Information and Software Technology 168 (2024) 107408

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

UXH-GEDAPP: A set of user experience heuristics for evaluating generative design applications

Daniela Quiñones [a,*], Claudia Ojeda [a], Rodrigo F. Herrera [b], Luis Felipe Rojas [a]

[a] Escuela de Ingeniería Informática, Facultad de Ingeniería, Pontificia Universidad Católica de Valparaíso, Chile
[b] School of Civil Engineering, Faculty of Engineering, Pontificia Universidad Católica de Valparaíso, Chile

ARTICLE INFO

Keywords:
Generative design
Use experience
Heuristic evaluation
User experience heuristics

ABSTRACT

Context: Traditional building and infrastructure design methodologies are inflexible and inefficient, leading to high costs and environmental damage. Generative design, with an algorithm that provides multiple options, could be a potential solution. The challenge is creating an intuitive, user-friendly application that optimizes engineers' time, reducing manual iterations and lead to a good user experience (UX). A method for evaluating the UX is heuristic evaluation, in which heuristics are used to inspect a software product.
Objective: Since generative design applications have specific features, generic heuristics may not detect all problems related to UX. This article presents a novel set of 9 heuristics to evaluate UX in generative design applications: UXH-GEDAPP. This set is focused on evaluating both UX attributes and specific features of generative design applications.
Method: A formal methodology was used to develop the heuristics, through 7 stages: exploratory, descriptive, correlational, selection, specification, validation, and refinement. We performed 3 iterations and validated UXH-GEDAPP in 2 iterations through: heuristic evaluation, expert judgment, and user test. Since the methodology can be applied iteratively, we validated and refined the set to improve the proposal.
Results: The results obtained in the validation stage indicate that UXH-GEDAPP is useful and more effective than generic heuristics when evaluating generative design applications. UXH-GEDAPP allows to detect specific usability/UX problems as well as more severe problems related to generative design applications. Furthermore, evaluators made fewer errors associating the detected problems with the proposed heuristics, compared to generic sets.
Conclusion: UXH-GEDAPP is a new set of heuristics that encourages the creation and use of generative design applications with good UX. It can detect usability/UX problems and help correct them, as well as guide the development of new generative design applications for a pleasant and intuitive user experience.

1. Introduction

In recent years, several methodologies and technologies have helped to optimize the processes and products generated in the architecture, engineering, and construction industry (AEC) [1]. One of the most contemporary advances is generative design, through which designers and engineers establish parameters or rules to optimize and then develop an algorithm that improves the design and delivers various alternatives [2]. This is because traditional design methodologies restrict flexibility, causing inefficient designs that use more material than necessary. They have become more expensive and have a greater environmental impact [3]. For example, leveraging generative design algorithms inspired by slime mold, aircraft designers efficiently cycled through numerous design variants, considering parameters like materials, loads, and constraints, to ultimately arrive at an optimal design, i.e., aircraft lighter, more efficient, and a smaller carbon footprint [4]. Within the AEC, generative design methodology has been mostly applied in architecture, with little use in engineering and construction [5]. For instance, in the field of structural, a subset of civil engineering, generative design has been applied to enhance elements based on mathematical, size, shape, and topology constraints [6]; for example, optimization of a cantilever beam using 3D-printing [3], material optimization for concrete beams or slabs [7]. Currently, engineering offices are transitioning from computer-aided design (CAD) to a methodology

* Corresponding author.
E-mail address: daniela.quinones@pucv.cl (D. Quiñones).

https://doi.org/10.1016/j.infsof.2024.107408
Received 2 September 2023; Received in revised form 12 January 2024; Accepted 15 January 2024
Available online 17 January 2024

based on building information modeling (BIM) [8]. The above, together with new methodologies such as generative design, presents an implementation challenge, where it is important to design intuitive and easy-to-use systems that generate a good experience for users, specifically engineers and designers. Generative designs made in BIM environments have required the creation of new software or adaptations of traditional software [9,10]. These new software and new functionalities require to be evaluated from the user's perspective, since it is a different design methodology that requires a different process flow than the traditional ones [11].

User experience (UX) is a relevant element to consider when designing software products. If a software product or application is difficult to use, or the user does not know what it is, they will look for a simpler alternative that meets their expectations. However, if a product is poorly designed, without considering the users (their characteristics and needs), their work will be affected, causing them to perform tasks less efficiently or generating frustration. Even products that are difficult to use can result in serious errors that compromise people's safety [12]. Several methods allow for the evaluation of UX [13]. One of these methods is heuristic evaluation, in which usability/UX experts inspect an interface using heuristics to identify potential usability/UX problems [14]. Each software product has features that differentiate it from the other systems. Therefore, even though traditional (or generic) heuristics have provided solid results for usability evaluation, the evolving landscape requires the development of domain-specific heuristics to effectively evaluate the user experience of new software and interfaces tailored to the context. Thus, several authors have designed new sets of heuristics to develop more effective evaluation instruments [15,16].

In this article, we present the UXH-GEDAPP, a novel set of heuristics that evaluates the user experience of generative design applications. This set was proposed to evaluate several aspects of the UX, including usability. The heuristics were created in three iterations using the methodology proposed by Quiñones et al. [17,18].

UXH-GEDAPP was validated and refined through several experiments with experts (expert judgment and heuristic evaluation) and users (thinking aloud test). The results obtained in the validations prove that UXH-GEDAPP detects more usability/UX problems than generic heuristics. Furthermore, experts perceived heuristics as useful and easy to understand when using them in a heuristic evaluation.

2. Background

2.1. Generative design applications

A new type of design that is currently in use, known as generative design, integrates computational technology with parametric design, while still considering the designer's data and requirements. The practice of applying building information modeling and treating the various connecting components as variables that can be changed at various stages of the design process is referred to as "parametric design". To automatically generate alternative solutions and optimized design possibilities that satisfy the initial objectives and regulations [7,19], generative design requires specifying geographic constraints, materials to be used, and geometric limitations, among other aspects [7]. These parameters are easily changeable, allowing any rule to be changed and immediately observing a change in the given design solutions without starting over. Using the traditional design form would not have produced attractive designs as our generative design technique does [20]. This form of design is used in a variety of fields, including engineering, fashion, music, literature, generative logos, and architecture, in which the Architecture Engineering and Construction (AEC) industry has grown the most [19]. By dynamically maximizing the many available design alternatives, generative design attempts to provide several models that satisfy previously determined design objectives while also satisfying the constraints and conditions specified by the user [20]. To construct a method or script that produces a solution, the designer's notion must be represented by rules or algorithms [19].

To expand its applicability in civil engineering, Díaz et al. [11] synthesized several generative design applications in structural engineering. Consequently, seven application cases were obtained, with the optimization of the material for structural parts being the primary purpose indicated for generative design. In addition, most respondents do not comprehend or know very little about this procedure, even though after learning about it, they think it can be used in their professional work [11]. In this view, it is also crucial to consider the various experiences that architecture has had with the procedure, which first and foremost aims to reduce the energy costs associated with building construction and operation while, delivering various design options quickly while considering the parameters and input variables of the code. At the outset of every project, the necessity of collaboration across the AEC sector became clear. Here, generative design aids in the creation of pre-design options quickly, while taking the most crucial factors into account, such as the structure's solar exposure and any spatial restrictions imposed by laws such as the ground level, land occupation coefficient, and constructability coefficient. Following the traditional trial and error design technique, civil engineers might then optimize the structural components of this design to consider configurations that they might not have anticipated. Finally, the generative design can be used to calculate the ideal crane placement for building the above model [11,21].

We identified the following features that describe generative design applications [22,23]. These features were used to develop the new set of heuristics, UXH-GEDAPP.

1. Adaptative: Generative design can be adapted to any parameter provided by the user, such as size, cost, and volume.
2. Optimizable: Generative design allows significant savings in design time, particulary in the early stages of the project, as well as material and cost savings.
3. Autonomous: Generative design during the development of its algorithm does not require any human intervention (autonomous in the operation).
4. Efficient: Generative design optimally fulfills the design process, thus generating hundreds of alternatives for the user.
5. Innovative: Generative design provides a wide variety of new and unthinkable designs by designers.
6. Customizable: Generative design is completely customizable depending on the user's preferences.
7. Flexible: Generative design can be integrated into any design context if required.
8. Multiple results: Generative design provides several output results based on the user's choice.
9. Selective: Generative design selects different results based on the parameters and restrictions given by the user, organizing them, and the last design given by the software is the optimal.

2.2. User experience

User Experience (UX) refers to the perceptions and responses of people because of the use (or anticipated use) of a product, system, or service [24]. The UX includes the emotions, preferences, physical and psychological perceptions that occur before, during or after the interaction [24]. The UX has different attributes that describes it. For developing UXH-GEDAPP, we considered the following UX attributes proposed by Morville [25]: useful, usable, findable, credible, valuable, and desirable. As stated by Morville and other authors [26], usability is part of the UX attributes. Usability is the degree to which specific users can use a product to achieve specific goals with efficacy, efficiency, and satisfaction given a specific use context [26].

In creating software, it is important to consider the user experience because a successful product depends on the software's ease of use, efficiency, and usefulness. The importance of evaluating the user experience in generative design applications arises because the relationship between generative design and user experience has been little explored. The interaction between the user and the generative design application must be positive considering its interface and system functionalities; access to the software's full potential must be enabled; that is, the user must be the one who designs the processes instead of using and designing processes [27]. Generative design applications must have the ability for the user to complete their objectives; they must also have a correct representation of data without ambiguities so that their understanding and analysis are carried out correctly.

2.3. Heuristic evaluation

Heuristic evaluation is an inspection method proposed by Nielsen [14]. This method consists of a group of three to five expert evaluators inspecting a user interface for detecting usability/UX problems and judging the interface elements following usability principles (heuristics) [28]. In this context, heuristics are general metrics that allow usability/UX problems to be detected [28]. A set of heuristics allows evaluating different aspects of the product, system, or service, considering the fulfilment of UX and usability attributes. Since UX considers more attributes to improve the UX of using a product, system, or service, a set of heuristics was created that evaluate UX in addition to usability.

2.4. Related work

We performed a literature review to find sets of existing heuristics or principles related to UX evaluation (see Section 4.1 and Appendices A and B). The 10 Nielsen heuristics are the most recognized and widely used set of heuristics [29] created to evaluate software interfaces. Each heuristic has a name, description, and example of use. The set evaluates general aspects of a product, system, or service, such as interface aspects, system efficiency, and user support. However, the Nielsen heuristics [29] can be generic for evaluating specific aspects of a product system or service.

Pribeanu [30] presented 14 heuristics to evaluate interactive systems. Each heuristic has a title, statement, and usability guidelines. The set was divided into four ergonomic criteria: user guidance, user effort, user control and freedom, and user support. The set evaluates ergonomic criteria, the physical, perceptual, and cognitive effort required to learn how to use the system, and the effort required to use it effectively. The heuristics have good coverage, which is achieved with a small number of heuristics. However, heuristics do not evaluate specific domains.

Gerhardt Powals [31] proposed 10 principles of cognitive engineering. Each principle has a name and definition. These principles focus on cognitive psychology to design and develop engineering systems to support users' cognitive processes. These principles are applicable to engineering systems. However, they did not evaluate specific systems.

Shneiderman [32] presented eight golden rules for better interface design. These eight rules have a name and description. The rules allow for proper guidance during the design process. These rules only focus on the user interface and do not consider other aspects of a software system.

All related studies present interesting proposals related to ergonomics, user interfaces, and cognitive engineering. However, none of these studies have focused on specific domains related to generative design applications. However, related works do not mention the iterations or validations performed. Our research had different validations and three iterations for the creation of the set. Moreover, it focuses on generative design applications. Therefore, a new set is established by adapting heuristics and existing works and creating new heuristics to evaluate domain-specific features.

3. Methodology applied to develop UXH-GEDAPP

For developing UXH-GEDAPP, we used the methodology proposed by Quiñones et al. [17,18]. We decided to apply this methodology since its stages are well explained and propose both qualitative and quantitative validation methods. The methodology has eight stages that can be applied iteratively to refine and improve the proposed heuristics [17,18]:

1. Exploratory stage: to perform a literature review to gather information on the specific domain, usability/UX, its characteristics and attributes and related work.
2. Experimental stage: to conduct experiments to gather information and complement it with that identified in the previous stage.
3. Description stage: to select and prioritize the most relevant information identified in the previous stages.
4. Correlational stage: to match domain-specific characteristics with usability/UX attributes and related work.
5. Selection stage: to maintain, adapt, or eliminate the related heuristics that were previously selected.
6. Specification stage: to specify and define the new set of heuristics.
7. Validation stage: to validate the set of heuristics through different experiments (heuristic evaluation, expert judgment, user testing).
8. Refinement stage: to refine and improve the proposed set of heuristics based on the feedback obtained in the validation stage.

4. Applying the methodology to develop UXH-GEDAPP

To develop UXH-GEDAPP, we performed 3 iterations and validated the set in 2 iterations through: heuristic evaluation, expert judgment, and user test (thinking-aloud test). Fig. 1 shows the steps made in each of the 3 iterations (iterations are marked as "It. N").

In iteration 1, seven methodology stages were carried out (exploratory, descriptive, correlational, selection, specification, validation, and refinement stage), and heuristics were developed and validated. In iteration 2, we performed the seven methodology stages once again, incorporating additional information contributing to the heuristic set. Finally, iteration 3 focused on specifying the final set of heuristics. Details of the inputs, outputs, and activities performed at each stage can be reviewed in Appendices A (first iteration), B (second iteration), and C (third iteration). Appendix D presents the set of heuristics for the generative design applications developed in each iteration. Each set is abbreviated differently for each iteration: HGD (first version, iteration 1), UXH-GD (second version, iteration 2), and UXH-GEDAPP (third version, iteration 3). The iterations performed are explained below.

4.1. First iteration: development process for HGD

In the first iteration, we created seven stages of methodology. In "Step 1: Exploratory stage", we conducted a literature review to obtain information about the generative design (see Section 2.1), user experience attributes (see Section 2.2), usability, set of existing heuristics, and related elements (specifically principles for cognitive engineering, see Section 2.4). To conduct the literature review, we considered ScienceDirect, Scopus, and Google Scholar databases. We searched the terms "heuristic evaluation", "usability", "user experience", "heuristics", and "generative design". Using the results obtained, we read the most relevant papers and documents whose titles and abstracts were related to the area [7,11,15,16,19,20,22,23,29,30,31]. Based on a literature review, we found that there is no specific heuristics to evaluate generative design applications. In addition, with the information collected, we defined the features of generative design. We decided not to do the "Step 2: Experimental stage" since step 1 provided us with enough information.

In "Stage 3: Descriptive stage", we grouped the information collected from Step 1 (definitions and features of generative design, usability and UX attributes, heuristics, and related elements). Then, we ordered and prioritized the information for each topic using a three-level scale. Finally, based on the assigned prioritization, we selected the following

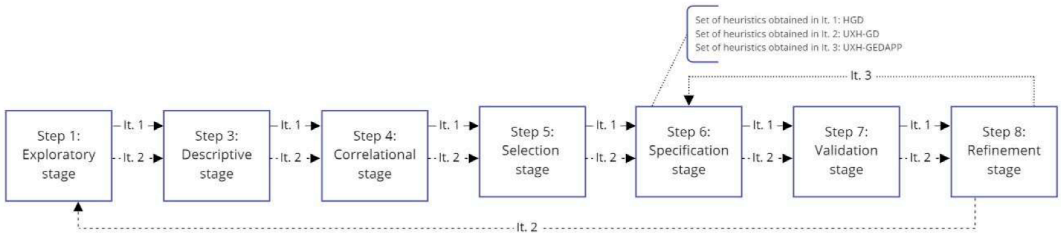

Fig. 1. Iterations and stages performed to develop the user experience heuristics for evaluation generative design applications (UXH-GEDAPP).

information to develop the heuristics (the detail of selected and discarded information can be reviewed in Appendix E):

- 9 generative design features [22,23]: Adaptive, optimizable, autonomous, efficient, innovative, customizable, flexible, multiplicity of results, and selective;
- 6 UX attributes [25]: useful, usable, findable, credible, valuable, and desirable;
- 2 two existing sets of heuristics: Nielsen [29] and Pribeanu [30] proposals; and
- 1 related element: Principles of cognitive engineering, by Gerhardt-Powals [31]

In “Stage 4: Correlational stage”, we matched generative design features, UX attributes, a set of existing heuristics and related items. When performing the matching, we found that there is no heuristic/principle that completely covers any feature of generative design applications (they slightly/partially cover some aspects). However, several heuristics are useful for generating the new sets (for more detail about the matching process, see Appendix F). In “Stage 5: Selection stage”, the sets of heuristics and elements selected in step 3, we determined the following actions: adapt, keep, or delete. We kept 8 heuristics, adapted 10 heuristics, and adapted 7 principles to evaluate general and domain-specific features. We eliminated 6 heuristics and 3 principles because they were not related to the features of generative design applications. Although there are no heuristics that cover the generative design features completely, mixing several proposals and adding a better specification will be very useful for creating a new set (for more details about the selection process, see Appendix G).

In “Stage 6: Specification stage”, we proposed the first version of the set of heuristics that evaluate generative design (HGD) applications (Appendix D). This version included 11 heuristics. We used the template proposed in the methodology for the specifications. The following elements were not incorporated: priority, benefits, problems, and checklist (incorporated in iteration 2). These elements were not incorporated because we decided to prioritize the main elements of the HGD set as the first version.

In “Step 7: Validation stage”, we validated the first version of heuristics through heuristic evaluation and expert judgment. Details of the validations performed can be reviewed in Sections 5.1 and 5.2 respectively. Finally, in “Stage 8: Refinement stage”, we refined the heuristics based on the results and feedback obtained in step 7 as follows:

- We refined 6 heuristics and improved their specifications, given unclear definitions and explanations (HGD1: “Adaptability of information”, HGD2: “Optimization of processes”, HGD4: “System efficiency”, HGD7: “Flexibility of use”, HGD8: “Multiplicity of results”, and HGD9: “Results selection”).
- We merged the heuristic HGD3: “Development autonomy” with HGD2: “Optimization of processes” because they were confused with each other.
- We merged the heuristic HGD6: “Personalization and data” with HGD7: “Flexibility of use” because they were confused with each other.
- We kept 3 heuristics without changes (HGD5: “Aesthetic and innovative design”, HGD10: “Error management”, and HGD11: “Help and documentation”).

We then decided to perform a second iteration by repeating the seven steps of the methodology.

4.2. Second iteration: development process for UXH-GD

In the second iteration, we performed seven methodology stages; again, we decided not to perform Step 2 (experimental stage) because we had the necessary information to develop the set. We repeated the first five stages of the methodology because we considered the incorporation of new information for developing the set of heuristics. In “Step 1: Exploratory stage”, we investigated new information relevant to the research (areas of application of generative design, relationship between UX and generative design, 7 UX design principles [33]; and 8 rules for interface design [32]).

In “Step 3: Descriptive stage”, we grouped, sorted, and prioritized the information collected from Stage 1. Subsequently, based on the assigned prioritization, we selected the 8 rules proposed by Shneiderman [32], to improve the development of heuristics (the detail of selected and discarded information in the second iteration can be reviewed in Appendix H).

In “Step 4: Correlational stage”, we matched the generative design features, UX attributes, and the 8 rules for interface design [32] (for more detail about the matching process, see Appendix I). In “Step 5: Selection stage”, we kept 3 elements and adapted 5 elements from the Shneiderman proposal [32] (see Appendix J).

In “Step 6: Specification stage”, we proposed a second version of the set of heuristics that evaluate generative design applications (UXH-GD). The UXH-GD set had 9 heuristics. We considered the refinement performed in the first iteration and incorporated the information gathered from the second iteration (from Stage 1 to Stage 5, see Appendix D). In this version, we incorporated all the template elements proposed in the methodology to specify the heuristics.

In “Step 7: Validation stage”, we validated the second version of heuristics through heuristic evaluation, expert judgment, and user test. Details of the validations performed can be reviewed in Sections 5.3, 5.4, and 5.5. respectively. Finally, in “Step 8: Refinement stage” based on the results obtained in step 7, we refined 5 heuristics given unclear definitions and explanations (UXH-GD2: “Optimization and development autonomy”, UXH-GD3: “System efficiency”, UXH-GD5: “Flexibility of use”, UXH-GD6: “Multiplicity of results”, and “UXH-GD7: Results selection”).

4.3. Third iteration: development process for the final set UXH-GEDAPP

In the final iteration, we performed “Step 6: Specification stage” to propose the final version of UXH-GEDAPP (see Appendices D and K). This version included 9 heuristics, improving the heuristics specification. We refined the heuristics based on the results obtained in validations performed in the second iteration. The final version of UXH-GEDAPP is presented in Section 6.

5. Validation of UXH-GEDAPP

We validated UXH-GEDAPP in two iterations through expert judgment (first iteration), heuristic evaluation (first and second iteration), and user tests (second iteration). The results are presented below.

5.1. First iteration: validation through heuristic evaluation

To perform the heuristic evaluation, six evaluators were divided into two groups (control and experimental group). The experimental group used the first version of heuristics proposed (HGD), while the control group used the Nielsen heuristics (NH). Each group comprised three evaluators with similar levels of expertise (the evaluators were distributed to ensure that the level of experience between the two groups was balanced).

The evaluators were contacted via email and LinkedIn and were selected based on their experience in UX research. All evaluators had a bachelor's degree in computer science (two evaluators had a master's degree, two were PhD students in computer science, and two were computer science engineers); and had moderate to high expertise in HCI and experience performing heuristic evaluations (they performed a heuristic evaluation on a regular basis between 2 and 4 years). The authors of this article did not participate as evaluators in any of the groups formed; they only used the results to refine heuristics.

Both groups evaluate a generative design application for "dimensioning retaining walls" [7,34]. The application provides different retaining wall designs based on the parameters given by the user with generative design. Both groups of evaluators evaluated this application under the same conditions and using the protocol for heuristic evaluations proposed by Nielsen [28]. Based on the methodology applied to develop the heuristics [17,18], the results obtained by both groups in the heuristic evaluation are compared using five criteria to evaluate the effectiveness of the new set of heuristics (see Table 1):

Table 1
Five criteria for evaluating the effectiveness of a new set of usability/UX heuristics (adapted from Refs. [17,18,35]).

Criterion description	Formula
1. Numbers of correct and incorrect associations of problems to heuristics	where: - CA: correct associations - IA: incorrect associations - T: total number of heuristics of the set - CAHn: number of correct associations of the problems to the heuristic "n" - IAHn: number of incorrect associations of the problems to the heuristic "n" - TP: total usability/UX problems identified
1. Number of usability/UX problems identified	- P1 = Problems that are identified by both groups of evaluators (common problems identified by both groups) - P2 = Problems that are identified only by the group that used the new set of heuristics (without considering the common problems) - P3 = Problems that are identified only by the group that used control heuristics (without considering the common problems)
1. Number of specific usability/UX problems identified	where: - ESS: effectiveness - NSP: number of specific usability/UX problems identified - TP: total usability/UX problems identified
1. Number of identified usability/UX problems that qualify as more severe (how catastrophic the usability/UX problem detected is)	where: - ESV: effectiveness - NPV: number of usability/UX problems identified qualified with a severity greater than 2 - TP: total usability/UX problems identified
1. Number of identified usability/UX problems that qualify as more critical (how severe and frequent the problem detected is)	where: - ESC: effectiveness - NPC: number of usability/UX problems identified qualified with a criticality greater than 4 - TP: total usability/UX problems identified

Table 2 shows the results obtained in the heuristic evaluations performed by the experimental and control groups. In addition, the effectiveness of HGD in terms of the five criteria is shown. As shown in Table 2, HGD performed better than NH on four of the five criteria. HGD detected more usability/UX problems than NH and, in addition, detected more specific and serious problems. It should be emphasized that NH detected more critical problems than HGD; this indicates that it requires refinement.

5.2. First iteration: validation through expert judgment

We applied a survey to the three evaluators (one evaluator had a master's degree, one was a PhD student in computer science, and one was a computer science engineer) who used the proposed HGD set in the heuristic evaluation conducted in the first iteration (the experimental group, see Section 5.1). The survey was designed to capture the evaluators' perception of the proposed set of heuristics set along four dimensions: Usefulness, Clarity, Ease of use, and Need for a checklist. We used a five-point Likert-type scale (1 being the worst rating and five being the best).

Table 3 shows the average values obtained for each dimension per heuristic. In general, the heuristics were rated out of 3, which means that the evaluators considered that the heuristics were understandable, useful, and clear. They also considered it necessary to incorporate a checklist. The lowest rated heuristic was HGD2 (Process Optimization); this indicates that it is not very understandable, unclear, and difficult to use; therefore, it requires refinement. On the other hand, heuristics HGD2 (Process optimization), HGD3 (Development autonomy), HGD8 (Multiplicity of results) and HGD9 (Results selection) were perceived as difficult to use. Heuristics HGD3 (Autonomy of development) and HGD8 (Multiplicity of results) are considered unclear. HGD6 (Personalization and data) is considered not so useful within the set of heuristics. Based on these results, the heuristics above were refined in the second iteration (step 6: specification stage).

5.3. Second iteration: validation through heuristic evaluation

We performed a second heuristic evaluation in the second iteration but using a different case study. We decided to change the case study to test and analyze the behavior of the UXH-GD set with a different application. The case study "Distribution of tables and desks within a defined space with generative design" (configured in Revit 2021 by a research academic expert in generative design) consists in distributing in automated and generative ways tables and desks in a defined space based on the parameters given by the user. Six evaluators were divided into two groups (experimental group and control group) for the heuristic evaluation. The experimental group used the proposed UXH-GD set of heuristics (the second version of heuristics), and the control group used the Nielsen set (NH) [29].

The evaluators were distributed to ensure that the level of experience between the two groups was balanced. In addition, the evaluators who participated in this evaluation were different from those who participated in the first heuristic evaluation (performed in the first iteration and presented in Section 5.1). However, all evaluators had experience in UX, computer science, and heuristic evaluations (they performed heuristic evaluations on a regular basis between 1 and 4 years). They were contacted via email based on their experience in UX research. All evaluators had a bachelor's degree in computer science (two evaluators had

Table 2
Effectiveness of HGD (first iteration).

	Experimental Group	Control Group	Observations
Amount of evaluators	3	3	–
Set of heuristics used	Heuristics for evaluating generative design applications (HGD)	Nielsen's heuristics (NH)	–
Amount of heuristics (T)	11	10	–
Total of problems identified (TP)	27	23	–
Number of specific problems identified (NSP)	11	8	–
Number of problems identified and qualified with a severity greater than 2 (NPV)	13	9	–
Number of problems identified and qualified with a criticality greater than 4 (NPC)	17	16	–
Problems identified by both groups (P1)	8		(P2) identified more problems than (P3), it is concluded that the proposed set performs better than the Nielsen set.
Problems identified by the experimental group (P2)	19	–	
Problems identified by the control group (P3)	–	15	
Total of the correct associations ($\sum$ **CAHn**)	19	16	–
Total of the incorrect associations ($\sum$ **IAHn**)	8	7	–
Percentage of the correct associations (CA)	CA1 = 70.37 %	CA2 = 69.56 %	CA1>CA2 it is concluded that the proposed set performs better than the Nielsen set, as it has a higher percentage of correct associations.
Percentage of the incorrect associations (IA)	IA1 = 29.62 %	IA2 = 30.43 %	IA1<IA2 it is concluded that the proposed set is better, since the Nielsen set has a higher percentage of incorrect associations.
Effectiveness in terms of number of specific problems identified (ESS)	ESS1 = 40.74 %	ESS2 = 34.78 %	ESS1>ESS2 the proposed set identified more specific problems than the Nielsen set, then it works better
Effectiveness in terms of number of problems identified and qualified with a severity greater than 2 (ESV)	ESV1 = 48.15 %	ESV2 = 39.13 %	ESV1>ESV2 it is concluded that the proposed set is better, since it finds more problems rated as severe than the Nielsen set.
Effectiveness in terms of number of problems identified and qualified with a criticality greater than 4 (ESC)	ESC1 = 62.96 %	ESC2 = 69.57 %	ESC1<ESC2 it is concluded that the Nielsen set encounters more problems rated as critical than the proposed set (requiring refinement).

Table 3
Average perception scores for HGD set in the four evaluated dimensions (first iteration).

Heuristic	**D1—Utility**	**D2—Clarity**	**D3—Ease of Use**	**D4—N. of Add. Elem.**
HGD1: Adaptability of information	4.66	4.33	4	5
HGD2: Optimization of processes	4	3.66	2.66	5
HGD3: Development autonomy	4	3.33	3	5
HGD4: System efficiency	4.66	3.66	3.66	5
HGD5: Aesthetic and innovative design	4.33	4.66	4.66	5
HGD6: Personalization and data	3.33	4	3.66	5
HGD7: Flexibility of use	3.66	3.66	3.66	5
HGD8: Multiplicity of results	4	3.33	3	5
HGD9: Results selection	4	3.66	3.33	5
HGD10: Error management	5	4.66	4.66	5
HGD11: Help and documentation	5	4.66	4.66	5
Average	4.24	3.96	3.72	5

a bachelor's degree, two were PhD students in computer science, and two were computer science engineers). The authors of this article did not participate as evaluators in any of the groups formed; they only used the results to refine heuristics.

The results obtained show that the proposed set is better and more effective in the five criteria evaluated according to the methodology (see Table 4). We compared the problems identified by both groups based on five criteria presented in Table 1.

5.4. Second iteration: validation through expert judgment

We applied another survey to the 3 evaluators (one evaluator had a bachelor's degree, one was a PhD student in computer science, and one was a computer science engineer) who used the UXH-GD set in the heuristic evaluation conducted in the second iteration (the experimental group, see Section 5.3). The survey was designed to capture once again the evaluators' perception of heuristics considering four dimensions: Usefulness, Clarity, Ease of use, and Need for a checklist. We used a five-point Likert-type scale (1 being the worst rating and 5 being the best).

Table 5 shows the average values obtained for each dimension per heuristic. In general, the heuristics were rated out of 3, which means that the evaluators consider that the heuristics are understandable, useful, and clear. They also consider it necessary to incorporate a checklist. The lowest rated heuristic was UXH-GD7 (Results selection), so it requires refinement to improve its understanding. On the other hand, the average per dimension was above 4; this means that the set of heuristics is understandable, clear, and easy to use. Finally, the heuristics UXH-GD2 (Optimization and development), UXH-GD3 (System efficiency), UXH-GD6 (Multiplicity of results) and UXH-GD7 (Results selection) should

Table 4
Effectiveness of UXH-GD (second iteration).

	Experimental Group	Control Group	Observations
Amount of evaluators	3	3	–
Set of heuristics used	Heuristics for evaluating generative design applications (UXH-GD)	Nielsen Heuristics (NH)	–
Amount of heuristics (T)	9	10	–
Total of problems identified (TP)	26	21	–
Number of specific problems identified (NSP)	10	4	–
Number of problems identified and qualified with a severity greater than 2 (NPV)	9	7	–
Number of problems identified and qualified with a criticality greater than 4 (NPC)	11	8	–
Problems identified by both groups (P1)	7		(P2) identified more problems than (P3), it is concluded that the proposed set performs better than the Nielsen set.
Problems identified by the experimental group (P2)	19	–	
Problems identified by the control group (P3)	–	14	
Total of the correct associations (∑ **CAHn**)	23	14	–
Total of the incorrect associations (∑ **IAHn**)	3	7	–
Percentage of the correct associations (CA)	CA1= 88.46 %	CA2= 66.67 %	CA1>CA2 it is concluded that the proposed set performs better than the Nielsen set, as it has a higher percentage of correct associations.
Percentage of the incorrect associations (IA)	IA1= 11.54 %	IA2= 33.33 %	IA1<IA2 it is concluded that the proposed set is better, since NH has a higher percentage of incorrect associations.
Effectiveness in terms of number of specific problems identified (ESS)	ESS1= 38.46 %	ESS2= 19.05 %	ESS1>ESS2 the proposed set identified more specific problems than NH, then it works better
Effectiveness in terms of number of problems identified and qualified with a severity greater than 2 (ESV)	ESV1= 34.62 %	ESV2= 33.33 %	ESV1>ESV2 it is concluded that UXH-GD is better, as it finds more problems rated as severe than NH
Effectiveness in terms of number of problems identified and qualified with a criticality greater than 4 (ESC)	ESC1= 42.30 %	ESC2=38.10 %	ESC1>ESC2 it is concluded that UXH-GD is superior to NH, since it identifies more problems rated as critical than NH.

Table 5
Average perception scores for UXH-GD set in the four evaluated dimensions (second iteration).

Heuristic	**D1—Utility**	**D2—Clarity**	**D3—Ease of Use**	**D4—N. of Add. Elem.**
UXH-GD1: Adaptability of information	4.66	4.33	4.33	5
UXH-GD2: Optimization and development autonomy	4.33	4.66	3.66	5
UXH-GD3: System efficiency	4.33	4.33	4.33	5
UXH-GD4: Aesthetic and innovative design	4.33	4.33	4.66	5
UXH-GD5: Flexibility of use	4.66	4.66	4.33	5
UXH-GD6: Multiplicity of results	4.66	4	3.66	5
UXH-GD7: Results selection	4.33	4	3.66	5
UXH-GD8: Error management	5	4.66	4	5
UXH-GD9: Help and documentation	5	4.66	4.66	4.66
Average	4.59	4.40	4.14	4.96

be refined since they had a lower rating than the average.

5.5. Second iteration: validation through user test

In the second iteration, we conducted a thinking-aloud test to evaluate the same case study used in the second iteration ("Distribution of tables and desks within a defined space with generative design", configurated in Revit 2021). In the thinking aloud method, the evaluators observe and record how the user expresses his thoughts and opinions while interacting with the system through a set of assigned tasks [36, 37]. The test was conducted online via videocall using the "Zoom" platform. "TeamViewer" software was used for remote access to the case study system. The users freely explored the system performing the assigned tasks while being monitored (supervised) by two evaluators. The participants were 4 users with knowledge in civil engineering and generative design with an age range of 25–29 years with prior knowledge regarding the domain. They were civil engineers and researchers working at the School of Civil Engineering at the Pontificia Universidad Católica de Valparaíso, Chile. Additionally, all users had basic knowledge of using the Revit tool. The authors of this article only supervised the test and did not participate as users in the testing.

The experiment was designed to: (1) validate that the problems found in the heuristic evaluation (second iteration) are perceived as real problems by the users; and (2) identify new usability/UX problems not detected in the heuristic evaluation and check if the UXH-GD set considers them. Before testing, we conducted a pilot test to estimate the time for each task. The user test was divided into four parts: (1) The activity instructions and confidentiality agreement so that the user is aware of the experiment; (2) a preliminary questionnaire to identify the user's profile and previous experience with generative design applications; (3) A list of tasks to complete (tasks were defined based on the problems detected in the heuristic evaluation performed in the second iteration); and (4) A post-experiment satisfaction questionnaire to know the user's perception using the system.

During the performance of the tasks by the users, the supervisors (the authors of this article) recorded (1): the time required for each task; (2) the number of tasks completed successfully; and (3) the comments that users gave out loud about the problems they had related to ease of use, interaction, navigation, and difficulties in completing tasks.

Based on: (1) verbal comments made by users during the execution of tasks and use of the application; and (2) the responses provided by users in the post-experiment satisfaction questionnaire, several usability/UX issues were identified and documented. Table 6 shows the tasks performed by the users in the thinking-aloud test and their results.

Based on the users' performance of the tasks, 14 usability/UX problems were identified. We noticed that problems P1, P2, P5, P6, P7 and P9 are perceived as the most critical problems for users, as it was difficult for them to complete the related tasks. We reviewed whether UXH-GD allows the detection of the identified problems (P1 to P14), concluding that UXH-GD covers all problems detected in the thinking-aloud test (see Table 6).

6. UXH-GEDAPP: user eXperience heuristics for evaluating generative design applications

Based on the iterations and validations described in the previous section, the UXH-GEDAPP set was refined and improved. We proposed 9 heuristics. Of all the proposed heuristics, 5 heuristics are brand-new heuristics created to evaluate specific features of generative design applications (UXH-GEDAPP-1, UXH-GEDAPP-2, UXH-GEDAPP-4, UXH-GEDAPP-6, UXH-GEDAPP-7).

The 9 heuristics are presented in Table 7, including 5 elements: ID; name; description; the generative design features evaluated with the heuristic; and the UX attributes evaluated with the heuristic. In addition, in Appendix K each heuristic is fully presented using the template specified in the methodology applied [17,18]. Each heuristic is presented as a table that includes 12 elements: ID; name; definition; explanation; priority (how important the heuristic is: critical, important, or useful); generative design features (features evaluated with the heuristic); UX attribute related (attributes evaluated with the heuristic); problems (anticipated problems for evaluators of confusing heuristics and misunderstanding); benefits (expected usability/UX benefits when the heuristic is satisfied); checklist; examples (examples with images of compliance with the heuristic); and existing heuristics or other elements related (set on which the heuristic is based, if exist, and the heuristic that evaluates to some degree a certain general feature or a specific feature of generative design applications).

7. Contributions

UXH-GEDAPP is an innovative evaluation instrument that incorporate the concept of generative design. We found that no instruments have been developed that focus on evaluating the user experience in this area [15,16]. We believe that this new set of heuristics contributes to promote the development and use of generative design applications considering UX.

The set was validated in 2 iterations using various methods and different case studies. This allowed us to refine the set and ensure its effectiveness. UXH-GEDAPP is an evaluation instrument that can be used to detect usability and UX problems in generative design applications. The proposed set will allow to improve the UX prior to the development of generative design applications avoiding interface errors. In addition, this set allows to detect potential UX problems that can be corrected, so that the applications can be more pleasant and intuitive during the user interaction.

8. Conclusions

Generative design is used to optimize and improve the design processes. To increase and improve user interaction, it is important to make it intuitive and enjoyable for both future and current users. Recently, it has become more popular in the engineering field because traditional design methodologies are inflexible, resulting in inefficient products with high operating costs. Evaluating UX is necessary to identify which elements hinder user interaction with generative design applications, detect problems, and improve UX.

We proposed a set of 9 heuristics to evaluate the UX in generative design applications: UXH-GEDAPP. The heuristics were developed using a formal methodology [17,18] and consider both generative design

Table 6
Thinking aloud test quantitative results (second iteration).

Task (T)	Percentage of Task Fulfillment	Observations	Usability/UX problems detected in user test (P)	Heuristic related (second version, UXH-GD)
T1: Distribution of tables in a common space on a single floor	50 % (2 of 4 users)	All users made a mistake in the first choice, then 50 % of them remedied the error and performed the task correctly. All users cannot find the filters and find it difficult to use them.	P1: The system does not give feedback. P2: Options have the identical name, but no obvious differentiation. P3: The system does not have a tutorial to help the user. P4: The system does not indicate the number of iterations performed. P5: The names of the filters are not very visible.	P1 is covered by UXH-GD8 P2 is covered by UXH-GD4 P3 is covered by UXH-GD9 P4 is covered by UXH-GD7 P5 is covered by UXH-GD4
T2: Distribution of desks in a classroom 1	50 % (2 of 4 users)	2 users were unable to perform the task correctly. Rotation of items was difficult for all users.	P6: Solution filtering is not intuitive. P7: The "rotate" option is unintuitive. P8: The system does not inform about generated errors.	P6 is covered by UXH-GD3 P7 is covered by UXH-GD3 P8 is covered by UXH-GD8
T3: Distribution of desks within a classroom 2	50 % (2 of 4 users)	2 users did not perform the task correctly. One user could not perform the rotation of the element.	P9: The solution filtering option is inaccurate. P10: The system does not indicate the number of solutions generated.	P9 is covered by UXH-GD5 P10 is covered by UXH-GD6
T4: 3D visualization of the generated space	100 % (4 of 4 users)	All users performed the task correctly.	P11: The system does not allow to edit a "study" already created.	P11 is covered by UXH-GD1

Table 7
UXH-GEDAPP: User eXperience Heuristics for evaluating GEnerative Design APPlications.

ID	Name	Description	Generative design feature	UX attribute
UXH-GEDAPP-1	Adaptability of information	The system should allow the input and modification of any parameter set by the user, according to the context of the software and adapting to the needs and user requirements.	Adaptative	Useful, Usable
UXH-GEDAPP-2	Optimization and development autonomy	The system should be autonomous in operation, optimizing and automating calculation and design processes, reducing the user's cognitive load during the development process.	Optimizable, Autonomous	Valuable, Useful, Credible
UXH-GEDAPP-3	System efficiency	The system should perform the functionalities requested by the user in an effective way. The functionalities should be visible, traceable, intuitive and with reasonable response times.	Efficient	Usable
UXH-GEDAPP-4	Aesthetic and innovative design	The system should have an intuitive and user-friendly interface. The system outputs should be novel, creative, and understandable to the user.	Innovative	Valuable, Desirable
UXH-GEDAPP-5	Flexibility of use	The system should be used by different types of users. The system should be adapted to different user profiles, making its use simple and intuitive. In addition, the system should have different ways to perform a functionality, i.e., it should be configurable by the user.	Customizable, Flexible	Usable, Findable
UXH-GEDAPP-6	Multiplicity of results	The system should provide a set of solutions in a generative and iterative way, based on the parameters provided by the user.	Multiple results	Useful, Valuable
UXH-GEDAPP-7	Results selection	The results provided by the system are ranked based on the user's needs, best fitting the parameters provided by the user.	Selective	Useful, Valuable
UXH-GEDAPP-8	Error management	The system should provide ways to prevent, diagnose, correct, and recover from errors.	Efficient	Usable, Credible
UXH-GEDAPP-9	Help and documentation	The system should provide help and documentation of how the system works in a place that is easily accessible to users.	Flexible	Credible

application features and UX attributes for its specification. For its development, three iterations were carried out with 3 types of experimental validations: heuristic evaluation, expert judgment, and user testing. Based on the results obtained in the validations, we concluded that UXH-GEDAPP is effective and covers the features of generative design. During the validations of the first and second iterations, the set of heuristics was perceived as useful and effective by the evaluators who participated in this research. Specifically, UXH-GEDAPP is considered effective since they allow to detect specific problems of generative design applications, as well as more severe problems related to this type of applications. On the other hand, evaluators make fewer errors associating the detected usability/UX problems with the proposed heuristics, compared to generic sets. In the third iteration we refined and proposed the final version of UXH-GEDAPP.

We believe that UXH-GEDAPP can be used to detect potential UX problems and subsequently correct these problems to improve the user experience. On the other hand, they can be a guide for the development of generative design applications and consider a pleasant and intuitive experience for potential users.

The limitations encountered during this research were the limited number and access to case studies for the validation of the set and information regarding the synergy between generative design and UX. However, we believe that the case studies used to validate the heuristics were adequate and useful for refining the set. Furthermore, other limitations are related to the number of users who participated in the thinking-aloud test (4 users) and the profile of the evaluators who performed heuristic evaluations.

Nielsen mentions that 5 users are enough for a user test [38]; and also recommends running as many small tests as you can afford [39]. However, owing to time and resource issues (access to users and budget), we were only able to perform one test with four users. To mitigate these limitations, for future experiments, it is very important to perform several small tests (as much as possible with the available resources), with a group of five users in each test (as mentioned by Nielsen in [38] and [39]), to obtain interesting insights and the maximum benefit-cost ratio [38]. Despite the above, we believe that the information obtained was valuable in verifying that user problems can be identified with the UXH-GEDAPP in a heuristic evaluation.

On the other hand, the evaluators who carried out the heuristic evaluations are experts who have been working in the UX area and carrying out heuristic evaluations for between 1 and 4 years. Nevertheless, they were all evaluators with experience in the field and knew how to carry out a correct evaluation (each evaluator had performed at least 5 heuristic evaluations).

As future work, we expect to innovate and propose an improvement of the set that allows it to be flexible in evaluating any area that applies generative design. For the above, we hope to carry out new evaluations and user tests with a larger number of participants.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The authors would like to thank all the participants (experts, users, evaluators, and researchers) who were involved in the experiments for this study. Luis Felipe Rojas is supported by Grant ANID BECAS/DOCTORADO NACIONAL, Chile, No. 21211272.

Appendix

Appendix A. Inputs, outputs, and activities for each step performed in iteration 1

See Table 8

Table 8
Inputs, outputs, and activities for each step performed in iteration 1.

Step	Input	Output	Activities Performed
Step 1: Exploratory stage	Generative design applications domain	① Information about generative design applications (five definitions, nine features, and the process of generative design); ② one proposal for usability attributes and one proposal for UX attributes [25]; and ③ two existing sets of heuristics related [29,30], and one related element (principles of cognitive engineering, [31])	Conduct a literature review about generative design applications (definitions and features); usability/UX attributes; existing sets of usability/UX heuristics related; and other relevant information
Step 3: Descriptive stage	① Information about generative design applications (five definitions, nine features, and the process of generative design); ② one proposal for usability attributes and one proposal for UX attributes; and ③ two existing sets of heuristics related and one related element (principles of cognitive engineering)	⑦ Selected information about generative design applications; ⑧ nine selected features of generative design applications; ⑨ six UX attributes (from one proposal); and ⑩ two selected sets of heuristics and one related element (principles of cognitive engineering)	Group the information collected. Sort and prioritize the information using a three-level scale (3: highly important; 2: somewhat important; 1: not important). Select the information to develop the heuristics.
Step 4: Correlational stage	⑦ Selected information about generative design applications; ⑧ nine selected features of generative design applications; ⑨ six UX attributes (from one proposal); and ⑩ two selected sets of heuristics and one related element (principles of cognitive engineering)	⑪ Matched features, attributes, existing heuristics, and related elements	Match the nine generative design application features with the six UX attributes, and two sets of heuristics (10 Nielsen heuristics [29], 14 interactive systems heuristics [30], and one related item (10 principles of cognitive engineering, [31]) The new heuristics will not be grouped into categories
Step 5: Selection stage	⑩ two selected sets of heuristics; ⑪ Matched features, attributes, existing heuristics, and related elements	⑬ Classified heuristics (8 heuristics to keep; 10 heuristics to adapt; 7 principles to adapt; 6 heuristics to eliminate; and 3 principles to eliminate)	Review Nielsen heuristics [29], interactive systems heuristics [30], and principles of cognitive engineering [31]; and determine what heuristics/principles to: keep, adapt, and eliminate. Identify which existing heuristics can be join into one
Step 6: Specification stage	⑪ Matched features, attributes, existing heuristics, and related elements; ⑬ Classified heuristics (8 heuristics to keep; 10 heuristics to adapt; 7 principles to adapt; 6 heuristics to eliminate; and 3 principles to eliminate)	⑭ Set of generative design application heuristics, HGD (first iteration)	Specify 11 UX heuristics for generative design applications (HGD), including: id, name, definition, explanation, generative design application feature, examples, UX attribute, and existing heuristics related (priority, benefits, problems, and checklist not included)
Step 7: Validation stage	⑭ Set of generative design application heuristics, HGD (first iteration)	⑮ Heuristic evaluation results: effectiveness of HGD; ⑯ Expert judgment results (survey)	Perform a heuristic evaluation to a case study ("dimensioning retaining walls with generative design" [7,34]) with six evaluators (three evaluators for the control group, three evaluators for experimental group) Evaluate the HGD effectiveness using five criteria for heuristic evaluation. Apply a survey to three experts to capture the evaluators' perception about HGD set in four dimensions (utility, clarity, ease of use, and need for additional elements—checklists)
Step 8: Refinement stage	⑮ Heuristic evaluation results: effectiveness of HGD; ⑯ Expert judgment results (survey)	⑱ Refining document: (1) 6 heuristics to refine, 4 heuristics to merge into 2 heuristics; (2) repeat steps 1, 3–8	Document the improvements to be performed in the specification of HGD It is decided to iterate repeating stages 1, 3–8

Appendix B. Inputs, outputs, and activities for each step performed in iteration 2

See Table 9

Table 9
Inputs, outputs, and activities for each step performed in iteration 2.

Step	Input	Output	Activities Performed
Step 1: Exploratory stage	① Information about generative design applications	① Information about generative design applications (2 major areas of application: architecture and engineering) ③ two related elements (7 UX design principles [33]; 8 rules for interface design [32])	Conduct a literature review about application areas of generative design, and relation between UX and generative design.
Step 3: Descriptive stage	① Information about generative design applications (2 major areas of application: architecture and	⑩ one related element (8 rules for interface design [32])	Group the information collected. Sort and prioritize the information using a three-level scale (3: highly important; 2: somewhat important; 1: not

(continued on next page)

Table 9 (*continued*)

Step	Input	Output	Activities Performed
	engineering); ③ two related elements (7 UX design principles [33]; 8 rules for interface design [32])		important). Select the information to develop the heuristics.
Step 4: Correlational stage	⑩ one related element (8 rules for interface design [32]) ⑪ Matched features, attributes, existing heuristics, and related elements	⑪ Matched features, attributes, existing heuristics, and related elements (updated)	Match the nine generative design application features with the six UX attributes, and one related item ([32]) The heuristics will not be grouped into categories
Step 5: Selection stage	⑩ one related element (8 rules for interface design [32]); ⑪ Matched features, attributes, existing heuristics, and related elements (updated)	⑬ Classified elements (3 rules to keep; 5 rules to adapt)	Review the 8 rules for interface design [32]; and determine what rule to: keep, adapt, and eliminate.
Step 6: Specification stage	⑪ Matched features, attributes, existing heuristics, and related elements (updated); ⑬ Classified elements (3 rules to keep; 5 rules to adapt); ⑭ Set of generative design application heuristics, HGD (first iteration); ⑱ Refining document: (1) 6 heuristics to refine, 4 heuristics to merge into 2 heuristics	⑭ Set of generative design application heuristics, UXH-GD (second iteration)	Refine and improve the specification of 9 UX heuristics for generative design applications (UXH-GD), including: id, priority, name, definition, explanation, generative design application feature, examples, benefits, problems, UX attribute, checklists, and existing heuristics related
Step 7: Validation stage	⑭ Set of generative design application heuristics, UXH-GD (second iteration)	⑮ Heuristic evaluation results: effectiveness of UXH-GD ⑯ Expert judgment results (survey) ⑰ User tests results: users' perceptions	Perform a heuristic evaluation to a case study with six evaluators (three evaluators for the control group, three evaluators for experimental group) Evaluate the UXH-GD effectiveness using five criteria for heuristic evaluation Perform a thinking-aloud test to evaluate a case study with four users. Based on the results: (1) validate if the UX problems identified in the heuristic evaluation (second iteration) are perceived as real problems for the users; and (2) identify new problems and review if these problems are evaluated by the UXH-GD set In both previous validation, the case study used was: "Distribution of tables and desks within a defined space with generative design" (configurated in Revit2022) Apply a survey to three experts to capture the evaluators' perception about UXH-GD set in four dimensions (utility, clarity, ease of use, and need for additional elements—checklists)
Step 8: Refinement stage	⑮ Heuristic evaluation results: effectiveness of UXH-GD; ⑯ Expert judgment results (survey); ⑰ User tests results: users' perceptions	⑱ Refining document: (1) five heuristics to refine; (2) repeat step 6	Document the improvements to be performed in the specification of UXH-GD It is decided to iterate repeating stage 6

Appendix C. Inputs, outputs, and activities for each step performed in iteration 3

See Table 10

Table 10
Inputs, outputs, and activities for each step performed in iteration 3.

Step	Input	Output	Activities Performed
Step 6: Specification stage	⑭ Set of generative design application heuristics, UXH-GD (second iteration); ⑱ Refining document: (1) five heuristics to refine; (2) repeat step 6	⑭ Set of generative design application heuristics, UXH-GEDAPP (third iteration)	Refine and improve the final specification of 9 UX heuristics for generative design applications (UXH-GEDAPP)

Appendix D. Set of heuristics for generative design applications developed at each iteration

See Table 11

Table 11
Set of heuristics for generative design applications developed at each iteration.

First Iteration (HGD)	Second Iteration (UXH-GD)	Third Iteration (UXH-GEDAPP)
HGD1: Adaptability of information	UXH-GD1: Adaptability of information	UXH-GEDAPP-1: Adaptability of information
HGD2: Optimization of processes	UXH-GD2: Optimization and development autonomy	UXH-GEDAPP-2: Optimization and development autonomy
HGD3: Development autonomy	UXH-GD3: System efficiency	UXH-GEDAPP-3: System efficiency
HGD4: System efficiency	UXH-GD4: Aesthetic and innovative design	UXH-GEDAPP-4: Aesthetic and innovative design
HGD5: Aesthetic and innovative design	UXH-GD5: Flexibility of use	UXH-GEDAPP-5: Flexibility of use
HGD6: Personalization and data	UXH-GD6: Multiplicity of results	UXH-GEDAPP-6: Multiplicity of results
HGD7: Flexibility of use	UXH-GD7: Results selection	UXH-GEDAPP-7: Results selection
HGD8: Multiplicity of results	UXH-GD8: Error management	UXH-GEDAPP-8: Error management
HGD9: Results selection	UXH-GD9: Help and documentation	UXH-GEDAPP-9: Help and documentation
HGD10: Error management		
HGD11: Help and documentation		

Appendix E: First iteration, Step 3: "Descriptive stage". Relevance for generative design application features, UX attributes, sets of existing heuristics, and related relevant elements

See Table 12

Table 12
First iteration, Step 3: "Descriptive stage". Relevance for generative design application features, UX attributes, sets of existing heuristics, and related relevant elements.

Topic	Value according to Relevance (3: Highly Important, 2: Somewhat Important, 1: Not Important)			Explanation
	3	2	1	
Generative design application feature	Adaptative, Optimizable, Autonomous, Efficient, Innovative, Customizable, Flexible, Multiple results, Selective	–	–	All features were selected since all of them were considered relevant.
Usability attribute	–	–	Learnability, Efficiency, Memorability, Errors, Satisfaction	One proposal for usability attributes was collected in Step 1, Nielsen [40], valued with grade 1 of importance. While the proposal is interesting, we decided to use the UX attributes proposed by Morville [25] since UX includes usability.
UX attribute	Useful, Usable, Desirable, Findable, Credible, Valuable	–	Accessible	One proposal for UX attributes were collected in step 1: Morville [25]. Of these, five attributes were selected. "Accessibility" attribute was not considered as it is a broad and complex topic, which would increase the number of heuristics considerably.
Set of heuristics	10 Nielsen's heuristics [29], 14 Pribeanu's heuristics [30]	–	–	Two sets of heuristics were selected [29,30].
Other elements	10 principles of cognitive engineering [31]	–	–	One related element was selected [31].

Appendix F: First iteration, Step 4: "Correlational stage". Match between the generative design application features, UX attributes, heuristics proposed by other authors, and related elements proposed by other authors

See Table 13

Table 13
First iteration, Step 4: "Correlational stage". Match between the generative design application features, UX attributes, heuristics proposed by other authors, and related elements proposed by other authors.

Feature	UX Attribute	Heuristic name or element related*
Adaptative	Useful, Usable	N3: User control and freedom [29] (slightly covered feature) P8: Explicit user actions [30] (slightly covered feature) P9: User control [30] (slightly covered feature) C9: Provide multiple coding of data [31] (partially covered feature)
Optimizable	Useful, Valuable	N6: Recognition rather than recall [29] (slightly covered feature) P6: Cognitive workload [30] (slightly covered feature) P7: Minimal actions [30] (slightly covered feature) C7: Limit data driven tasks [31] (slightly covered feature)
Autonomous	Useful, Credible	N7: Flexibility and efficiency of use [29] (partially covered feature) P7: Minimal actions [30] (partially covered feature) C1: Automate unwanted workload [31] (partially covered feature) C7: Limit data driven tasks [31] (slightly covered feature)
Efficient	Useful	N2: Match between system and the real world [29] (slightly covered feature) N7: Flexibility and efficiency of use [29] (partially covered feature) P11: Compatibility with the user [30] (slightly covered feature) C2: Reduce uncertainty [31] (slightly covered feature)
Innovative	Valuable, Desirable	N8: Aesthetic and minimalist design [29] (slightly covered feature) C4: Present new information with meaningful aids to interpretation [31] (slightly covered feature)
Customizable	Useful	N7: Flexibility and efficiency of use [29] (slightly covered feature) P10: Flexibility [30] (partially covered feature) C9: Provide multiple coding of data [31] (slightly covered feature)
Flexible	Useful, Findable	N7: Flexibility and efficiency of use [29] (partially covered feature) P10: Flexibility [30] (partially covered feature) C9: Provide multiple coding of data [31] (slightly covered feature)
Multiple results	Useful, Valuable	N1: Visibility of system status [29] (slightly covered feature) P2: Feedback [30] (slightly covered feature) P4: Grouping/Distinction [30] (slightly covered feature) C6: Group data in consistently, meaningful ways [31] (slightly covered feature)
Selective	Useful, Valuable	N1: Visibility of system status [29] (slightly covered feature) P2: Feedback [30] (slightly covered feature) P4: Grouping/Distinction [30] (partially covered feature) C6: Group data in consistently, meaningful ways [31] (slightly covered feature) C8: Include in the displays only that information needed by the operator at a given time [31] (slightly covered feature)

The letter "N" is used as ID to indicate the number of Nielsen's heuristics [29], the letter "P" for Pribeanu's heuristics [30], and the letter "C" for the principles of cognitive engineering [31].

Appendix G: First iteration, Step 5: "Selection stage". Heuristics and principles selection process

See Table 14

Table 14
First iteration, Step 5: "Selection stage". Heuristics and principles selection process.

ID	Heuristic/Principle Name	Action	Refs.	Generative design application feature covered	Applicability
N1	Visibility of system status	Adapt	[29]	Multiple results, Selective	(3) Critical
N2	Match between system and the real world	Adapt	[29]	Efficient	(2) Important
N3	User control and freedom	Adapt	[29]	Adaptative	(3) Critical
N4	Consistency and standards	Delete	[29]	The heuristic does not evaluate any feature	–
N5	Error prevention	Delete	[29]	The heuristic does not evaluate any feature	–
N6	Recognition rather than recall	Keep	[29]	Optimizable	(2) Important
N7	Flexibility and efficiency of use	Adapt	[29]	Autonomous, Efficient, Customizable, Flexible	(3) Critical
N8	Aesthetic and minimalist design	Adapt	[29]	Innovative	(1) Useful
N9	Help users recognize, diagnose, and recover from errors	Keep	[29]	Efficient	(3) Critical
N10	Help and documentation	Keep	[29]	Flexible	(2) Important
P1	Prompting	Delete	[30]	The heuristic does not evaluate any feature	–
P2	Feedback	Keep	[30]	Efficient	(2) Important
P3	Information architecture	Delete	[30]	The heuristic does not evaluate any feature	–
P4	Grouping/distinction	Adapt	[30]	Multiple results, Selective	(1) Useful
P5	Consistency	Delete	[30]	The heuristic does not evaluate any feature	–
P6	Cognitive workload	Keep	[30]	Optimizable	(2) Important
P7	Minimal actions	Adapt	[30]	Optimizable, Autonomous, Efficient	(1) Useful
P8	Explicit user actions	Keep	[30]	Adaptative	(1) Useful
P9	User control	Adapt	[30]	Adaptative	(3) Critical
P10	Flexibility	Adapt	[30]	Customizable, Flexible	(3) Critical
P11	Compatibility with the user	Adapt	[30]	Efficient	(2) Important
P12	Task guidance and support	Delete	[30]	The heuristic does not evaluate any feature	–
P13	Error management	Keep	[30]	Efficient	(3) Critical
P14	Help and documentation	Keep	[30]	Flexible	(2) Important
C1	Automate unwanted workload	Adapt	[31]	Autonomous	(2) Important
C2	Reduce uncertainty	Adapt	[31]	Efficient	(1) Useful
C3	Fuse data	Delete	[31]	The principle does not evaluate any feature	–
C4	Present new information with meaningful aids to interpretation	Adapt	[31]	Innovative	(1) Useful
C5	Use names that are conceptually related to function	Delete	[31]	The principle does not evaluate any feature	–
C6	Group data in consistently, meaningful ways	Adapt	[31]	Multiple results, Selective	(2) Important
C7	Limit data driven tasks	Adapt	[31]	Optimizable, Autonomous	(2) Important
C8	Include in the displays only that information needed by the operator at a given time	Adapt	[31]	Selective	(1) Useful
C9	Provide multiple coding of data	Adapt	[31]	Adaptative, Customizable, Flexible	(1) Useful
C10	Practice judicious redundancy	Delete	[31]	The principle does not evaluate any feature	–

Appendix H: Second iteration, Step 3: "Descriptive stage". Relevance for related relevant elements

See Table 15

Table 15
Second iteration, Step 3: "Descriptive stage". Relevance for related relevant elements.

	Value according to Relevance (3: Highly Important, 2: Somewhat Important, 1: Not Important)			
Topic	3	2	1	Explanation
Other elements	8 rules for interface design [32]	–	7 UX design principles [33]	One related element was selected [32]. The 7 UX design principles [33] are useful and interesting, but they were already covered by the elements included in Iteration 1.

Appendix I: Second iteration, Step 4: "Correlational stage". Match between the generative design application features, UX attributes, and related elements proposed by other authors

See Table 16

Table 16
Second iteration, Step 4: "Correlational stage". Match between the generative design application features, UX attributes, and related elements proposed by other authors.

Feature	UX Attribute	Element related*
Adaptative	Useful, Usable	R1: Strive for consistency [32] (slightly covered feature) R7: Keep users in control [32] (slightly covered feature)

(*continued on next page*)

Table 16 (*continued*)

Feature	UX Attribute	Element related*
Optimizable	Useful, Valuable	R8: Reduce short-term memory load [32] (slightly covered feature)
Autonomous	Useful, Credible	R8: Reduce short-term memory load [32] (slightly covered feature)
Efficient	Useful	R4: Design dialogs to yield closure [32] (slightly covered feature)
Innovative	Valuable, Desirable	No one
Customizable	Useful	R2: Seek universal usability [32] (slightly covered feature)
Flexible	Useful, Findable	R2: Seek universal usability [32] (slightly covered feature) R6: Permit easy reversal of actions [32] (partially covered feature)
Multiple results	Useful, Valuable	R3: Offer informative feedback [32] (slightly covered feature)
Selective	Useful, Valuable	R3: Offer informative feedback [32] (slightly covered feature) R4: Design dialogs to yield closure [32] (slightly covered feature)

The letter "R" is used as ID to indicate the number of Shneiderman's rules [32].

Appendix J: Second iteration, Step 5: "Selection stage". Rules selection process

See Table 17

Table 17
Second iteration, Step 5: "Selection stage". Rules selection process.

ID	Heuristic/Principle Name	Action	Refs.	Generative design application feature covered	Applicability
R1	Strive for consistency	Adapt	[32]	Adaptative	(1) Useful
R2	Seek universal usability	Adapt	[32]	Customizable, Flexible	(2) Important
R3	Offer informative feedback	Keep	[32]	Multiple results, Selective	(1) Useful
R4	Design dialogs to yield closure	Adapt	[32]	Efficient, Selective	(1) Useful
R5	Prevent errors	Adapt	[32]	Efficient	(3) Critical
R6	Permit easy reversal of actions	Adapt	[32]	Adaptative	(2) Important
R7	Keep users in control	Keep	[32]	Adaptative	(3) Critical
R8	Reduce short-term memory load	Adapt	[32]	Optimizable, Autonomous	(2) Important

Appendix K: Full UXH-GEDAPP specification, using the template proposed in the methodology applied [17,18]

See Table 18,Table 19,Table 20,Table 21,Table 22,Table 23,Table 24,Table 25,Table 26

Table 18
UXH-GEDAPP-1: Adaptability of information.

ID	UXH-GEDAPP-1
Name	Adaptability of information
Definition	The system should allow the input and modification of any parameter set by the user, according to the context of the software and adapting to the needs and user requirements.
Explanation	The system must adapt to the user's requirements, performing correctly the actions requested. The user can enter and modify different parameters according to their objectives. In addition, the system should provide different forms of data representation based on user preferences, according to the context of the generative design application.
Priority	(3) Critical
Generative design feature	Adaptative
UX attribute related	Useful, Usable
Problems	The evaluator may failure to understand the purpose and context of the heuristic.
Benefits	The user will be able to obtain results according to his/her requirements. In addition, the user will be able to add and modify the entered parameters to obtain a new result.
Checklist	1. The system allows to modify the parameter already entered. 2. The system allows to enter the desired parameter. 3. Actions not selected by the user are not automatically initiated. 4. The system provides different data representation. 5. The system allows the reversal of actions.
Examples	Example of compliance: Fig. 2 shows a panel in Revit 2021 software where the user can create a new study on generative design, where the system allows to incorporate the parameters that the user wants.
Sets of heuristics or elements related	N3/P9/R7: User control and freedom [29] / User control [30] / Keep users in control [32] P8: Explicit user actions [30] C9: Provide multiple coding of data [31] R1: Strive for consistency [32] R6: Permit easy reversal of actions [32]

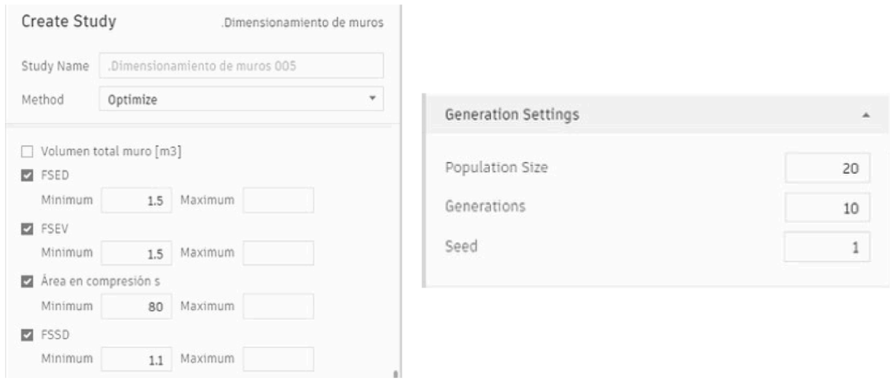

Fig. 2. Example of compliance UXH-GEDAPP-1: Adaptability of information.

Table 19
UXH-GEDAPP-2: Optimization and development autonomy.

ID	UXH-GEDAPP-2
Name	Optimization and development autonomy
Definition	The system should be autonomous in operation, optimizing and automating calculation and design processes, reducing the user's cognitive load during the development process.
Explanation	The system should automate and optimize design and data calculation processes, minimizing the number of actions required to obtain optimal results, improving system action times, data analysis and reducing the amount of information flow.
Priority	(3) Critical
Generative design feature	Optimizable, Autonomous
UX attribute related	Valuable, Useful, Credible
Problems	The evaluator may not understand how to use the heuristic or identify what problem to associate with this heuristic.
Benefits	The user can obtain the expected results quickly and in an automated way.
Checklist	1. The user does not need to perform manual calculations. 2. The user does not have to remember previously completed fields. 3. The system recommends default values. 4. The system performs mathematical calculations automatically. 5. The user is not required to perform additional actions.
Examples	Example of compliance: Fig. 3 shows a panel in Revit 2021 software where the user can create a new study on generative design. The system allows to use as a method "optimize", causing the system to perform all calculations relevant to the context of the software, generating different results.
Sets of heuristics or elements related	N7: Flexibility and efficiency of use [29] P7: Minimal actions [30] C1/C7: Automate unwanted workload/ Limit data driven tasks [31] N6/P6: Minimize memory burden [29] / Cognitive workload [30] R8: Reduce short-term memory load [32]

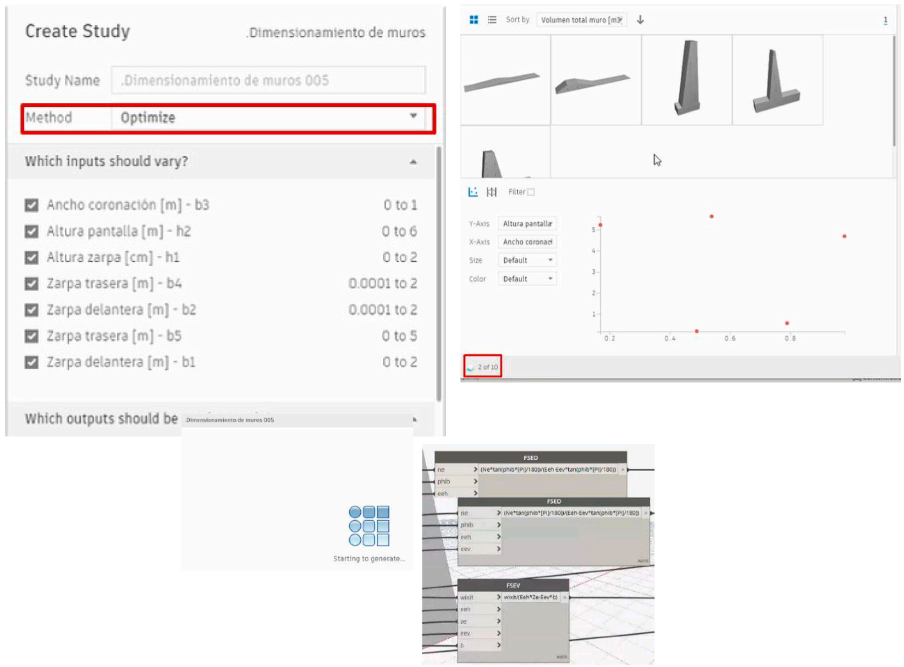

Fig. 3. Example of compliance UXH-GEDAPP-2: Optimization and development autonomy.

Table 20
UXH-GEDAPP-3: System efficiency.

ID	UXH-GEDAPP-3
Name	System efficiency
Definition	The system should perform the functionalities requested by the user in an effective way. The functionalities should be visible, traceable, intuitive and with reasonable response times.
Explanation	The system should allow the user to use the functionalities in an effective way and should also generate results by performing the stages corresponding to the generative design, which are: generation, analysis, ranking and evolution. The system should correctly perform the calculations corresponding to the development of a structure and with reasonable response times. In addition, the system should display the information in a clear and concise manner; and the results should be consistent with the parameters provided by the user.
Priority	(2) Important
Generative design feature	Efficient
UX attribute related	Usable
Problems	The evaluator may confuse this heuristic with heuristic UXH-GEDAPP-2.
Benefits	The user will be able to obtain the expected results in a fast and effective way.
Checklist	1. Essential functionalities are available to the user. 2. The response time of the system is acceptable to the user. 3. The information provided by the system is clear and understandable to the user. 4. The results provided by the system are consistent and free of errors. 5. The functionalities are intuitive for the users.
Examples	Example of compliance: Fig. 4 shows the Revit 2021 software. Based on the parameters set by the user, the system provides different design solutions where the details of each of these can be visualized.
Sets of heuristics or elements related	N7/C2: Flexibility and efficiency of use [29] / Reducing uncertainty [31] N2/P11: Match between the system and the real world [29] / Compatibility with the user [30] R4: Design dialogs to yield closure [32]

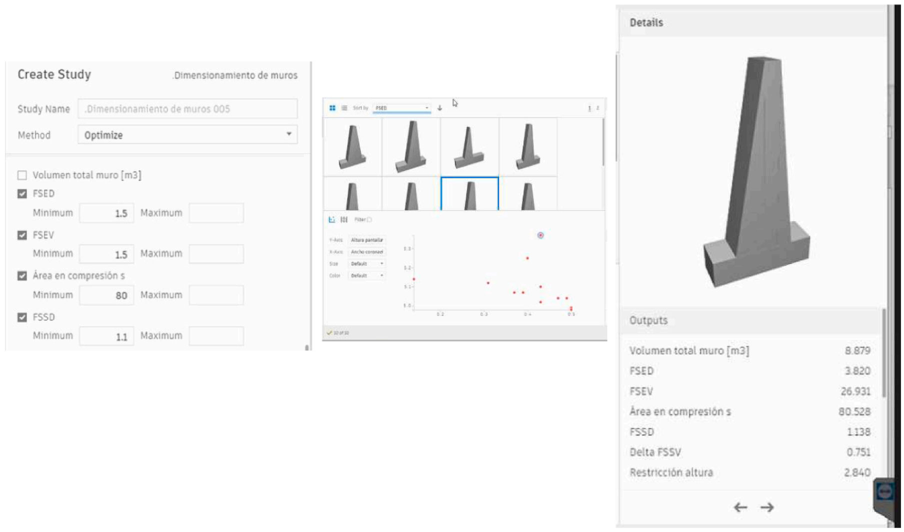

Fig. 4. Example of compliance UXH-GEDAPP-3: System efficiency.

Table 21
UXH-GEDAPP-4: Aesthetic and innovative design.

ID	UXH-GEDAPP-4
Name	Aesthetic and innovative design
Definition	The system should have an intuitive and user-friendly interface. The system outputs should be novel, creative, and understandable to the user.
Explanation	The system should be navigable with an adequate structure of the information without overload of irrelevant data for the user, with an intuitive and pleasant interface making clear the fields and options to be completed. In addition, the system should have representative icons and visible selection buttons. The results provided by the system should be credible, innovative, and creative, and should be understandable to the user.
Priority	(2) Important
Generative design feature	Innovative
UX attribute related	Valuable, Desirable
Problems	The evaluator may not understand the description of the results shown by the system, since they must have knowledge of generative design.
Benefits	The user can interact with the application without data overload, additionally the application provides novel and creative results for the user.
Checklist	1. The content provided does not visually clutter with excessive images, text, or other elements. 2. Colors and images do not hinder the visualization of the application. 3. Font sizes and application elements are appropriate. 4. The representation of the results is credible, novel, and creative.
Examples	Example of compliance: Fig. 5 shows the Revit 2021 software. The user can appreciate the different design solutions. The system correctly specifies the fields that can be completed by the user and demarcates the main buttons to perform its functions.
Sets of heuristics or elements related	N8: Aesthetic and minimalist design [29] C4: Present new information with meaningful aids to interpretation [31]

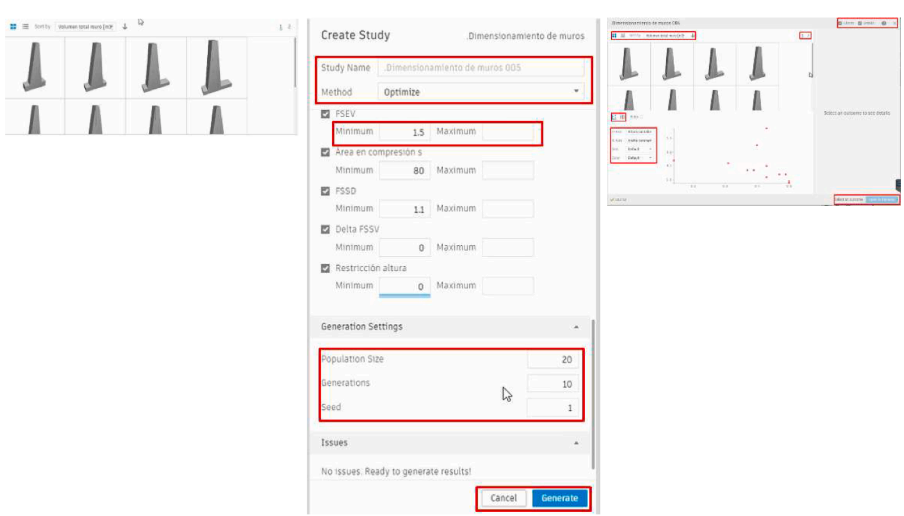

Fig. 5. Example of compliance UXH-GEDAPP-4: Aesthetic and innovative design.

Table 22
UXH-GEDAPP-5: Flexibility of use.

ID	UXH-GEDAPP-5
Name	Flexibility of use
Definition	The system should be used by different types of users. The system should be adapted to different user profiles, making its use simple and intuitive. In addition, the system should have different ways to perform a functionality, i.e., it should be configurable by the user.
Explanation	The system should allow both expert and inexperienced users to use the system functionalities easily and without major inconveniences. The system should have different ways to perform an action or functionality by the user, and it should also allow the user to configure and modify system properties.
Priority	(2) Important
Generative design feature	Customizable, Flexible
UX attribute related	Usable, Findable
Problems	The evaluator may confuse this heuristic with heuristic UXH-GEDAPP-1.
Benefits	The user will be able to navigate through the application without major inconveniences and will be able to reach his goals quickly.
Checklist	1. There are shortcuts for expert users. 2. The system allows configuring properties or elements. 3. The system allows filtering the data provided. 4. The system allows to modify the way data or information is represented. 5. The system can be used by different user profiles.
Examples	Example of compliance: Fig. 6 shows the Revit 2021 software. The panel for creating a generative design study can be used by any type of user. On the other hand, once the solutions are generated, the software generates varied forms of representation of results available to the user.
Sets of heuristics or elements related	N7/P10: Flexibility and efficiency of use [29] / Flexibility [30] C9: Provide multiple coding of data [31] R2: Seek universal usability [32]

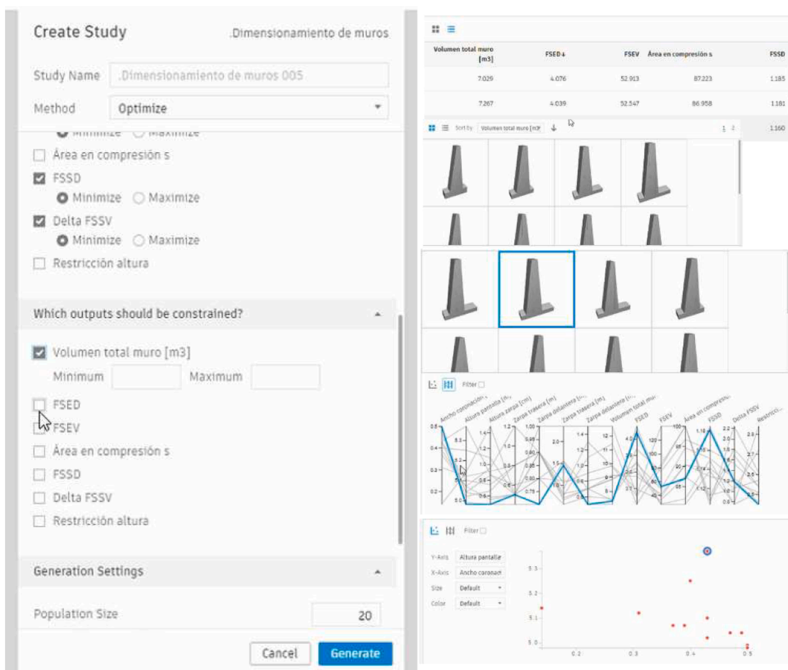

Fig. 6. Example of compliance UXH-GEDAPP-5: Flexibility of use.

Table 23
UXH-GEDAPP-6: Multiplicity of results.

ID	UXH-GEDAPP-6
Name	Multiplicity of results
Definition	The system should provide a set of solutions in a generative and iterative way, based on the parameters provided by the user.
Explanation	The set of results generated by the system should be clearly represented to the user and should be consistent with the parameters provided by the user.
Priority	(3) Critical
Generative design features	Multiple results
UX attribute related	Useful, Valuable
Problems	The evaluator may not understand the results if he/she is not familiar with the concept of generative design.
Benefits	The user can obtain a set of results based on their parameters.
Checklist	1. The system provides a set of solutions. 2. The system gives the results in a generative and iterative way. 3. The user can modify the number of iterations to be performed by the system.

(*continued on next page*)

Table 23 (*continued*)

ID	UXH-GEDAPP-6
Examples	Example of compliance: Fig. 7 shows a set of design solutions provided by Revit 2021 software.
Sets of heuristics or elements related	N1: Visibility of system status [29] P2: Feedback [30] P4: Grouping/distinction [30] C6: Group data in consistently, meaningful ways [31] R3: Offer informative feedback [32]

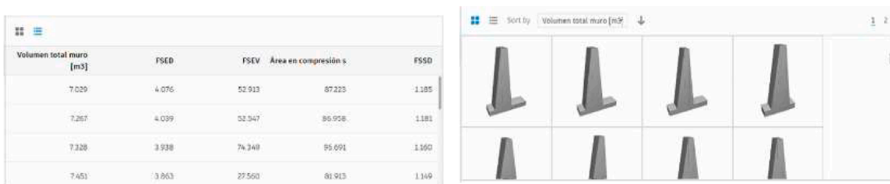

Fig. 7. Example of compliance UXH-GEDAPP-6: Multiplicity of results.

Table 24
UXH-GEDAPP-7: Results selection.

ID	UXH-GEDAPP-7
Name	Results selection
Definition	The results provided by the system are ranked based on the user's needs, best fitting the parameters provided by the user.
Explanation	The results generated by the system should be ranked in each iteration by the system, using as criteria the parameters given by the user, being the last iteration the most adapted and optimal result to the user's parameters.
Priority	(3) Critical
Generative design feature	Selective
UX attribute related	Useful, Valuable
Problems	The evaluator may not understand the results if he/she is not familiar with the concept of generative design.
Benefits	The user will be able to have at his disposal the best results adapted to the delivered parameters.
Checklist	1. The number of iterations performed by the system is visible to the user. 2. The user can rank the solutions based on his/her requirements. 3. It is possible to visualize the best result obtained by the system based on the user's request. 4. The system notifies the user of the best result obtained based on his/her requirements.
Examples	Example of compliance: Fig. 8 shows the iterations performed by Revit 2021 software, generating solutions in an iterative manner. In addition, the system presents a graph with the solutions generated, with the selected one being the most appropriate based on the parameters provided.
Sets of heuristics or elements related	N1: Visibility of system status [29] P2: Feedback [30] C6: Group data in consistently, meaningful ways [31] C8: Include in the displays only that information needed by the operator at a given time [31] R3: Offer informative feedback [32] R4: Design dialogs to yield closure [32]

Fig. 8. Example of compliance UXH-GEDAPP-7: Results selection.

Table 25
UXH-GEDAPP-8: Error management.

ID	UXH-GEDAPP-8
Name	Error management
Definition	The system should provide ways to prevent, diagnose, correct, and recover from errors.
Explanation	The system should prevent the occurrence of errors by displaying warning messages before the action that results in an error. If an error does occur, the system should display clear, informative, and unambiguous error messages, notifying the user of the problem that occurred and possible solutions to it, allowing the user to recover from these errors.
Priority	(3) Critical
Generative design feature	Efficient

(*continued on next page*)

Table 25 (*continued*)

ID	UXH-GEDAPP-8
UX attribute related	Usable, Credible
Problems	The evaluator may confuse this heuristic with heuristic UXH-GEDAPP-9.
Benefits	The user has no inconvenience in navigating the application.
Checklist	1. The user is warned before closing or deleting any element of the application. 2. The user is asked for authorization before performing an important action. 3. The user is provided with warning messages prior to the occurrence of an error. 4. Clear and informative messages are given to the user once the error has occurred.
Examples	Example of compliance: Fig. 9 shows a warning message to the user before making an action (in this case, open a graph).
Sets of heuristics or elements related	N9: Help the user to recognize, diagnose and recover from errors [29] P2: Feedback [30] P13: Error management [30] R5: Prevent errors [32]

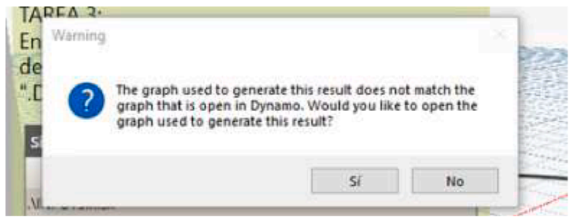

Fig. 9. Example of compliance UXH-GEDAPP-8: Error management.

Table 26
UXH-GEDAPP-9: Help and documentation.

ID	UXH-GEDAPP-9
Name	Help and documentation
Definition	The system should provide help and documentation of how the system works in a place that is easily accessible to users.
Explanation	The system should have a user's manual, which will help users to solve doubts and concerns they may have about the use of the software. The system should have a section with clear and specific documentation that allows users to perform their tasks without major inconveniences.
Priority	(3) Critical
Generative design feature	Flexible
UX attribute related	Credible
Problems	The evaluator may confuse this heuristic with heuristic UXH-GEDAPP-8.
Benefits	The user will be able to solve his doubts and concerns when required. In addition, the user will be able to access the documentation related to the application to be used.
Checklist	1. The user has access to a help section that allows to solve the doubts regarding the application. 2. The help section is visible and easy to find for the user. 3. The documentation provided is written in a clear and understandable language for the user.
Examples	Example of compliance: Fig. 10 shows a user manual for the system, in addition to a help button for the user.
Sets of heuristics or elements related	N10/P14: Help and documentation [29,30]

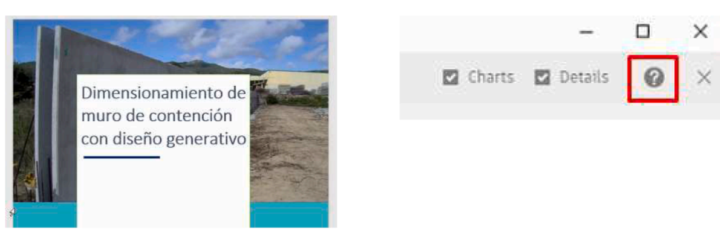

Fig. 10. Example of compliance UXH-GEDAPP-9: Help and documentation.

References

[1] A. Cavieres, Diseño paramétrico orientado a la performance, Rev. Arquit. 13 (16) (2007) 39–45, https://doi.org/10.5354/0719-5427.2007.28201. Jan.
[2] R. Lajas Benéitez, "Diseño, automatización y modelización de una nave prefabricada mediante metodología BIM", Jan. 2019, Accessed: Jan. 03, 2022. [Online]. Available: https://dehesa.unex.es:8443/handle/10662/10155.
[3] H. Abdallah, et al., Employing generative design for sustainable construction, Creat. Constr. Conf. (2019) 692–698, https://doi.org/10.3311/CCC2019-095, 2019.
[4] M. McKnight, Generative Design: what it is? How is it being used? Why it'sa game changer, KnE Eng. (2017) 176–181.
[5] R. Johan, M. Chernyavsky, A. Fabbri, N. Gardner, M. Haeusler, and Y. Zavoleas, "Building intelligence through generative design Structural analysis and optimisation informed by material performance", 2019.
[6] S. Tafraout, N. Bourahla, Y. Bourahla, A. Mebarki, Automatic structural design of RC wall-slab buildings using a genetic algorithm with application in BIM environment, Autom. Constr. 106 (2019) 102901.
[7] G. Díaz, R.F. Herrera, F. Muñoz-La Rivera, E. Atencio, Generative design for dimensioning of retaining walls, Mathematics 9 (16) (2021) 1918, https://doi.org/10.3390/MATH9161918, 2021Vol. 9, Page 1918volAug.

[8] F. Muñoz-La Rivera, J.C. Vielma, R.F. Herrera, J. Carvallo, Methodology for building information modeling (BIM) implementation in structural engineering companies (SECs), Adv. Civ. Eng. (2019), https://doi.org/10.1155/2019/8452461 vol2019.

[9] W. Ma, X. Wang, J. Wang, X. Xiang, J. Sun, Generative design in building information modelling (BIM): approaches and requirements, Sensors 21 (16) (2021) 5439.

[10] S. Abrishami, J.S. Goulding, F. Pour-Rahimian, A. Ganah, Integration of BIM and generative design to exploit AEC conceptual design innovation, J. Inf. Technol. Constr. 19 (2014) 350–359.

[11] G. Diaz, R.F. Herrera, F.C. Muñoz-La Rivera, E. Atencio, Applications of generative design in structural engineering, Rev. Ing. Constr. 36 (1) (2021) 29–47.

[12] R. Hartson, P.S. Pyla, The UX book : Agile UX Design For a Quality User Experience, 2nd ed., Morgan Kaufmann, 2018.

[13] Experience Research Society, "UX evaluation methods", 2023. https://experienceresearchsociety.org/ux/evaluation-methods/, (accessed Sep. 01, 2023).

[14] J. Nielsen, Usability inspection methods, in: Proceedings of the CHI '94: Conference Companion on Human Factors in Computing Systems 25, 1994, pp. 413–414, https://doi.org/10.1145/259963.260531.

[15] D. Quiñones, C. Rusu, How to develop usability heuristics: a systematic literature review, Comput. Stand. Interfaces 53 (2017), https://doi.org/10.1016/j.csi.2017.03.009.

[16] S. Hermawati, G. Lawson, Establishing usability heuristics for heuristics evaluation in a specific domain: is there a consensus? Appl. Ergon. 56 (2016) 34–51, https://doi.org/10.1016/j.apergo.2015.11.016.

[17] D. Quiñones, C. Rusu, V. Rusu, A methodology to develop usability/user experience heuristics, Comput. Stand. Interfaces 59 (2018) 109–129, https://doi.org/10.1016/j.csi.2018.03.002. November 2017.

[18] D. Quiñones, C. Rusu, Applying a methodology to develop user eXperience heuristics, Comput. Stand. Interfaces 66 (2019), https://doi.org/10.1016/j.csi.2019.04.004.

[19] R. Velasco Pérez, Estudio De La Aplicación Del Diseño Generativo Al Diseño Conceptual Arquitectónico, Universitat Politècnica de València, 2015.

[20] H. Bohnacker, B. Gross, J. Laub, C. Lazzeroni, Generative Design: Visualize, Program, and Create with Processing, Princeton Architectural Press, 2012 [Online]. Available, https://books.google.cl/books?id=tSS9uAAACAAJ.

[21] D. Henríquez, R.F. Herrera, J.C. Vielma, Method for designing prequalified connections using generative design, Buildings 12 (10) (2022) 1579.

[22] E. Souza, "¿Cómo impacta el Diseño Generativo en la arquitectura? | ArchDaily en Español", 2020. https://www.archdaily.cl/cl/937716/como-impacta-el-diseno-generativo-en-la-arquitectura (accessed Aug. 31, 2023).

[23] Autodesk, "What is Generative Design | Tools Software | Autodesk", 2023. https://www.autodesk.com/solutions/generative-design (accessed Aug. 31, 2023).

[24] ISO, ISO 9241-210 : 2010 Ergonomics of human-system interaction — Part 210: human-centred design for interactive systems, Int. Stand. (2019). https://www.iso.org/standard/77520.html.

[25] P. Morville, "Facets of the user experience - user experience design", 2004. https://semanticstudios.com/user_experience_design/, (accessed Feb. 13, 2022).

[26] ISO 9241-11, "ISO - ISO 9241-11:2018 - Ergonomics of human-system interaction — Part 11: usability: definitions and concepts". https://www.iso.org/standard/63500.html (accessed Jun. 01, 2022).

[27] R. Gatica, M. Ortuño, and P. Velásquez, "Diseño Generativo", 2010. Accessed: Aug. 31, 2023. [Online]. Available: https://www.u-cursos.cl/fau/2010/0/DIT-205/1/material_docente/bajar?id_material=456459.

[28] J. Nielsen, "How to Conduct a Heuristic Evaluation", 2023. https://www.nngroup.com/articles/how-to-conduct-a-heuristic-evaluation/, (accessed Aug. 31, 2023).

[29] J. Nielsen, "10 Usability Heuristics for User Interface Design", 2020. https://www.nngroup.com/articles/ten-usability-heuristics/, (accessed Jun. 21, 2022).

[30] C. Pribeanu, A revised set of usability heuristics for the evaluation of interactive systems, Inform. Econ. 21 (3) (2017) 31.

[31] J. Gerhardt-Powals, Cognitive engineering principles for enhancing human-computer performance, Int. J. Hum. Comput. Interact 8 (2) (1996) 189–211.

[32] B. Shneiderman, C. Plaisant, M. Cohen, S. Jacobs, N. Elmqvist, N. Diakopoulos, Designing the User interface: Strategies For Effective Human-Computer Interaction, Pearson, 2016.

[33] 99designs Team, "The 7 principles of UX design—And how to use them - 99designs", 2021. https://99designs.com/blog/web-digital/ux-design-principles/, (accessed Nov. 17, 2023).

[34] G. Díaz, Dimensionamiento De Muros De Contención Con Diseño Generativo, Pontificia Universidad Católica de Valparaíso, 2020.

[35] D. Quiñones, C. Rusu, D. Arancibia, S. González, M.J. Saavedra, SNUXH: a set of social network user experience heuristics, Appl. Sci. 10 (18) (2020), https://doi.org/10.3390/APP10186547.

[36] C. Lewis, Using the" thinking-aloud" Method in Cognitive Interface Design, IBM TJ Watson Research Center Yorktown Heights, NY, 1982.

[37] L. Cayola, J.A. Macías, Systematic guidance on usability methods in user-centered software development, Inf. Softw. Technol. 97 (2018) 163–175.

[38] J. Nielsen, "How Many Test Users in a Usability Study?", 2012. https://www.nngroup.com/articles/how-many-test-users/, (accessed Nov. 17, 2023).

[39] J. Nielsen, "Why you only need to test with 5 users", 2000. https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/, (accessed Jan. 12, 2024).

[40] J. Nielsen, "Usability 101: introduction to Usability", Jan. 03, 2012. https://www.nngroup.com/articles/usability-101-introduction-to-usability/, (accessed Feb. 13, 2022).