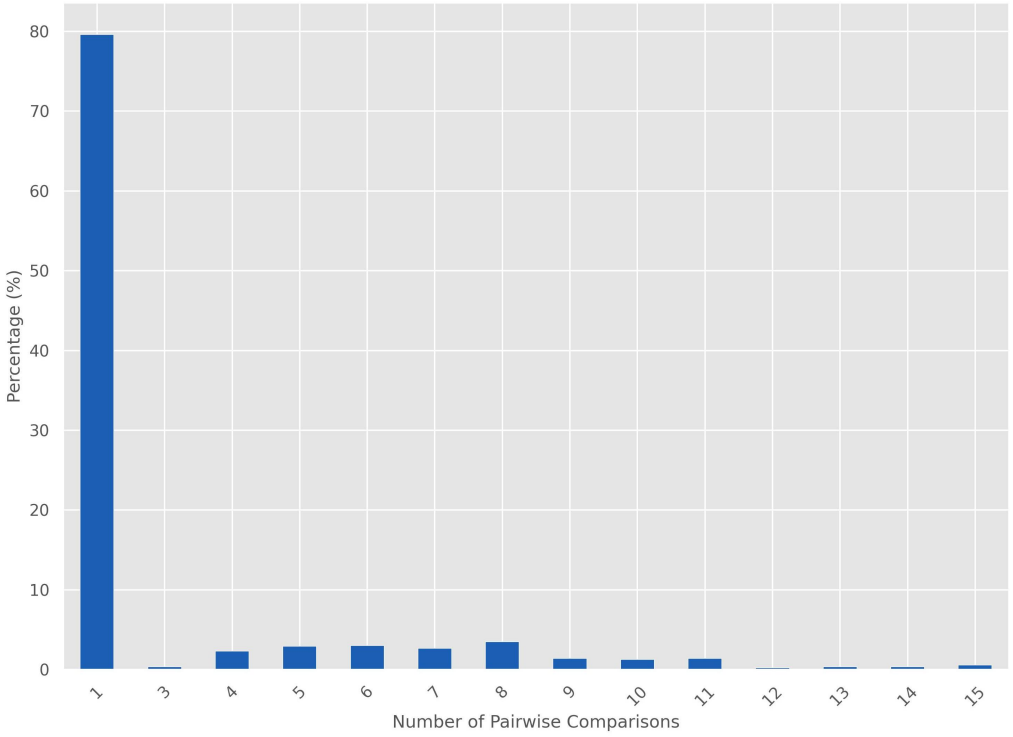

Figure 4. Frequency distribution of the number of pairwise comparisons with acceptable consistency.

4.2. Weighted Heuristic Application

After testing the algorithm, we applied the AHP to the UTL to streamline the application process. Our findings demonstrate a more streamlined and consistent weighting process. The UTL adeptly sets weights based on a discerning evaluation of each heuristic's comparative relevance for a specific website. We significantly reduced the required comparisons—from the exhaustive 105 to a more manageable 30—thereby simplifying the evaluative experience. (Table 4 shows the final results.) Considering the equation $\frac{N(N-1)}{2m}$ for estimating the total number of comparisons, the average group size during this experiment was $m = 2.98$, while the expected number with $m = 3$ was 35, indicating a successful approach. We provide detailed comparisons to generate the weights as follows:

- H1 Comparisons
  - H1 vs. H5: H1 and H5 are considered to have equal importance.
  - H1 vs. H2, H4, H9, H12, H13: H1 is moderately more important than H2, H4, H9, H12, and H13.
  - H1 vs. H3, H6, H7, H8: H1 is strongly more important than H3, H6, H7, and H8.
  - H1 vs. H10, H11, H14, H15: H1 is very strongly more important than H10, H11, H14, and H15.
- H2 Comparisons
  - H2 vs. H5: H2 is moderately less important than H5.
  - H2 vs. H9, H12, H13: H2 has equal importance compared to H9, H12, and H13.
  - H2 vs. H3, H4, H6, H7, H8: H2 is moderately more important than H3, H4, H6, H7, and H8.
  - H2 vs. H10, H11, H14, H15: H2 is strongly more important than H10, H11, H14, and H15.
- H3 Comparisons
  - H3 vs. H5: H3 is very strongly less important than H5.
  - H3 vs. H4, H9, H12, H13: H3 has equal importance compared to H4, H9, H12, and H13.
  - H3 vs. H6, H7, H8: H3 is moderately more important than H6, H7, and H8.
  - H3 vs. H10, H11, H14, H15: H3 is strongly more important than H10, H11, H14, and H15.
- H4 Comparisons
  - H4 vs. H5: H4 is moderately less important than H5.
  - H4 vs. H6, H7, H8: H4 has equal importance compared to H6, H7, and H8.
  - H4 vs. H10, H11, H14, H15: H4 is moderately more important than H10, H11, H14, and H15.
- H5 Comparisons
  - H5 vs. H9, H12, H13: H5 is moderately more important than H9, H12, and H13.
  - H5 vs. H6, H7, H8: H5 is strongly more important than H6, H7, and H8.
  - H5 vs. H10, H11, H14, H15: H5 is very strongly more important than H10, H11, H14, and H15.
- H6 Comparisons
  - H6 vs. H9, H12, H13: H6 is moderately less important than H9, H12, and H13.
  - H6 vs. H10, H11, H14, H15: H6 is strongly more important than H10, H11, H14, and H15.
- H7 Comparisons
  - H7 vs. H9, H12, H13: H7 is moderately less important than H9, H12, and H13.
  - H7 vs. H10, H11, H14, H15: H7 is strongly more important than H10, H11, H14, and H15.
- H8 Comparisons

  - H8 vs. H9, H12, H13: H8 is moderately less important than H9, H12, and H13.
  - H8 vs. H10, H11, H14, H15: H8 is strongly more important than H10, H11, H14, and H15.
- H9 Comparisons
  - H9 vs. H10, H11, H14, H15: H9 is strongly more important than H10, H11, H14, and H15.
- H10 Comparisons
  - H10 vs. H11, H14, H15: H10 has equal importance compared to H11, H14, and H15.
  - H10 vs. H12, H13: H10 is strongly less important than H12 and H13.
- H11 Comparisons
  - H11 vs. H12, H13: H11 is strongly less important than H12 and H13.
- H12 Comparisons
  - H12 vs. H14, H15: H12 is strongly more important than H14 and H15.
- H13 Comparisons
  - H13 vs. H14, H15: H13 is strongly more important than H14 and H15.

The AHP analysis produced promising results, indicating a max eigenvalue of 15.97 and a set of normalized weights spanning a diverse range (see Table 5). The Usability Testing Leader's expert judgment determined the weights, which indicate varying degrees of importance for each heuristic, ranging from approximately 18.06% for the most significant to 1.45% for the least. This methodology provides a nuanced view of heuristic relevance without dependence on the number of components in each heuristic. The consistency ratio, which measures the reliability of pairwise comparisons, is 0.044, which is well within the acceptable threshold, thus confirming the assessment's consistency.

Table 4. Comparison matrix using the algorithm to enhance consistency.

	Heuristic														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	1.00	3.00	5.00	3.00	1.00	5.00	5.00	5.00	3.00	7.00	7.00	3.00	3.00	7.00	7.00
2	0.33	1.00	3.00	3.00	0.33	3.00	3.00	3.00	1.00	5.00	5.00	1.00	1.00	5.00	5.00
3	0.2	0.33	1.00	1.00	0.14	3.00	3.00	3.00	1.00	5.00	5.00	1.00	1.00	5.00	5.00
4	0.33	0.33	1.00	1.00	0.33	1.00	1.00	1.00	1.00	3.00	3.00	1.00	1.00	3.00	3.00
5	1.00	3.00	7.00	3.00	1.00	5.00	5.00	5.00	3.00	7.00	7.00	3.00	3.00	7.00	7.00
6	0.2	0.33	0.33	1.00	0.20	1.00	1.00	1.00	0.33	5.00	5.00	0.33	0.33	5.00	5.00
7	0.2	0.33	0.33	1.00	0.20	1.00	1.00	1.00	0.33	5.00	5.00	0.33	0.33	5.00	5.00
8	0.2	0.33	0.33	1.00	0.20	1.00	1.00	1.00	0.33	5.00	5.00	0.33	0.33	5.00	5.00
9	0.33	1.00	1.00	1.00	0.33	3.00	3.00	3.00	1.00	5.00	5.00	1.00	1.00	5.00	5.00
10	0.14	0.20	0.20	0.33	0.14	0.20	0.20	0.20	0.20	1.00	1.00	0.20	0.20	1.00	1.00
11	0.14	0.20	0.20	0.33	0.14	0.20	0.20	0.20	0.20	1.00	1.00	0.20	0.20	1.00	1.00
12	0.33	1.00	1.00	1.00	0.33	3.00	3.00	3.00	1.00	5.00	5.00	1.00	1.00	5.00	5.00
13	0.33	1.00	1.00	1.00	0.33	3.00	3.00	3.00	1.00	5.00	5.00	1.00	1.00	5.00	5.00
14	0.14	0.20	0.20	0.33	0.14	0.20	0.20	0.20	0.20	1.00	1.00	0.20	0.20	1.00	1.00
15	0.14	0.20	0.20	0.33	0.14	0.20	0.20	0.20	0.20	1.00	1.00	0.20	0.20	1.00	1.00

Table 5. Weights obtained by the UTL using the algorithm to enhance comparison consistency.

	Heuristic$_i$														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
w_i[%]	18.06	9.26	6.99	5.06	18.96	4.21	4.21	4.21	7.74	1.45	1.45	7.75	7.75	1.45	1.45

In parallel, experts analyzed the heuristics and valued them using Granollers instrument [55]. Table 6 details the evaluators' scores, illuminating the practical application of these heuristics (Figure 4 presents a summary for each heuristic). The presence of zeros in the scores indicates instances where evaluators deemed certain heuristics inapplicable, reflecting their professional judgment. Such agreements and differences are crucial as they affirm the heuristic evaluation process's robustness, ensuring alignment in the overall system usability assessment even when subjective judgments vary. A key point to highlight is that the value in each heuristic depends on the number of questions in that evaluation component, necessitating standardization to make a clearer and more in-depth comparison between heuristic values.

Table 6. Evaluation results: score by heuristic.

	Evaluator						
Heuristic	**1**	**2**	**3**	**4**	**5**	**6**	**7**
H_1	5.0	5.0	4.0	4.5	4.0	5.0	4.0
H_2	3.0	2.0	4.0	4.0	3.5	4.0	2.0
H_3	3.0	2.0	3.0	2.0	2.0	1.0	1.5
H_4	5.0	4.0	5.0	5.5	3.5	4.0	3.5
H_5	5.0	4.0	4.0	5.0	5.0	5.0	4.0
H_6	5.0	3.0	3.0	6.0	3.0	5.0	3.0
H_7	3.0	3.0	0.0	0.0	2.0	2.0	4.0
H_8	2.0	2.0	2.0	0.0	2.0	2.0	1.0
H_9	4.0	3.0	4.0	3.0	4.0	4.0	1.0
H_{10}	0.0	0.0	0.0	0.0	0.5	0.0	0.0
H_{11}	2.0	0.0	0.0	1.0	1.0	0.0	0.0
H_{12}	4.0	3.0	2.5	4.0	4.0	2.0	2.0
H_{13}	2.0	2.5	2.0	3.0	3.0	3.0	2.0
H_{14}	0.0	0.0	0.0	2.0	1.0	0.0	2.0
H_{15}	1.0	1.0	0.0	0.0	1.0	0.0	0.0

According to Table 6, "Visibility of system status" (H_1) and "Recognition rather than recall" (H_5) received high average scores, indicating the system excels at keeping users informed and minimizing memory load. These aspects are crucial for an intuitive user experience, ensuring users can easily understand the system's status and navigate effortlessly without relying heavily on memory. Conversely, evaluators assigned low average scores to "Help and documentation" (H_{10}) and "Latency reduction" (H_{15}), suggesting significant areas for improvement. "Help and documentation" is essential for guiding users, and its low score shows potential difficulties in accessing necessary support. Similarly, "Latency reduction" aims to minimize delays and provide timely feedback, which are critical for maintaining user engagement and satisfaction.

The standard deviations in Table 7 reflect variability in scores, highlighting differing perceptions among evaluators. For instance, "Flexibility and efficiency" (H_6) and "Help users recognize, diagnose, and recover from errors" (H_7) showed higher standard deviations, indicating varied experiences with the system's efficiency features. Meanwhile, "Help and documentation" (H_10) has a low value, indicating consensus between evaluators. However, it is important to keep in mind that Granoller's heuristics set contains different numbers of items per heuristic, and its quantification depends on the number of items, so to make a more proper comparison, it is crucial to standardize the values; in this case, we implemented the min–max scaler to compare the results properly. Those final calculations appear in Table 8 and Figure 5.

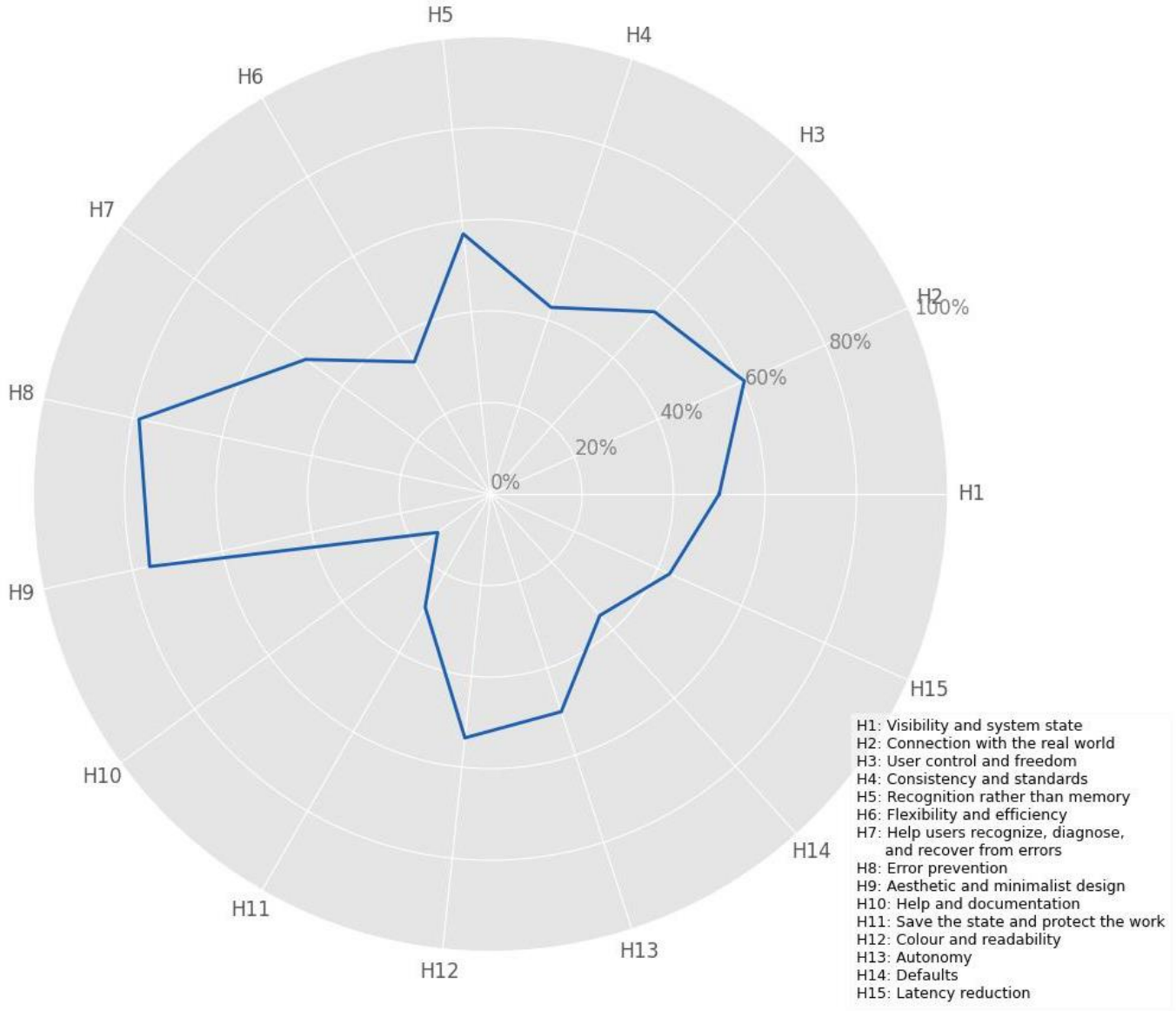

Figure 5. Radar plot with the normalized average in each heuristic.

Table 7. Summary statistics for heuristic evaluations.

	H_1	H_2	H_3	H_4	H_5	H_6	H_7	H_8	H_9	H_{10}	H_{11}	H_{12}	H_{13}	H_{14}	H_{15}
count	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00
mean	4.50	3.21	2.07	4.36	4.57	4.00	2.00	1.57	3.29	0.07	0.57	3.07	2.50	0.71	0.43
std	0.50	0.91	0.73	0.80	0.54	1.29	1.53	0.79	1.11	0.19	0.79	0.93	0.50	0.95	0.54
min	4.00	2.00	1.00	3.50	4.00	3.00	0.00	0.00	1.00	0.00	0.00	2.00	2.00	0.00	0.00
25%	4.00	2.50	1.75	3.75	4.00	3.00	1.00	1.50	3.00	0.00	0.00	2.25	2.00	0.00	0.00
50%	4.50	3.50	2.00	4.00	5.00	3.00	2.00	2.00	4.00	0.00	0.00	3.00	2.50	0.00	0.00
75%	5.00	4.00	2.50	5.00	5.00	5.00	3.00	2.00	4.00	0.00	1.00	4.00	3.00	1.50	1.00
max	5.00	4.00	3.00	5.50	5.00	6.00	4.00	2.00	4.00	0.50	2.00	4.00	3.00	2.00	1.00

Table 8. Summary statistics for heuristic evaluations after normalization.

	H_1	H_2	H_3	H_4	H_5	H_6	H_7	H_8	H_9	H_{10}	H_{11}	H_{12}	H_{13}	H_{14}	H_{15}
count	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00
mean	0.50	0.61	0.54	0.43	0.57	0.33	0.50	0.79	0.76	0.14	0.29	0.54	0.50	0.36	0.43
std	0.50	0.45	0.37	0.40	0.53	0.43	0.38	0.39	0.37	0.38	0.39	0.47	0.50	0.48	0.53
min	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
25%	0.00	0.25	0.38	0.13	0.00	0.00	0.25	0.75	0.67	0.00	0.00	0.13	0.00	0.00	0.00
50%	0.50	0.75	0.50	0.25	1.00	0.00	0.50	1.00	1.00	0.00	0.00	0.50	0.50	0.00	0.00
75%	1.00	1.00	0.75	0.75	1.00	0.67	0.75	1.00	1.00	0.00	0.50	1.00	1.00	0.75	1.00
max	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

The normalized results in Figure 5 facilitate an overview of the software performance. The software delivers commendable performance in several key usability domains, which could indicate a user-centric design philosophy. The high "Visibility and System State" and "Connection with the Real World" scores demonstrate the software's robustness in providing users with clear feedback and employing user-friendly language that aligns with real-world conventions. This alignment likely enhances user engagement and reduces the cognitive load required to interact with the software. The moderate-to-high "User Control and Freedom" and "Consistency and Standards" scores reflect a system that respects user agency, offering control through undo/redo functionalities and maintaining a consistent interface that adheres to recognized standards. These aspects foster user confidence and facilitate a smooth learning curve. However, the software's usability suffers from its moderate "Recognition Rather than Memory" score, where there is room for improvement to reduce reliance on user memory. Integrating a more intuitive design will further streamline user interactions. In "Flexibility and Efficiency", the software excels, suggesting it allows expert users to operate more efficiently, possibly through customizable shortcuts or adaptive interfaces.

This flexibility marks mature software design, catering to a broad user base with varied expertise. The lower scores in "Help Users Recognize, Diagnose, and Recover from Errors" and "Error Prevention" underscore critical areas of concern. Clarifying error messages and incorporating preventative measures could reduce user frustration and boost productivity. Addressing these issues should be a priority to enhance error management and build a more resilient system. While not alarming, the moderate score in "Aesthetic and Minimalist Design" indicates that developers could further refine the software's design to eliminate superfluous elements, thereby adhering to minimalist design principles to create a more focused user experience. A significant usability shortcoming emerges from the low score in "Help and Documentation", indicating that the help resources may be inadequate. Improving help systems is crucial for user support, especially when users face challenges or learn new features. The low scores in "Save the State" and "Protect the Work through Latency Reduction" signify systemic usability challenges that demand immediate attention. The software's evident deficiencies in preserving user states, optimizing readability through color usage, enabling user autonomy, setting effective defaults, and minimizing latency could be addressed to substantially improve the overall user experience.

After establishing the weights, we computed the usability percentages using Granollers' traditional method, subsequently labeled as "Traditional" in our analysis. We then applied our modified algorithm, referred to as "Modified". Figure 6 illustrates these outcomes, presenting paired usability scores from seven evaluators. The y-axis on the boxplot indicates the usability scores obtained under both scenarios: the Traditional approach and the Modified one. The boxplot delineates the trends and distributions of scores, while the connecting lines illustrate the shifts based on the weights applied in our enhanced calculations. Although usability percentages occasionally decrease with the new algorithm, the general trend shows an elevation in the overall usability score from 78.12% to 80.97%. Remarkably, variability also diminishes, with the standard deviation decreasing from 8.94% to 6.61%. This improvement results from our methodology's strategic deprioritization of

heuristics, which showed significant expert disagreement, thus refining their relevance through the UTL pairwise comparison.

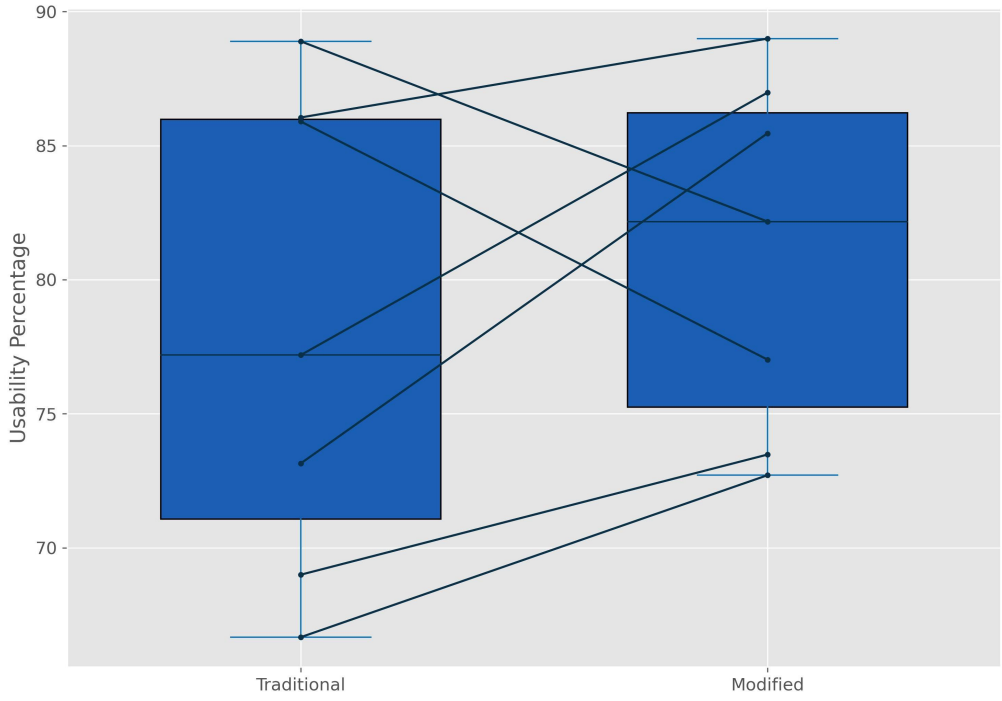

Figure 6. Paired boxplot of usability percentages.

Considering the modified approach, an overview of the relevance of each heuristic appears in Figure 7. As a result of the UTL analysis, the modified set prioritizes "Visibility and System State" and "Connection with the Real World", as evidenced by their substantial positive differences. This finding highlights that these heuristics have high standardized values and low deviations, indicating a consensus among evaluators. These heuristics are crucial for an intuitive usable user interface, suggesting the new system effectively communicates with users and aligns with their expectations. The high standardized scores for the first heuristics, while having relatively low deviations, confirm their successful implementation and the system's enhanced usability. The new evaluation system de-emphasizes heuristics like "Help and Documentation" and "Latency Reduction", as they have low standardized scores and negative differences. Without prior prioritization by the UTL, the low values with low deviations could fail to indicate areas needing development to meet user needs comprehensively. However, the significant negative deviation suggests that those areas are not a priority for improvement. "Autonomy", despite a high standardized score and low deviation among evaluators, is considered less critical in the new system's weight set. This result suggests a shift in focus or a different usability strategy adopted by the developers, implying no need for improvements. However, this component is not as relevant as other heuristics. Heuristics with low values and negative differences—such as "Error Prevention", "Aesthetic and Minimalist Design", and others—indicate these areas are less relevant in the new approach. This shift calls for careful consideration to ensure it aligns with the overall goals of the software and user expectations; that is, software engineers should prioritize those components for future enhancements to increase the overall system's usability.

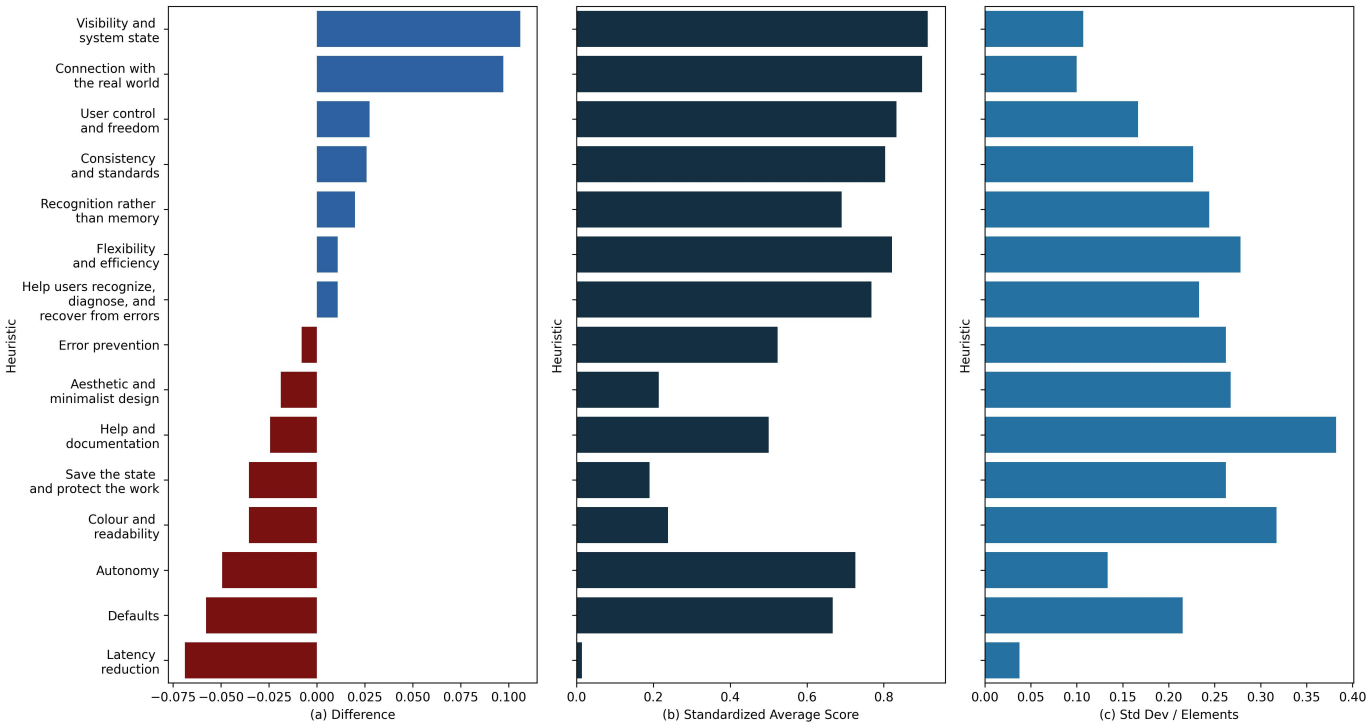

Figure 7. Barplot assorted for the major difference between Traditional and Modified weights. (**a**) Difference between both methods, (**b**) standardized average score in each heuristic, and (**c**) standardized deviation standard.

5. Discussion

The pursuit of standardization in usability testing methodologies, considering the diversity of heuristic evaluations reported in the literature [5,30,32,35,45,55,60,86–91], underscores a pivotal aim of this research. This study aims to standardize the assessment process, especially for emerging mixed-method scenarios, by focusing on a core set of heuristics [55] applicable across various usability tests. By pinpointing significant areas for improvement in our case study, we identified heuristics that scored low in evaluations despite their high importance. This discrepancy signals a critical need for targeted enhancements to improve the system's usability. By leveraging tailored algorithms that employ transitive properties for pairwise comparison, we substantially decreased the necessary comparisons, streamlining the evaluation process in those frameworks with many criteria. This method simplifies the evaluation process and significantly contributes to the accuracy of usability heuristic test results. Integrating the analytic hierarchy process and a customized algorithm using transitive features enhances the precision of prioritizing heuristics.

Comparing user perception-based methods like the SUS [53] and the technology acceptance model [31] with expert-driven heuristic evaluations [13,30,32], we find that expert evaluations offer a broader perspective by incorporating detailed analyses and guiding questions about the system. This depth of insight is instrumental in addressing nuanced usability components that non-expert assessments might overlook [19]. Prioritizing heuristics based on their relevance, as determined by the Usability Testing Leader, instead of the number of items in each heuristic as identified in the literature enhances the efficiency and focus of the assessment process [8,18,54]. This approach ensures that evaluators concentrate on the most critical aspects early on, reducing the risk of fatigue and potential bias towards the end of the evaluation. Despite heuristics being mental shortcuts, there are approaches in the literature to quantify usability to make it comparable as a strategy to record software quality and then make improvements [41,74]. Quantifying user experiences and usability is challenging due to inherent biases, regardless of the approach [8]. Prioritizing heuristics can help software developers focus on enhancing specific components to improve overall

quality [21]. Moreover, using a consistent heuristic order during evaluation can reduce biases [22]. However, considering the dynamic nature of a software development project and its relationship with the life-cycle phases [25], the UTL must consider which heuristics are relevant in each phase and how the prioritization impacts future phases.

This study advances the theoretical foundations of usability testing by standardizing the evaluation process across different methodologies and scenarios. It offers a structured approach that addresses inconsistencies in current practices and contributes to theoretical advancements in computing [63–67], especially when facing instruments with diverse numbers of items per construct [69,70]. Practically, this method facilitates the adoption of mixed-method approaches, expanding the applicability and relevance of heuristic evaluations in the evolving landscape of human–computer interaction [56]. Developing a pre-configured set of weights for various software families or technologies will further facilitate usability testing. Implementing this method requires additional expertise and training to be applied effectively. Relying on expert judgments may limit the inclusiveness of the usability testing process, potentially overlooking diverse user perspectives. The model application requires assumptions that do not always meet, such as the incidence or relationship between criteria in their components [80]. These limitations might affect the generalizability of our findings and the inclusivity of the usability testing process. However, ongoing support and training for usability professionals, as well as modifications to the strategy to estimate weights, are essential to mitigate these effects.

Future research should explore more inclusive methods that integrate diverse user perspectives into the heuristic evaluation process. Studies could also investigate the long-term impact of standardized evaluation on software quality and user satisfaction. Future studies could address the limitations identified by incorporating additional training programs for usability professionals and developing more inclusive evaluation frameworks. We chose AHP because it is a useful approach for prioritization based on natural language and does not require data assumptions like those needed for correlational models such as factor analysis, and models that depend on this fundamental approach include confirmatory factor analysis and structural equation modeling [82,83]. In addition, heuristics do not consider subcriteria, we assume independence between criteria, and we prioritize aspects for improvement rather than comparing alternatives. So we assume that the UTL has accepted the decoupling of UI heuristics. However, if relationships between criteria exist, future research must modify the comparison matrix and implement other approaches, such as the analytical network process [80]. It is important to note that choosing a multicriteria method can impact the research methodology. In this study, we selected the method based on expertise and the problem's characteristics, which align with the UTL. Evidence of this is the multicriteria selection process [75]. However, there are always biases in the method, scope, and assumptions, so carefully consider the method selected is crucial.

Another relevant modification in this work is the tailored algorithm for pairwise comparison implementing transitivity properties as an alternative to dealing with the exponential number of comparisons as the number of criteria increases: a flaw of the AHP. The tailored algorithm might focus on UTL's subjectivity to achieve a more consistent matrix, but it is a variation of the general methodology developed by Saaty [24]. In future works, the UTL can apply traditional AHP until it obtains a consistent pairwise matrix or can multiply the comparison matrix with the eigenvector of the highest value and normalize the data for consistency based on expert suggestions without implementing the tailored algorithm to avoid that possible bias. This step does not change the primary goal of quantifying and prioritizing heuristics during evaluation. In our case, we have only one hierarchy level, so transitivity does not affect the relative importance of criteria. Future works considering the relationship between subcriteria must pay attention to those relationships and be careful about the transitivity implementation. Practitioners must carefully implement AHP for weighted UI assessment, considering all assumptions regarding transitivity and emergent bias. Finally, this approach does not focus on the traditional comparison of alternatives. Still, it aims to provide a framework for better understanding

software quality improvements based on heuristic UI assessment. It generates a score to generate a baseline for comparison before and after software improvements. This score must be carefully interpreted by software developers considering the life-cycle phase of software development.

6. Conclusions

This study advances the field of human–computer interaction by introducing a standardized approach to heuristic UI assessment in usability testing. We significantly streamline the evaluation process by integrating the analytic hierarchy process and a tailored algorithm that employs transitive properties for pairwise comparison. This method not only simplifies the complexity and workload associated with the traditional prioritization process but also improves the accuracy and relevance of the usability heuristic testing results. By prioritizing heuristics based on their importance as determined by the Usability Testing Leader rather than merely depending on the number of items, scale, or heuristics, our approach ensures evaluations focus on the most critical usability aspects from the start. Furthermore, our approach addresses the challenges associated with traditional usability assessments, such as biases introduced by varying scales and the exhaustive nature of numerous comparisons. The findings from this study highlight the importance of expert-driven evaluations for gaining a thorough understanding of usability, offering a wider perspective than user-perception-based methods like the questionnaire approach. By incorporating these expert-driven methodologies, we provide a robust framework that can be adapted and extended in future research to enhance the precision and efficiency of usability testing across diverse applications.

Author Contributions: Conceptualization was collaboratively handled by L.T.-S., A.G., and H.L.-D., who laid the groundwork for the research's thematic and theoretical foundation. The methodology was developed by L.T.-S., A.G. and H.L.-D., who ensured a robust framework for the study. L.T.-S. was solely responsible for software development and providing the technical tools necessary for the research. The findings were validated by A.G., H.L.-D. and M.G.-C., who ensured the results' reliability and accuracy. Formal analysis was conducted by H.L.-D. and M.G.-C., who contributed to the rigorous examination of the data. The investigation process was led by L.T.-S., K.P.-R. and M.G.-C., who drove the research's empirical inquiry. Resources were procured by L.T.-S. and A.G., who supported the study's logistical needs. Data curation was managed by L.T.-S. and H.L.-D., who organized and maintained the study's data integrity. L.T.-S., M.G.-C. and K.P.-R. crafted the initial manuscript and undertook writing—original draft preparation. Writing—review and editing were conducted by L.T.-S. and K.P.-R., who refined the manuscript's content. L.T.-S. also took on the visualization tasks, enhancing the presentation of the study's findings. Supervision was conducted by A.G. and H.L.-D., who guided the research's strategic direction. Finally, project administration was a collective effort by all authors (L.T.-S., A.G., H.L.-D., M.G.-C. and K.P.-R.), who coordinated the study's operational aspects. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Colombian Bureau of Science (Minciencias, *Ministerio de Ciencia, Tecnología e Innovación*), grant number BPIN 2019000100019—CDP 820.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original data presented in the study are openly available and can be accessed by registering at https://ruxailab-prod.web.app (accessed on 1 January 2024). The original contributions presented in the study are included in this article. Further inquiries can be directed to the corresponding author.

Acknowledgments: We gratefully acknowledge the support provided by "Programa Institucional de Bolsas de Iniciação em Desenvolvimento Tecnológico e Inovação Facens" (PIBITIF · grant 800916/2022-0). The authors also thank the Universidad Autónoma de Bucaramanga and its mobility program, the Universitat de Lleida and the Universidad Industrial de Santander for their invaluable support and contributions to this research.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AHP	Analytic Hierarchy Process
CI	Consistency Index
CR	Consistency Ratio
DOC	Document (Microsoft Word Format)
HCI	Human–Computer Interaction
ISO	International Organization for Standardization
PDF	Portable Document Format
PSSUQS	Post-Study System Usability Questionnaire Software
QUIS	User Interaction Satisfaction
SUS	System Usability Scale
SUMI	Usability Measurement Inventory
UI	User Interface
UP	Usability Percentage
UTL	Usability Testing Leader
XLS	Excel Spreadsheet (Microsoft Excel Format)

Appendix A. Tailored Algorithm for AHP

```
def calculate_eigen(matrix):
    eigenvalues, eigenvectors = np.linalg.eig(matrix)
    max_eigenvalue = np.max(eigenvalues)
    max_eigenvector = eigenvectors[:, np.argmax(eigenvalues)]

    # Normalize the eigenvector to get the weights
    normalized_weights = max_eigenvector / np.sum(max_eigenvector)

    # Calculate the consistency index (CI)
    n = matrix.shape[0]
    CI = (max_eigenvalue - n) / (n - 1)

    # Random consistency index (RI), values depend on matrix size
    RI_dict = {1: 0, 2: 0, 3: 0.58, 4: 0.90, 5: 1.12, 6: 1.24, 7: 1.32,
               8: 1.41, 9: 1.45, 10: 1.49, 11: 1.52, 12: 1.54, 13: 1.56,
               14: 1.58, 15: 1.59, 16: 1.60, 17: 1.61,18: 1.62, 19: 1.63,
               20: 1.64, 21: 1.65, 22: 1.66, 23: 1.67, 24: 1.68, 25: 1.69,
               26: 1.70, 27: 1.71, 28: 1.72, 29: 1.73, 30: 1.74}
    RI = RI_dict.get(n, 1.49)  # 1.49 is an average fallback value

    # Calculate the consistency ratio (CR)
    CR = CI / RI

    consistency_interpretation =
    (‘‘Consistent because CR is lower than 0.1’’) if CR <= 0.1 ...
    else ‘‘Inconsistent because CR is greater than CR’’

    return max_eigenvalue, normalized_weights.real, ...
    CR, consistency_interpretation

def initialize_ahp_matrix(df, column_name):
    categories = df[column_name].tolist()
    n = len(categories)

    # Initialize a zero matrix of dimensions n x n
    ahp_matrix = np.zeros((n, n))
```

```
    # Create a labeled DataFrame to hold the AHP matrix
    ahp_df = pd.DataFrame(ahp_matrix, index=categories, columns=categories)

    return ahp_df

def generate_saaty_scale_with_explanations():
    return {
        'Equal Importance': 1,
        'Moderate Importance': 3,
        'Strong Importance': 5,
        'Very Strong Importance': 7,
        'Extreme Importance': 9,
        'Moderately Less Important': 1/3,
        'Strongly Less Important': 1/5,
        'Very Strongly Less Important': 1/7,
        'Extremely Less Important': 1/9
    }

def fill_ahp_matrix(ahp_df, row_name, col_names, comparison):
    saaty_scale = generate_saaty_scale_with_explanations()
    if comparison in saaty_scale:
        value = saaty_scale[comparison]
        for col_name in col_names:
            ahp_df.loc[row_name, col_name] = value
            ahp_df.loc[col_name, row_name] = 1 / value
    else:
        print(''Invalid comparison description. ...
        Please select one from Saaty's scale.'')
    return ahp_df

def populate_ahp_matrix(ahp_df):
    saaty_scale_dict = {i+1: option for I, ...
    option in enumerate(generate_saaty_scale_with_explanations().keys())}

    for row in ahp_df.index:
        temp_saaty_scale_dict = saaty_scale_dict.copy()

        criteria_dict = {i+1: col for i, col in enumerate(ahp_df.columns) ...
        if col != row and ahp_df.loc[row, col] == 0}
        temp_criteria_dict = criteria_dict.copy()

        while temp_criteria_dict:
            print(f''\nSelect an option for comparisons involving {row} ...
            against remaining criteria:'')

            # Show available Saaty's scale options
            for num, option in temp_saaty_scale_dict.items():
                print(f''Saaty {num}. {option}'')

            # Show remaining criteria mapped to numbers
            for num, criteria in temp_criteria_dict.items():
                print(f''Criteria {num}. {criteria}'')

            saaty_selection = int(input(''Enter the number of your Saaty ...
            scale selection: ''))
            selected_comparison = temp_saaty_scale_dict[saaty_selection]

            print(f''Indicate all criteria from the list above that have ...
```

```
                    '{selected_comparison}' when compared to {row}. ...
                    Separate multiple criteria by comma.'')
                    relevant_cols_numbers = input().split(',')
                    relevant_cols = [temp_criteria_dict[int(num.strip())] ...
                    for num in relevant_cols_numbers]

                    ahp_df = fill_ahp_matrix(ahp_df, row, relevant_cols, ...
                    selected_comparison)

                    # Pre-fill for transitive relations, i.e., if A = B and ...
                    A = C, then B = C
                    if selected_comparison == 'Equal Importance':
                        for i in range(len(relevant_cols)):
                            for j in range(i+1, len(relevant_cols)):
                                ahp_df.loc[relevant_cols[i], relevant_cols[j]] = 1
                                ahp_df.loc[relevant_cols[j], relevant_cols[i]] = 1

                    # Update temp_criteria_dict to remove selected items
                    temp_criteria_dict = {num: col for num, col in ...
                    temp_criteria_dict.items() if col not in relevant_cols}

                    # Update temp_saaty_scale_dict to exclude the selected comparison
                    del temp_saaty_scale_dict[saaty_selection]

            # Set diagonal elements to 1
            np.fill_diagonal(ahp_df.values, 1)

            return ahp_df
```

References

1. Vlachogianni, P.; Tselios, N. Perceived usability evaluation of educational technology using the System Usability Scale (SUS): A systematic review. *J. Res. Technol. Educ.* **2022**, *54*, 392–409. [CrossRef]
2. Giacomin, J. What is human centred design? *Des. J.* **2014**, *17*, 606–623. [CrossRef]
3. Holeman, I.; Kane, D. Human-centered design for global health equity. *Inf. Technol. Dev.* **2020**, *26*, 477–505. [CrossRef] [PubMed]
4. Peruzzini, M.; Carassai, S.; Pellicciari, M. The Benefits of Human-centred Design in Industrial Practices: Re-design of Workstations in Pipe Industry. *Procedia Manuf.* **2017**, *11*, 1247–1254. [CrossRef]
5. Ng, J.; Arness, D.; Gronowski, A.; Qu, Z.; Lau, C.W.; Catchpoole, D.; Nguyen, Q.V. Exocentric and Egocentric Views for Biomedical Data Analytics in Virtual Environments—A Usability Study. *J. Imaging* **2024**, *10*, 3. [CrossRef] [PubMed]
6. Harrison, R.; Flood, D.; Duce, D. Usability of mobile applications: Literature review and rationale for a new usability model. *J. Interact. Sci.* **2013**, *1*, 1–16. [CrossRef]
7. Sari, I.; Tj, H.W.; Wahyoedi, S.; Widjaja, B.T. The Effect of Usability, Information Quality, and Service Interaction on E-Loyalty Mediated by E-Satisfaction on Hallobumil Application Users. *KnE Soc. Sci.* **2023** , *8*, 211–229. [CrossRef]
8. Tullis, T.; Albert, W. *Measuring the User Experience, Second Edition: Collecting, Analyzing, and Presenting Usability Metrics*, 2nd ed.; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2013.
9. Brooke, J. *SUS: A 'Quick and Dirty' Usability Scale*; CRC Press: Boca Raton, FL, USA, 1996. [CrossRef]
10. Chin, J.P.; Diehl, V.A.; Norman, K.L. Development of an instrument measuring user satisfaction of the human-computer interface. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Washington, DC, USA, 15–19 May 1988; Part F130202. [CrossRef]
11. Kirakowski, J.; Cierlik, B. Measuring the Usability of Web Sites. *Proc. Hum. Factors Ergon. Soc. Annu. Meet.* **1998**, *42*, 424–428. [CrossRef]
12. Lewis, J.R. Psychometric evaluation of the post-study system usability questionnaire: The PSSUQ. In *Proceedings of the Human factors Society Annual Meeting*; Sage Publications: Los Angeles, CA, USA, 1992; Volume 2. [CrossRef]
13. Nielsen, J.; Molich, R. *Heuristic Evaluation of User Interfaces*; ACM Press: New York, NY, USA, 1990; pp. 249–256. [CrossRef]
14. Bangor, A.; Kortum, P.T.; Miller, J.T. An empirical evaluation of the system usability scale. *Int. J. Hum.-Comput. Interact.* **2008**, *24*, 574–594. [CrossRef]
15. Păsărelu, C.R.; Kertesz, R.; Dobrean, A. The Development and Usability of a Mobile App for Parents of Children with ADHD. *Children* **2023**, *10*, 164. [CrossRef]
16. Weichbroth, P. Usability Testing of Mobile Applications: A Methodological Framework. *Appl. Sci.* **2024**, *14*, 1792. [CrossRef]

17. Rosenzweig, E. *Usability Inspection Methods*; Elsevier: Amsterdam, The Netherlands, 2015; pp. 115–130. [CrossRef]
18. Maqbool, B.; Herold, S. Potential effectiveness and efficiency issues in usability evaluation within digital health: A systematic literature review. *J. Syst. Softw.* **2024**, *208*, 111881. [CrossRef]
19. Generosi, A.; Villafan, J.Y.; Giraldi, L.; Ceccacci, S.; Mengoni, M. A Test Management System to Support Remote Usability Assessment of Web Applications. *Information* **2022**, *13*, 505. [CrossRef]
20. Veral, R.; Macías, J.A. Supporting user-perceived usability benchmarking through a developed quantitative metric. *Int. J. Hum. Comput. Stud.* **2019**, *122*, 184–195. [CrossRef]
21. Bugayenko, Y.; Bakare, A.; Cheverda, A.; Farina, M.; Kruglov, A.; Plaksin, Y.; Pedrycz, W.; Succi, G. Prioritizing tasks in software development: A systematic literature review. *PLoS ONE* **2023**, *18*, e0283838. [CrossRef] [PubMed]
22. Israel, G.D.; Taylor, C. Can response order bias evaluations? *Eval. Program Plan.* **1990**, *13*, 365–371. [CrossRef]
23. Paz, F.; Pow-Sang, J.A. A systematic mapping review of usability evaluation methods for software development process. *Int. J. Softw. Eng. Its Appl.* **2016**, *10*, 165–178. [CrossRef]
24. Saaty, R.W. The analytic hierarchy process-what it is and how it is used. *Math. Model.* **1987**, *9*, 161–176. [CrossRef]
25. Dhillon, B.S. *Usability Engineering Life-Cycle Stages and Important Associated Areas*; CRC Press: Boca Raton, FL, USA, 2019. . [CrossRef]
26. Alonso-Ríos, D.; Vázquez-García, A.; Mosqueira-Rey, E.; Moret-Bonillo, V. Usability: A Critical Analysis and a Taxonomy. *Int. J. Hum.-Comput. Interact.* **2009**, *26*, 53–74. [CrossRef]
27. Villani, V.; Lotti, G.; Battilani, N.; Fantuzzi, C. Survey on usability assessment for industrial user interfaces. *IFAC-PapersOnLine* **2019**, *52*, 25–30. [CrossRef]
28. Shneiderman, B. Designing the user interface strategies for effective human-computer interaction. *ACM SIGBIO Newsl.* **1987**, *9*. [CrossRef]
29. Norman, D. *The Design of Everyday Things*; Vahlen: Munich, Germany 2016. [CrossRef]
30. Tognazzini, B. First Principles, HCI Design, Human Computer Interaction (HCI), Principles of HCI Design, Usability Testing. 2014. Available online: http://www.asktog.com/basics/firstPrinciples.html (accessed on 1 January 2024).
31. Shyr, W.J.; Wei, B.L.; Liang, Y.C. Evaluating Students' Acceptance Intention of Augmented Reality in Automation Systems Using the Technology Acceptance Model. *Sustainability* **2024**, *16*, 2015. [CrossRef]
32. Nielsen, J.; Molich, R. Heuristic evaluation of user interfaces. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, 1–5 April 1990; pp. 249–256.
33. Scholtz, J. Beyond Usability: Evaluation Aspects of Visual Analytic Environments. In Proceedings of the 2006 IEEE Symposium On Visual Analytics Science And Technology, Baltimore, MD, USA , 31 October–2 November 2006; pp. 145–150. [CrossRef]
34. Lewis, C.; Poison, P.; Wharton, C.; Rieman, J. Testing a walkthrough methodology for theory-based design of walk-up-and-use interfaces. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, 1–5 April 1990. [CrossRef]
35. Thomas, C.; Bevan, N. Usability Context Analysis: A Practical Guide, Version 4.04. Edited by Cathy Thomas and Nigel Bevan. *Serco Usability Services*, Loughborough University, 1996. Available online: https://hdl.handle.net/2134/2652 (accessed on 1 January 2024).
36. Jaspers, M.W. A comparison of usability methods for testing interactive health technologies: Methodological aspects and empirical evidence. *Int. J. Med. Inform.* **2009**, *78*, 340–353. [CrossRef]
37. Rubin, J.; Chisnell, D.; Spool, J. *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests*, 2nd ed.; Wiley: Hoboken, NJ, USA, 2008; p. 384.
38. Law, E.L.C.; Hvannberg, E.T. Analysis of combinatorial user effect in international usability tests. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Vienna, Austria, 24–29 April 2004; pp. 9–16. [CrossRef]
39. Molich, R.; Nielsen, J. Improving a Human-Computer Dialogue. *Commun. ACM* **1990**, *33*, 338–348. [CrossRef]
40. Hartson, R.; Pyla, P.S. *Rapid Evaluation Methods*; Elsevier: Amsterdam, The Netherlands, 2012; pp. 467–501. [CrossRef]
41. Delice, E.K.; Güngör, Z. The usability analysis with heuristic evaluation and analytic hierarchy process. *Int. J. Ind. Ergon.* **2009**, *39*, 934–939. [CrossRef]
42. Virzi, R.A.; Sorce, J.F.; Herbert, L.B. Comparison of three usability evaluation methods: Heuristic, think-aloud, and performance testing. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*; Sage Publications: Los Angeles, CA, USA, 1993; Volume 1. [CrossRef]
43. Lazar, J.; Feng, J.H.; Hochheiser, H. *Usability Testing*; Elsevier: Amsterdam, The Netherlands, 2017; pp. 263–298. [CrossRef]
44. Quiñones, D.; Rusu, C. How to develop usability heuristics: A systematic literature review. *Comput. Stand. Interfaces* **2017**, *53*, 89–122. [CrossRef]
45. Jaferian, P.; Hawkey, K.; Sotirakopoulos, A.; Velez-Rojas, M.; Beznosov, K. Heuristics for evaluating IT security management tools. In Proceedings of the Seventh Symposium on Usable Privacy and Security, Menlo Park, CA, USA, 9–11 July 2014; Volume 29. [CrossRef]
46. Lechner, B.; Fruhling, A.L.; Petter, S.; Siy, H.P. The Chicken and the Pig: User Involvement in Developing Usability Heuristics. In *19th Americas Conference on Information Systems, AMCIS 2013 - Hyperconnected World: Anything, Anywhere, Anytime*; Association for Information Systems: Chicago, IL, USA, 2013; Volume 5, pp. 3263–3270.

47. Sim, G.; Read, J.C.; Cockton, G. Evidence based design of heuristics for computer assisted assessment. In *Human-Computer Interaction–INTERACT 2009: 12th IFIP TC 13 International Conference, Uppsala, Sweden, August 24–28. 2009*; Springer: Berlin/Heidelberg, Germany, 2009; Volume 5726 LNCS._25. [CrossRef]
48. Ling, C.; Salvendy, G. Extension of heuristic evaluation method: A review and reappraisal. *Ergon. IJE HF* **2005**, *27*, 179–197.
49. Paddison, C.; Englefield, P. Applying heuristics to accessibility inspections. *Interact. Comput.* **2004**, *16*, 507–521. [CrossRef]
50. Inostroza, R.; Rusu, C.; Roncagliolo, S.; Rusu, V.; Collazos, C.A. Developing SMASH: A set of SMArtphone's uSability Heuristics. *Comput. Stand. Interfaces* **2016**, *43*, 40–52. [CrossRef]
51. Hermawati, S.; Lawson, G. Establishing usability heuristics for heuristics evaluation in a specific domain: Is there a consensus? *Appl. Ergon.* **2016**, *56*, 34–51. [CrossRef]
52. Bailey, R.W.; Wolfson, C.A.; Nall, J.; Koyani, S. Performance-Based Usability Testing: Metrics That Have the Greatest Impact for Improving a System's Usability. In *Human Centered Design*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 3–12._1. [CrossRef]
53. Mitta, D.A. A Methodology for Quantifying Expert System Usability. *Hum. Factors J. Hum. Factors Ergon. Soc.* **1991**, *33*, 233–245. [CrossRef]
54. Benaida, M. Developing and extending usability heuristics evaluation for user interface design via AHP. *Soft Comput.* **2023**, *27*, 9693–9707. [CrossRef]
55. Granollers, T. *Usability Evaluation with Heuristics, Beyond Nielsen's List*; ThinkMind Digital Library: Lisbon, Portugal, 2018; pp. 60–65.
56. Sharp, H.; Preece, J.; Rogers, Y. *Interaction Design: Beyond Human-Computer Interaction*, 5th ed.; John Wilei & Sons Inc.: Hoboken, NJ, USA, 2019; p. 656.
57. Bonastre, L.; Granollers, T. A set of heuristics for user experience evaluation in E-commerce websites. In Proceedings of the 7th International Conference on Advances in Computer-Human Interactions, Barcelona, Spain, 23–27 March 2014.
58. Paz, F.; Paz, F.A.; Sánchez, M.; Moquillaza, A.; Collantes, L. Quantifying the usability through a variant of the traditional heuristic evaluation process. In *Design, User Experience, and Usability: Theory and Practice: 7th International Conference, DUXU 2018, Held as Part of HCI International 2018, Las Vegas, NV, USA, July 15–20. 2018*; Springer International Publishing: Cham, Switzerland, 2018; Volume 10918 LNCS._36. [CrossRef]
59. Kemp, E.A.; Thompson, A.J.; Johnson, R.S. Interface evaluation for invisibility and ubiquity—An example from E-learning. In Proceedings of the 9th ACM SIGCHI New Zealand Chapter's International Conference on Human-Computer Interaction: Design Centered HCI, Wellington, New Zealand, 2 July 2008. [CrossRef]
60. Pierotti, D. *Heuristic Evaluation: A System Checklist*; Xerox Corporation: Norwalk, CT, USA, 1995; pp. 1–22. Available online: https://users.polytech.unice.fr/~pinna/MODULEIHM/ANNEE2010/CEIHM/XEROX (accessed on 1 January 2024).
61. Khowaja, K.; Al-Thani, D. New Checklist for the Heuristic Evaluation of mHealth Apps (HE4EH): Development and Usability Study. *JMIR MHealth UHealth* **2020**, *8*, e20353. [CrossRef]
62. Holey, R.H. *Handbook of Structural Equation Modeling*, 1st ed.; The Guilford Press: New York, NY, USA, 2012; pp. 3–16.
63. Brodsky, S.L.; Lichtenstein, B. The Gold Standard and the Pyrite Principle: Toward a Supplemental Frame of Reference. *Front. Psychol.* **2020**, *11*, 562. [CrossRef]
64. Williamson, K. *Questionnaires, Individual Interviews and Focus Group Interviews*; Elsevier: Amsterdam, The Netherlands, 2018; pp. 379–403. [CrossRef]
65. Thiem, A.; Duşa, A. *Qualitative Comparative Analysis with R*; Springer: New York, NY, USA, 2013; Volume 5, p. 99. [CrossRef]
66. Contreras-Pacheco, O.E.; Talero-Sarmiento, L.H.; Camacho-Pinto, J.C. Effects of Corporate Social Responsibility on Employee Organizational Identification: Authenticity or Fallacy. *Contaduría Y Adm.* **2019**, *64*, 1–22. [CrossRef]
67. Leventhal, B.C.; Ames, A.J.; Thompson, K.N. Simulation Studies for Psychometrics. In *International Encyclopedia of Education*, Fourth ed.; Elsevier: Amsterdam, The Netherlands, 2022. [CrossRef]
68. Tanner, K. Survey Designs. In *Research Methods: Information, Systems, and Contexts*, 2nd ed.; Elsevier: Amsterdam, The Netherlands, 2018. [CrossRef]
69. Bubaš, G.; Čižmešija, A.; Kovačić, A. Development of an Assessment Scale for Measurement of Usability and User Experience Characteristics of Bing Chat Conversational AI. *Future Internet* **2024**, *16*, 4. [CrossRef]
70. van der Linden, W.J. Item Response Theory. In *Encyclopedia of Social Measurement*; Elsevier: Amsterdam, The Netherlands, 2004. [CrossRef]
71. Gupta, K.; Roy, S.; Poonia, R.C.; Nayak, S.R.; Kumar, R.; Alzahrani, K.J.; Alnfiai, M.M.; Al-Wesabi, F.N. Evaluating the Usability of mHealth Applications on Type 2 Diabetes Mellitus Using Various MCDM Methods. *Healthcare* **2022**, *10*, 4. [CrossRef]
72. Muhammad, A.; Siddique, A.; Naveed, Q.N.; Khaliq, U.; Aseere, A.M.; Hasan, M.A.; Qureshi, M.R.N.; Shahzad, B. Evaluating Usability of Academic Websites through a Fuzzy Analytical Hierarchical Process. *Sustainability* **2021**, *13*, 2040. [CrossRef]
73. Iryanti, E.; Santosa, P.I.; Kusumawardani, S.S.; Hidayah, I. Inverse Trigonometric Fuzzy Preference Programming to Generate Weights with Optimal Solutions Implemented on Evaluation Criteria in E-Learning. *Computers* **2024**, *13*, 68. [CrossRef]
74. Gulzar, K.; Tariq, O.; Mustafa, S.; Mohsin, S.M.; Kazmi, S.N.; Akber, S.M.A.; Abazeed, M.; Ali, M. A Fuzzy Analytic Hierarchy Process for Usability Requirements of Online Education Systems. *IEEE Access* **2023**, *11*, 146076–146089. [CrossRef]
75. Munier, N.; Hontoria, E. *Uses and Limitations of the AHP Method*; Springer International Publishing: Cham, Switzerland, 2021. [CrossRef]

76. Sakulin, S.; Alfimtsev, A. Multicriteria Decision Making in Tourism Industry Based on Visualization of Aggregation Operators. *Appl. Syst. Innov.* **2023**, *6*, 74. [CrossRef]
77. Salanti, G.; Nikolakopoulou, A.; Efthimiou, O.; Mavridis, D.; Egger, M.; White, I.R. Introducing the Treatment Hierarchy Question in Network Meta-Analysis. *Am. J. Epidemiol.* **2022**, *191*, 930–938. [CrossRef]
78. Marttunen, M.; Belton, V.; Lienert, J. Are objectives hierarchy related biases observed in practice? A meta-analysis of environmental and energy applications of Multi-Criteria Decision Analysis. *Eur. J. Oper. Res.* **2018**, *265*, 178–194. [CrossRef]
79. Abrahantes, J.C.; Molenberghs, G.; Burzykowski, T.; Shkedy, Z.; Abad, A.A.; Renard, D. Choice of units of analysis and modeling strategies in multilevel hierarchical models. *Comput. Stat. Data Anal.* **2004**, *47*, 537–563. [CrossRef]
80. Saaty, T.L. Decision making—The Analytic Hierarchy and Network Processes (AHP/ANP). *J. Syst. Sci. Syst. Eng.* **2004**, *13*, 1–35. [CrossRef]
81. Ishizaka, A.; Labib, A. Review of the main developments in the analytic hierarchy process. *Expert Syst. Appl.* **2011**, *38*, 14336–14345. [CrossRef]
82. Sluser, B.; Plavan, O.; Teodosiu, C. *Environmental Impact and Risk Assessment*; Elsevier: Amsterdam, The Netherlands, 2022; pp. 189–217. [CrossRef]
83. Vaidya, O.S.; Kumar, S. Analytic hierarchy process: An overview of applications. *Eur. J. Oper. Res.* **2006**, *169*, 1–29. [CrossRef]
84. Fishburn, P.C. Nontransitive preferences in decision theory. *J. Risk Uncertain.* **1991**, *4*, 113–134. [CrossRef]
85. Wu, Z.; Tu, J. Managing transitivity and consistency of preferences in AHP group decision making based on minimum modifications. *Inf. Fusion* **2021**, *67*, 125–135. [CrossRef]
86. Bevan, N. Measuring usability as quality of use. *Softw. Qual. J.* **1995**, *4*, 115–130. [CrossRef]
87. Omar, K.; Rapp, B.; Gómez, J.M. Heuristic evaluation checklist for mobile ERP user interfaces. In Proceedings of the 2016 7th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 5–7 April 2016, pp. 180–185.
88. Aballay, L.; Lund, M.I.; Gonzalez Capdevila, M.; Granollers, T. Heurísticas de Usabilidad utilizando una Plataforma Abierta y Colaborativa Práctica Áulica Aplicada a Sitios e-commerce. In *V Congreso Internacional de Ciencias de la Computación y Sistemas de Información 2021–CICCSI 2021*; CICCSI: Mendoza, Argentina, 2021. Available online: https://www.researchgate.net/publication/358430089_Heuristicas_de_Usabilidad_utilizando_una_Plataforma_Abierta_y_Colaborativa_Practica_Aulica_Aplicada_a_Sitios_e-commerce (accessed on 1 January 2024).
89. Yáñez Gómez, R.; Cascado Caballero, D.; Sevillano, J.L. Heuristic Evaluation on Mobile Interfaces: A New Checklist. *Sci. World J.* **2014**, *2014*, 434326. [CrossRef] [PubMed]
90. Komarkova, J.; Visek, O.; Novak, M. Heuristic evaluation of usability of GeoWeb sites. In *Web and Wireless Geographical Information Systems: 7th International Symposium, W2GIS 2007, Cardiff, UK, November 28–29. 2007*; Proceedings 7; Springer: Berlin/Heidelberg, Germany, 2007; pp. 264–278.
91. Almenara, A.P.; Humanes, J.; Granollers, T. MPIu+aX, User-Centered Design methodology that empathizes with the user and generates a better accessible experience. (From theory to practice). In Proceedings of the XXIII International Conference on Human Computer Interaction, Copenhagen, Denmark, 23–28 July 2023; pp. 1–3. [CrossRef]