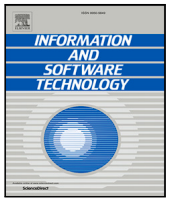
Information and Software Technology 153 (2023) 107081

Contents lists available at ScienceDirect

Information and Software Technology

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



A systematic literature review of the Design Critique method

Lorans Alabood *, Zahra Aminolroaya, Dianna Yim, Omar Addam, Frank Maurer

SEER Lab, Department of Computer Science, University Dr NW University of Calgary, ICT Building, 2500 University Dr NW, Calgary, T2N1N4, Alberta, Canada

ARTICLE INFO

Keywords:
Design Critique
Human-centered design
User-experience research
AgileUX
Systematic literature review

ABSTRACT

Context: The Design Critique (DC) method is becoming more common in Human–Computer Interaction (HCI) and User Experience (UX) studies as the need for new evaluation methods of emerging technologies is increasing. However, there is an clear lack of guidelines on how to conduct DC studies in the UX context.
Objective: The goal of this paper is to provide an overview of the DC method in the fields of UX. In addition, this paper aims to propose a generic process of running DC studies in the same context.
Methods: We present a systematic literature review of the DC method. Moreover, we conduct a course of thematic analysis on the selected papers to identify the various DC processes and explore the following attributes: participant categories, data collection methods, and data analysis methods in each process.
Results: We identified three different trends of DC processes: detailed, moderate and minimal. In addition, we proposed a generic DC process consisting of 10 steps divided into three main phases: preparation, conducting design critique, and pro-processing. We found that domain experts represent the majority of studies participants. Using interviews to collect qualitative data and using script coding analysis are the two most common methods of collecting and analyzing data.
Conclusion: Conducting DC studies can improve overall systems usability by addressing design flaws at an early stage of development. The process of conducting a DC varies, depending on the project goals and states. The DC method aligns well with the small light-weight steps approach in Agile methods.

1. Introduction

Collecting feedback is an integral part of any design process, especially in domains where creativity is essential such as engineering, education, and architecture [1–3]. The Design Critique (DC) is one of the common methods of collecting feedback from stakeholders and experts. One way to define the DC method is the approach of "constructive criticism" in order to collect feedback on a design at any stage of the project lifespan to determine whether the design meets its objectives or not [4,5]. Typically, collecting feedback includes participants from two major categories; end-users and experts, allowing designers to view design matters from different perspectives. Furthermore, this approach is believed to reduce costs and efforts by eliminating design issues as much as possible before moving to the testing and implementation phases [4].

The User-Centered Design (UCD) method is widely applied in the field of HCI and UX. A typical UCD process follows these steps: investigating, ideating, prototyping, testing and finally implementing. Stakeholders' input is collected during the investigation phase only, such as conducting requirements elicitation study, user surveys, and needs assessment [6,7]. It is fair to say there is a gap in stakeholders' involvement between the 'investigation' and the 'testing' phases. Having that the DC allows participants to provide feedback at any stage of the design process, integrating the DC method into UCD processes can improve systems' design by ensuring that prototyping and development are not completed blindly [8].

Additionally, this integration contributes to introducing new design knowledge and design practices that can reduce time, effort and cost spent during the prototyping and testing phases [4]. The promise is that the chances of finding major design issues during the testing phase are reduced by identifying and addressing design issues prior to testing. This is because UX designers would optimize the system design while working closely with end-users and experts throughout the UCD process [5,9].

Recent research investigated incorporating the UCD into Agile processes (see [10] for an overview). Agile methods aim to deliver useful and usable computer systems effectively and efficiently by conducting small, lightweight iterations [11]. Agile methods aim to iteratively and incrementally hand over completed system features in a timely manner [11]. Incorporating Agile methods with the UCD is known as the Agile User-Centered Design (AUCD) or AgileUX (AUX). This

* Corresponding author.
E-mail addresses: lorans.alabood@ucalgary.ca (L. Alabood), zahra.aminolroaya@ucalgary.ca (Z. Aminolroaya), dyim@ucalgary.ca (D. Yim), okaddam@ucalgary.ca (O. Addam), fmaurer@ucalgary.ca (F. Maurer).

https://doi.org/10.1016/j.infsof.2022.107081
Received 10 December 2021; Received in revised form 16 September 2022; Accepted 19 September 2022
Available online 26 September 2022

integration brings developers and UX designers to work closer together while receiving feedback from users and stakeholders during the UCD process [11–13]. Common techniques for integrating UX into Agile processes include short design iterations, designers working one iteration ahead of the development team and utilizing design techniques with limited effort spent on upfront research and design [13–16]. Given that agile teams work in short iterations of one to four weeks and expect input from UX in the same time frames, UX needs to be lightweight and timely. As illustrated below, a DC process based on early-stage low-fidelity prototypes can be utilized by an AUX on these time scales: it can work iteratively on parts of the user interface that will be developed in the upcoming iteration. Therefore, the UX-Design Critiques integration can be described as approaches to improving a software system's usability that fits well in Agile processes.

The HCI community has highlighted the importance and benefits of DC for HCI [17,18]. The COVID-19 pandemic shed more light on the significance of remote design evaluation methods. For instance, researchers in [19], and [20] described tailored DC processes for evaluating immersive systems remotely due to the pandemic. Despite its benefits, the DC method is not well-investigated in the HCI literature [4]. In this paper, we conduct a Systematic Literature Review (SLR) of using the DC method in HCI and UX, and we attempt to address the following questions:

1. How is the DC method defined in the literature?
2. What are the different contexts of using the DC method?
3. What are the different processes of conducting DC studies in the literature?
4. What are the participant categories in these studies?
5. What are the common data collection and analysis methods in these studies?
6. Based on previous DC studies, how would a generic DC process for UX and HCI studies look like?

In the next section of the paper, we describe our research methodology in detail, followed by the data analysis method. In the following section, the results of our analysis are presented. After that, we thoroughly discuss specific attributes of our study analysis. We propose a generic DC process based on the different DC processes extracted from literature studies. In the last two sections, we discuss the limitations of our study and provide a conclusion.

2. Methodology

Our research methodology is entirely driven by the guidelines for performing SLRs by Kitchenham and Charters [21]. Our first step was developing a search strategy for relevant work. After that, we introduced a set of inclusion and exclusion criteria to filter our search results in three stages; title, abstract and full script screening. In this section of the paper, we describe each of these steps in detail.

2.1. Search strategy

Our preliminary searches aimed to identify existing SLR papers and assess the volume of potentially relevant studies. However, our search for other SLR papers did not yield any results. Therefore, we moved on to conducting trial searches using a different combination of search terms extracted from papers that ran DC studies. Consequently, our search terms were developed and refined iteratively in a snowballing fashion where the search strings were modified every time we identified a new keyword. Below, we include the search terms used in our final queries. The terms were looked up in titles, abstracts, keywords, and full text. See Table 1 for our search strings.

The terms "design critique" and "design critics" were the main terms used to create search strings. Additional terms were used to focus our search on computer science and software engineering related research work such as: "computer science", "HCI", "human–computer interaction", "human–computer interaction", "agile", "UX", "human-centered design", "UCD", "task-centered design", "act-
ivity-centered design", and "participatory design".

2.2. Search process

We used two search engines: we mainly used Google Scholar linked to our university's library resources. The university library has subscriptions to almost all computer science and software engineering databases, including but not limited to: IEEE Xplore, ACM Digital Library, Scopus, Web of Science, CiteSeerX and more. After consulting with a librarian, we were advised that all these databases would be searched via Google Scholar when linked to thee university resources. We used the Publish or Perish software to run all Google Scholar queries. A significant benefit of this approach is to – automatically – record all search results in a spreadsheet. Due to the lack of published work in this area, we did not apply any search filters to maximize our findings. We combined all search results in one master spreadsheet consisting of study titles, year of publication, and authors.

Our librarian advised that using Google Scholar and university library only, there would be a chance to miss some results due to search engine functional limitations. Therefore, we picked the top three databases in our field and ran another round of searches on each database separately. We picked ACM Digital Library, IEEE Xplore, and Scopus as the most related databases for this field. Our search for new relevant papers on these databases revealed a small number of publications that did not appear via Google Scholar and our university library. We accumulated many duplicate results after we running all searches. We used Excel's 'delete duplicates' feature to keep unique entries only. The final spreadsheet was shared with other authors for blind screening, which we describe in the next section.

2.3. Screening process

We developed three screening stages using a set of inclusion and exclusion criteria, see Table 2. First, we performed screening on the studies' title, abstract, and full text. We skipped the introduction and conclusion stages as we thoroughly screened papers that passed the abstract screening. Papers selection was based on the authors' interpretation and guided by the inclusion and exclusion criteria. To improve the validity of the screening process, at least two authors completed each stage. A total of three authors completed all screenings, while a fourth author supervised the screening process. After every screening stage, authors exchanged a spreadsheet of the selected papers for cross-checking. In the case of a conflict or a disagreement, the fourth author was consulted. In the final screening stage, we used the same inclusion and exclusion criteria to screen unique new findings from the three databases we described in the previous section, and we included these new findings in the analysis.

2.4. Screening results

Four runs of online searches were performed on Google Scholar, and our university library search tool via Publish or Perish software resulting in a total of 1455 papers. After deleting duplicates and non-English language papers, the final number of papers was reduced to 704. After the title screening was performed, 253 papers were selected for abstract screening. By applying the abstract screening inclusion and exclusion criteria, 104 papers were selected for the final full paper screening process. Out of the 104 papers, only 39 papers were selected after applying the full paper screening criteria. After running separate searches on ACM DL, IEEE Xplore and Elsevier Scopus, we retrieved

Table 1
Used keywords and search string.

“design critique” AND (“computer science” OR “HCI” OR “human–computer interaction” OR “human–computer interaction” OR “UX”).
“design crits” AND (“computer science” OR “HCI” OR “human–computer interaction” OR “human–computer interaction” OR “UX”)
“design critique“ AND (”computer science“ OR ”software engineering“ OR ”HCI“) AND (”interaction design“ OR ”human-centered design“ OR ”human-centered design“ OR ”UCD“ OR ”task-centered design“ OR ”activity-centered design“ OR ”participatory design")
“design crits” AND (“computer science” OR “software engineering” OR “HCI”) AND (“interaction design” OR “human-centered design” OR “human-centered design” OR “UCD” OR “task-centered design” OR “activity-centered design” OR “participatory design”)
“agile” (“design critique” OR design crits”)

Table 2
The inclusion and exclusion criteria for each screening process.

Screening process	Inclusion criteria	Exclusion criteria
Title screening	Titles that included “design critique” or “design crits”. Or titles indicated to using or describing a DC process in the context of computer science and software engineering. Titles that indicated to discussing DC definitions.	All non-English language papers. Titles that did not indict to using DC to evaluate designs. Titles did not’ indicate to defining or describing DC processes.
Abstract screening	Studies that discussed the DC method in the context of UX research or computer science education, studies that mentioned conducting evaluations using DC on system prototypes. Studies that indicate to providing definitions of DC in the context of HCI or UX	Any theoretical study that did not perform any evaluation using design critique, studies that are not related to computer science, systematic literature review paper that did not discuss conducting design critique evaluation on prototypes were excluded.
Full paper screening	Papers were included if they properly described a DC process, the DC was applied practically in the context of a computer science related topics such as a user interface or user experience design, papers that discussed the definition or the importance of the DC method in the context of HCI or UX.	Studies that conducted DC but different fields such as biomedical engineering or industrial design as these fields are out of the scope of this paper. Studies that mentioned conducting the DC process but did not provide any information beyond that. Studies that did not discuss DC in computer science education setup. Studies that did not provide definitions of the DC in the scope of this paper.

190 hits. By applying the same title screening criteria, we identified 17 potentially relevant papers that did not appear in the previous searches. A total of 14 of these papers were selected for full script screening. Only 7 of these papers passed the full script screening making the final total number of papers included in our study n=48. See Fig. 1 for the screening process.

3. Analysis

Three major tasks were the focus of our analysis (a) extracting the DC steps in every study, (b) categorizing studies based on their contexts and (c) identifying the following three attributes: participant category, data collection method, and data analysis method. Our analysis is driven by Cruzes, and Dyba’s recommendations for thematic analysis in software engineering research [22]. This combination of systematic literature review and thematic analysis has recently gained popularity in the field of HCI and software engineering, such as [23–25]. We downloaded all selected papers that described conducting DC and imported them into NVivo[1] software. Each paper was imported as a ‘case’ to NVivo. The next step was to read each script carefully for the purpose of conducting inductive coding. For instance, when coding [17,26], the code ’define the purpose’ and ’data analysis’ were identified based on the below quoted part of the scripts.

"We asked four SNL domain experts (two who participated in the focus group and two who did not) to use AstroTouch to transform a spacecraft’s initial orbit to align it with that of the ISS. Following this in-person task evaluation, we conducted a design critique to gather initial qualitative feedback about the feasibility of the application. We present the results of this critique in terms of the design guidelines outlined above", [26]

"The first phase of analyzing the video data was a free-form text annotation that identified points of interest and aimed to capture overall first impressions. We subsequently conducted an in-depth analysis of interactions occurring while the annotator tool was switched on, coding for all instances of the child expressing either critical feedback or contributions to the design" [17]

Following a reflexive approach, we refined our codes multiple times and merged redundant ones. Then, we grouped related codes into three main ’themes.’ Each of these themes was eventually translated into a DC phase where the ‘codes’ were translated into steps. The entire thematic framework is presented in Table 3. After completing the thematic analysis, we assigned the above attributes to each case. Then, we ran multiple ’Matrix Coding’ queries to investigate the relationships between different DC processes and these attributes. The results of these queries are discussed in Section 5.

4. Results

4.1. DC definitions

As stated earlier, the DC method is relativity new in the HCI and UX literature. Thus, researchers have different definitions of this method. To improve the understanding of the DC for HCI researchers, we present various definitions that we found in the literature in this section of

[1] https://www.qsrinternational.com/nvivo-qualitative-data-analysis-software/home

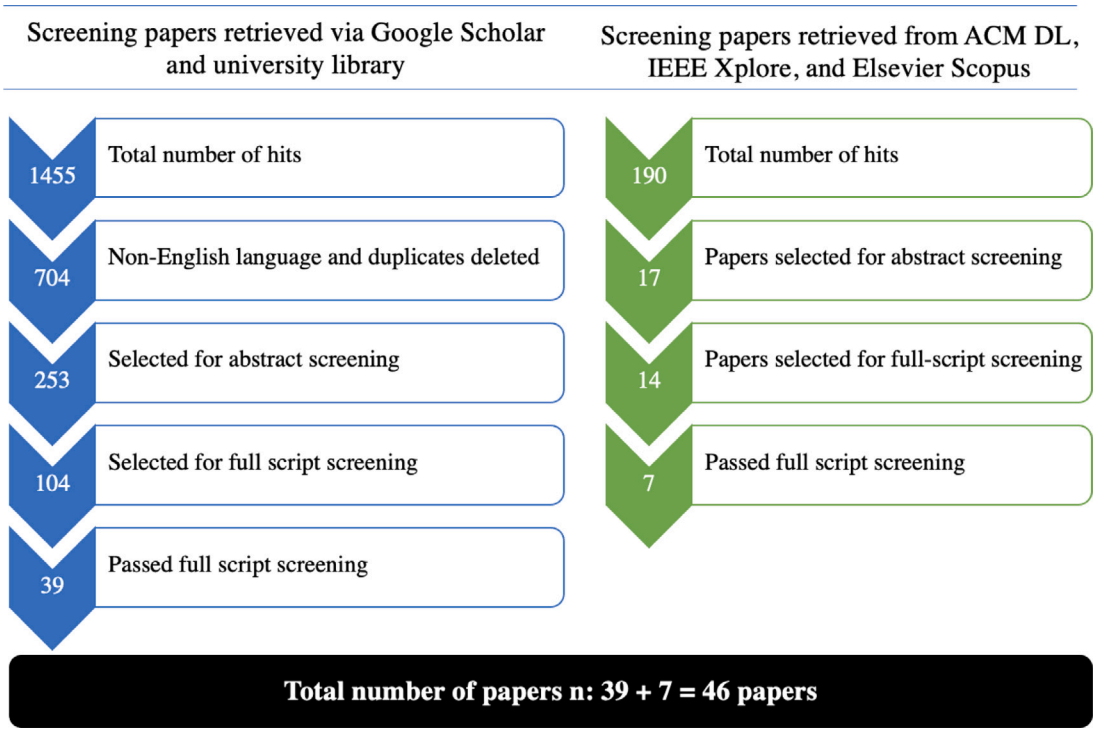


Fig. 1. Selection and screening process.

Table 3
The thematic framework with the code references.

Themes	Codes	Paper count
Preparation	Define the purpose	23
	Participant's selection	23
	Setting agenda	14
	Deciding on feedback structure	11
Conducting DC sessions	Present design	32
	Receiving feedback	27
Post Processing	Data analysis	26
	Reflections	20
	Response validation	9
	Implications	24

the paper. For instance, the DC was described as a useful tool that supports innovative prototyping by aiding collaborative design environments [27]. Hallnas explained the purpose of DC as providing an answer to the question of "why" - the basic rationale that motivates the system [28]. Blevis et al. [4] state that "design critiques accommodates and provides a mechanism for the consideration of ethics, aesthetics, and reasoning with respect to understanding interaction design", and it is also faster than traditional usability studies [4,29]. It can also help receive input on current design ideas, creating a collaborative process that can spur new thoughts by finding solutions to design flaws within current concepts. Herring et al. state [30] that "Design critiques allow a structured means for determining whether a design adheres to good practice and confirming that design judgement was carefully rendered throughout all aspects of the design". Design critiques can help build the criteria needed to perform heuristic evaluations, especially because custom guidelines are important when innovative projects are being built [31].

According to Xu et al. [32] reaching an effective solution using a design critique approach requires conducting multiple DC iterations with different viewpoints. This is also supported by Dow et al. who state that producing and sharing multiple alternatives could result in exploring more diverse ideas and creating higher quality work [33]. Another perspective is that conducting DC iteratively can promote sharing multiple perspectives over different stages of the project [2].

4.2. DC processes

As was mentioned earlier, there is an absence of a clearly defined DC process for HCI and UX researchers. The majority of studies that conducted in DC, did not layout their processes clearly. Therefore, in this section of our study, we present the various DC processes identified through our thematic analysis. We extracted and summarized the steps in each DC process to identify commonalities and discuss differences. Our analysis yielded three main trends of design critique processes: minimal processes (4–5 steps), moderate processes (6–7 steps) and detailed processes (8–10 steps).

4.2.1. Detailed processes

We identified nine papers where the DC process was very detailed and took place over 8–9 steps, refer to Table 4 for details. For instance, the authors of [3,34,35] recruited students to participate in three roles: potential end-user, experts, and researchers. At an early stage, students elicited design requirements by conducting interviews and questionnaires with other student groups to help establish functionality, usability, and user experience design elements. Using this data, students introduced low-fidelity prototypes that addressed five system tasks. The design was reviewed by other student participant groups, where all sessions were recorded and processed as part of the post-critique analysis, qualitative data analysis, then making design decisions. Some differences between the DC processes in these three papers were noted, such as missing the response validation or the reflection steps. A similar approach is presented in [19] to evaluate mixed reality applications for seniors living with dementias. The process starts with defining the purpose and deciding on the feedback structure where online interviews are the main data collection method. Then, five categories of participants, including potential end-users, domain experts (assistive technology experts) and developers, are interviewed to collect feedback. The post-processing steps include reflections, qualitative data analysis, response validation and implications. Researchers in [36]

Table 4
Papers with detailed DC processes.

DC phase	DC step	[19]	[37]	[17]	[12]	[3]	[34]	[35]	[38]	[39]	[32]	[36]
Preparation	Define the purpose	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Participant's selection	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
	Setting agenda			✓	✓	✓		✓	✓	✓	✓	✓
	Feedback structure	✓		✓	✓	✓	✓		✓		✓	✓
Conducting DC sessions	Present design	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Receiving feedback	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Post processing	Data analysis	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Reflections	✓	✓	✓			✓	✓		✓	✓	✓
	Response validation	✓	✓		✓	✓		✓			✓	
	Implications	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

evaluated a VR application with potential-end users, domain experts and developers following all 10 DC steps except for the response validation.

Researchers in [12] collected feedback from children on the autism spectrum to evaluate the software for learning. The study introduced a very detailed process that implemented all the DC steps except for the response validation. All DC sessions were video recorded while the children interacted with the software. Participants answered simple questions such as: "why did you like that?" and "why didn't you like that?". Script analysis was conducted to look for common themes and patterns.

In some cases, such as [38,39], the design critique processes were aimed at collecting feedback from online crowds using web forms. Both papers implemented a similar approach with minor differences: the reflections or deciding on feedback structure steps were not performed. Interestingly, the very goal of [38] was to create a software called CrowdCrit for conducting online DC studies. The concept of crowed critique was discussed again in [32], where a comprehensive DC process following all ten steps was implemented in a classroom setup. Using a computer-mediated critique, Easterday et al. [37] introduced the following process for design critiques: (a) presenting, (b) prompts, (c) written comments, (d) threaded commenting, and (e) anonymous up-and-down voting. The step that was added to this particular design process is that people were allowed to up-and-down vote comments previously made, adding and subtracting value as a group.

4.2.2. Moderate processes

In the second set of DC processes, a moderate number of steps were common, where the DC process consisted of 6–7 steps. For instance, Martin et al. [40] conducted a course of DC sessions with older adults as critiques. Their process starts with a pre-survey to determine participant eligibility. Then, participants were divided into groups, and an interface was demoed to each group. Next, participants reviewed the design with the research team and provided their feedback. A second round was carried out to identify current applications and what the users were accustomed to. Thereafter, researchers ran a focus group to discuss participants' feedback synchronously and record feedback on sticky notes. These notes were later used in an Affinity Diagramming session to identify common patterns and extract themes.

As seen in Table 5, [41], and [42] share similar moderate processes with more emphasis on the second and third DC phases. Furthermore, the response validation step was present in these two papers, while it was missing in almost all other papers. This could indicate that participants provided different input, which required further validation before making final design decisions. On the other hand, [6,43] shows a complete absence of the response validation step. It is worth mentioning that these four papers did not complete the setting agenda step in the preparation phase. Whereas this step (setting agenda) was undertaken in [40,44–46], yet we notice different variations of post-processing steps. Therefore, it is fair to say that the setting agenda step did not affect the post-processing phase in these moderate DC processes.

During our analysis, we came across three papers that conducted moderate DC processes and were published by the same research lab. In [10], researchers recruited domain experts (reservoir engineers)to participate in hands-on sessions that were video recorded and completed feedback surveys. Although we could assume that these video recordings and surveys were collected for post-processing, the paper did not mention the data analysis method. However, in [47], further DC on the same project (interactive reservoir engine system) was carried out where the Script Analysis method was used to analyze DC session data. Finally, the same research lab published another paper describing a collision reconstruction system called Re-Collision. The DC process in this study was conducted with police officers in a focus group. These three paper presented similar DC processes except in [26,44], the reflections and the response validation steps were not conducted.

In [48], the DC was conducted during the Spring semester with 13 computer science students to learn more about UI design methods. The study took an exploratory approach; as such, course instructors did not define the exact problems that needed to be addressed in the DC process. Therefore, all DC sessions took place in design studios where undergraduate students reviewed UI design ideas with each other with the help of graduate students. Notably, in this paper and in [45], the structure of the critique feedback was set prior to conducting DC sessions. As can be seen in Table 5, the setting agenda, deciding on feedback structure, and response validation steps were the least performed steps in the moderate DC processes. The most common steps were defining the purpose, participants selection, and implications.

4.2.3. Minimal processes

In the minimal processes category, we found that DC would implement only 4–5 steps. For instance, in [49,50], the DC is limited to preparing materials to collect rationale critique about a system design, followed by reflecting on feedback and ending with implications. Dunlap in [51] also follows a similar process: they started by reviewing the design with 9 participants, then discussing the results, and concluding the study by reporting findings. Researchers in [52,53] described a DC process that consists of defining the purpose, running the sessions, analyzing data, and reporting reflections. In [33,54], the same steps sequence was applied where both studies started with selecting participant categories, running the DC sessions, then data analysis and reflections. A similar minimal approach is presented in [30], but the receiving feedback part was done in the form of co-creating sessions rather than only reviewing the design. Whereas in [55], receiving feedback took place in an online form where participants entered their feedback in the comments section. Researchers used an API to retrieve feedback and build a database file.

Table 6 shows all papers with minimal processes along with the steps taken. Notably, in minimal DC processes, the following steps were not conducted in any paper: the setting agenda, deciding on feedback structure and response validation. The data analysis step was carried out in almost all papers except in [49,51], whereas the reflections step was present. This indicates that in minimal processes, some researchers made design decisions based on their reflections on the DC data without conducting formal data analysis.

Table 5
Papers with moderate DC processes.

DC phase	DC step	[41]	[42]	[6]	[43]	[40]	[10]	[47]	[44]	[48]	[45]	[46]	[26]
Preparation	Conducting DC sessions	✓		✓	✓		✓	✓	✓		✓	✓	✓
	Participant's selection		✓	✓	✓	✓	✓	✓	✓	✓			✓
	Setting agenda					✓			✓		✓	✓	✓
	Feedback structure		✓							✓	✓		
Conducting DC sessions	Present design	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Receiving feedback	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Post processing	Data analysis	✓	✓	✓	✓			✓	✓	✓	✓		✓
	Reflections			✓		✓	✓			✓	✓	✓	
	Response validation	✓	✓			✓							
	Implications	✓	✓	✓	✓		✓	✓	✓	✓		✓	✓

Table 6
Papers with minimal DC processes.

DC phase	DC step	[52]	[54]	[49]	[33]	[51]	[30]	[55]	[53]	[50]
Preparation	Define the purpose	✓		✓		✓	✓		✓	
	Participant's selection		✓		✓	✓	✓	✓		✓
	Setting agenda									
	Feedback structure									
Conducting DC sessions	Present design	✓	✓	✓	✓	✓	✓	✓	✓	✓
	Receiving feedback	✓	✓	✓	✓	✓			✓	
Post processing	Data analysis	✓	✓		✓		✓	✓	✓	
	Reflections		✓	✓	✓	✓	✓	✓	✓	✓
	Response validation									
	Implications	✓		✓				✓		✓

4.3. DC utility context

In our search for the use of the DC method in the context of HCI and UX, we found that all published works fall under two major context categories: HCI education and UX design research.

4.3.1. DC in UX design research

Researchers in [43] introduced Snakey as a tangible user interface for reservoir engineering for project planning. With the help of two reservoir engineers, researchers conducted a DC study on Snakey prototype to collect feedback about the UX design specifically. Mostafa in [56], recruited domain experts as well to evaluate their virtual reality visualization tool. The goal was to understand user-system interactions in the virtual reality experience better. Clark et al. proposed a software named Fluid for UI design. A main feature of this system is the ability to conduct DC sessions and to record participants' feedback in the system [57]. Reviewing the design with end-users is another option for receiving feedback. Tozser et al. carried out a DC study with police officers to review the design of the Re-Collision software prototype. The goal of this software is to reconstruct car collisions from a top view [44]. Similarly, Wang et al. followed a DC process to evaluate a smart home system design while the development was still in progress [53].

Frauenberger et al. discuss how gathering feedback from children on the autism spectrum was difficult while performing a design critique [12], see Fig. 2. for DC sessions photographs. Similarly, Martin et al. gathered design critique feedback from older adults who were not necessarily familiar with technology [40]. A similar challenge was highlighted by [19], where conducting DC sessions with seniors living with dementia, caregivers, domain experts, and mixed-reality developers required different interview tactics for each round of design critiquing. Therefore, understanding the potential end-users' nature before conducting a DC session is essential. Furthermore, it is evident that studies that involve multiple participant categories require making decisions about the feedback structure in advance.

Using crowd sources from online communities can generate faster feedback at lower costs. However, the quality of feedback should be checked when the crowds are not design experts [38]. Luther et al. [38] described an application called CrowdCrit, which is a web-based system that allows designers to receive critiques from non-experts online, see Fig. 3. CrowdCrit was evaluated in multiple studies to investigate the nature of crowd-sourced feedback. These evaluations reported that the feedback spurred designers' inspiration as diversity positively influenced decision-making. It is worth mentioning that crowd-UX studies showed structured feedback focused on the visual design and aesthetic of the product and less focus on user stories and context of use [39,46].

Fig. 2. Gathering feedback from a child with autism.

4.3.2. DC in HCI education

The DC method is commonly used in studio-based courses [58–60]. This approach is long practiced in other academic fields such as architecture, industrial design, and engineering. Using the DC in computer science education is relatively new. In our SLR, we collected and analyzed papers conducting DC in the context of HCI education. One of the advantages of the DC in education is to establish design guidelines, especially in emerging technologies such as augmented and virtual reality [5,9]. In most of the papers, we found instructors would

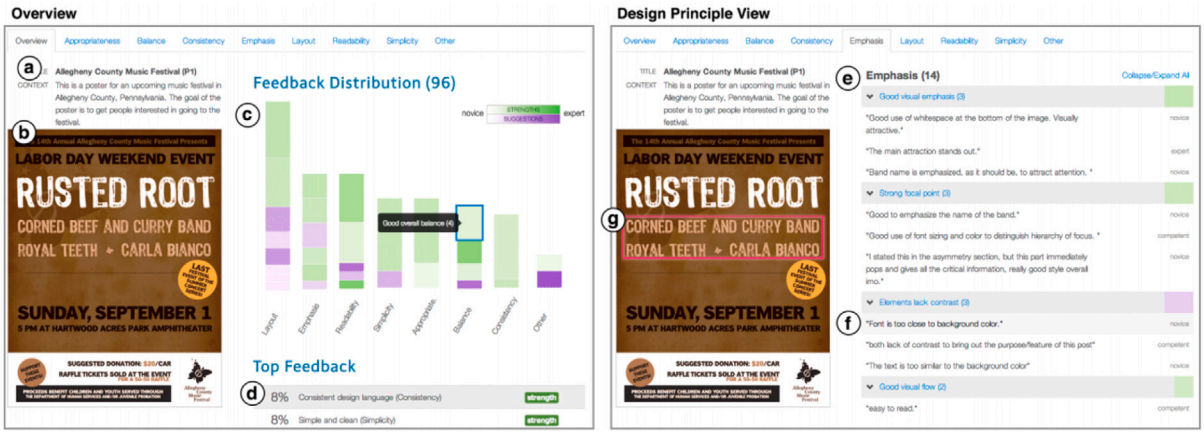


Fig. 3. Receiving DC feedback from crowds.

task students to play the designer role and critique role interchangeably. This process is believed to enhance the student's learning skills in providing and extracting feedback from DC sessions. For instance, Wright et al. had their students extract user requirements, create an abstract design based on these requirements, and receive feedback for their designs [45]. Harrison et al. used in-class review sessions with mockups and prototypes created by students to show that through design critique, many issues can be solved by breaking free of the mindset of "one problem/one solution" [7]. Billinghurst et al. [9] applied the "I like, I wish, what if [the design aspect]" approach for collecting feedback about immersive user interface designs in an online course.

Miller et al. applied design critiques within a classroom setting to argue that DC can be used for design evaluation, not only design review [61]. In [19], DC sessions were carried out with university students to evaluate the design of a tangible user-interface system prototype to stimulate the visual and auditory senses of humans. Yuan et al. [62] introduced an AR application called ARCritique to help students and instructors to conduct DC sessions remotely by allowing users to 3D scan and share artifacts simultaneously. One of the less applied approaches for the design critique in HCI education, which requires more investigation, is self-critique. In self-critique, a designer or a design team would critique their own work without inviting external participants. In this case, the validity of the feedback would be questionable [63,63,64]. Running online DC sessions is recommended when attending in-person studios is not possible. Conducting this type of DC was explored once in an educational setup in [32]. The study found that online feedback led to meaningful UX and UI design changes.

5. Discussions

Conducting DC studies could involve different participant categories depending on the project's goal. Furthermore, the literature shows various ways of collecting and analyzing DC data. In order to explore these three attributes, we ran a number of Matrix Coding queries in NVivo software to identify the various methods of collecting and analyzing DC data and the different participant categories; see Fig. 4 for detail. This section of the paper thoroughly discusses each of these attributes.

5.1. Participant categories

As illustrated in Fig. 4, there are four major categories of DC participants. The breakdown of the number of studies based on participant category is as follows: 5 studies with potential end-users [12,40,44,46,53], 4 studies with domain experts [10,43,47,55], 3 studies with designers [30,33,42] and 9 studies with students (all 9 studies were conducted in an educational environment) [3,6,7,32,34,35,48,54,61]. Furthermore, we identified 7 studies that reported conducting DC with more than one participant category such as a combination of end-users and domain experts, UX experts and end-users, etc. [19,31,37–39,41,52]. Domain experts appear to be the most popular participant category in studies that conducted DC sessions with more than one participant category. It is important to differentiate between domain experts and UX expert critiques. In our analysis, we identified only one paper that recruited UX experts and end-users as critiques [39]. Further analysis among papers that conducted DC with domain experts shows that the number of participants is relatively low (1–5 participants) as compared to other studies (9+). In some studies, recruiting domain experts can be challenging, such as in [10] where the domain experts are engineers specialized in reservoir design and in [19] where domain experts are assistive technology experts which explains to the low number of participants. As described in [38], the type of feedback collected from UX experts is valuable due to their previous design experience. However, based on our analysis, combining domain experts with end-users is an ideal situation where end-users can provide creative and new perspectives to the UX design. In contrast, domain experts provide more detailed insights. In addition, we found that studies that conducted a DC with a combination of more than one participant category also conducted the response validation step [32,37,41] which could mean that discussing feedback from one group with another increased the validity of the feedback.

5.2. Data analysis

We identified four stand-alone data collection methods: three papers used interviews only [30,53,54], two papers used surveys [6,61], five papers used web-tools [32,37,38,46,55] and three papers used recorded interactive sessions [12,43,47]. Meanwhile, the majority of papers (total of 15) used a mix of more than one data collection method [3,7,10,31,33–35,39–42,44,48,52,65]. Interestingly, most of these 15 papers used a combination of interviews and one other method (surveys or interactive sessions). Therefore, it is fair to say that interviews are the most common method of collecting DC data. This finding can be linked to the previously discussed point, the participant category, where the majority of DC participants were domain experts. The focus of conducting DC with experts is to better understand design elements rather than evaluating system usability. Thus, interviews appear to be the ideal method where a two-way conversation takes place, allowing for deeper discussions. Even in the case of conducting DC with end-users, interviews as a data collection method were performed [53]. Consequently, there could be more potential for fostering DC by conducting online studies and increasing accessibility to various participant categories. The second most popular method of collecting data was using web tools

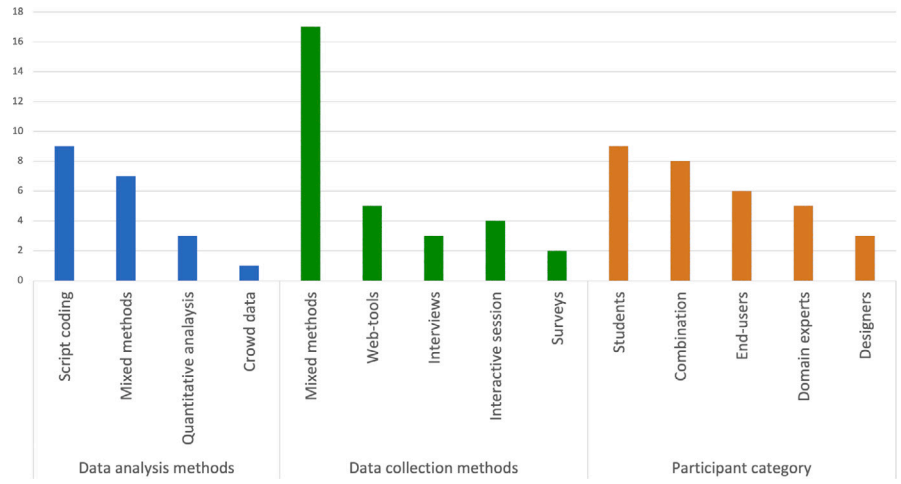


Fig. 4. Studies breakdown based on three attributes: participant category, data collection methods and data analysis methods.

and online forums. It appears that many of the crowd-sourced data collection papers followed this approach when interviews were not the most practical method

5.3. Data analysis methods

We found that one paper [32] used crowd-sourced data and two papers [6,61] used quantitative data analysis, all of which were used in an educational setup. On the other hand, 18 papers used a form of script coding analysis such as textual analysis, thematic analysis, grounded theory, free form text annotation and Klaus Krippendorff content analysis [3,10,12,19,30,31,33–35,37–39,43,44,46,47,53–55]. Six papers in our survey used a combination of script analysis and statistical analysis [7,40–42,48,52]. Evidently, script analysis is the most common method of analyzing DC data. Mainly because most of the DC data were qualitative (interviews and recorded sessions). Therefore, scripting interviews and recorded interactive sessions are ideal for documenting and analyzing data. Secondly, the main aim of a DC study is to better understand UX design elements rather than measuring system usability. Thus, script coding, in this case, can provide deeper and more meaningful insights about the relationship between system requirements, user needs, and UX design.

6. Proposed design critique process

Based on our early findings, we present ten steps and break them down into three major phases: preparation, design critique sessions, and postprocessing (see Fig. 5. In addition, we provide some guidelines for conducting each step. The detailed steps can differ based on the project goal, and we expect UX teams to select a subset of the steps based on a judgement call about what fits into the current project context. However, the three main phases that we present below were present in every paper we found and thus, we recommend going through these three phases when running DC studies.

6.1. Preparation

Proper preparations before the DC sessions yield an organized flow of information. Understanding user needs and defining system purpose are essential steps before conducting a design critique study. Depending on the critique context, different requirement elicitation approaches, such as interviews, questionnaires, field observations, and knowledge gathering, can be helpful as a first step. If the user requirements are pre-discussed, then the first and most important step is to define the purpose of the DC study itself, which was found in almost every paper we included in this survey. Discussing the feedback structure in the preparation phase is essential, especially when it is a collaborative study. Generally, we found two main types of feedback structure: free-form and structured feedback [38,39]. Different feedback formats can be used depending on the data collection method and the participant category (e.g., use surveys with end-users and interviews with domain experts). Based on our review of previous studies [38,39,54,66,67], we suggest the following is to be taken into consideration for more effective feedback collection:

- Functionality: we noticed that if the DC purpose is to review the system functionality, involving UX experts and developers as critiques can be helpful.
- UI design: If the purpose of DC is to review the UI design and aesthetics, then inviting potential end-users was found to be common. Our survey shows that crowdsourced feedback was used more than once for this purpose [18,47]. Because appearance and aesthetics are out of domain experts' scope, we found that involving designers to provide their input was an ideal solution.
- Originality: In papers where the purpose of the DC is to evaluate the design of an emerging application (e.g. mixed reality applications), We found that a combination of the domain experts and end-user participants were recommended. Setting agendas prior to conducting DC sessions is especially helpful when multiple team members conduct the DC sessions. This step allows for creating standard procedures with a streamlined feedback structure.

6.2. Running DC sessions

The main two steps when running a DC study are to present the design to participants and to collect their feedback. Feedback can be gathered in two fashions: structured and open forms [38]. Structured feedback can be in the form of predefined questions and can use methods such as Likert scale questions, up–down voting, or answers to questions. During the critique sessions, taking notes and audio/video recordings are the two main ways to record data. Detailed notes later help in the post-processing phase, especially during the reflections step.

Depending on the method of extracting feedback, the session duration varies. For example, for a structured, asynchronous critique and crowdsource feedback, data collection can be spread out over anywhere between two days to a week [32,68]. However, if the design critique is discussion-based, sufficient time is required to allow for open discussion, typically between 30–90 min [3,34,35]. As was discussed before, structured, semi-structured and open-ended interviews are the most common methods of data collection. Two papers reported that researchers should not consider participants' feedback as personal criticism [69,70]. At the same time, the feedback should be aimed at improving the design itself, not criticizing the designers [69].

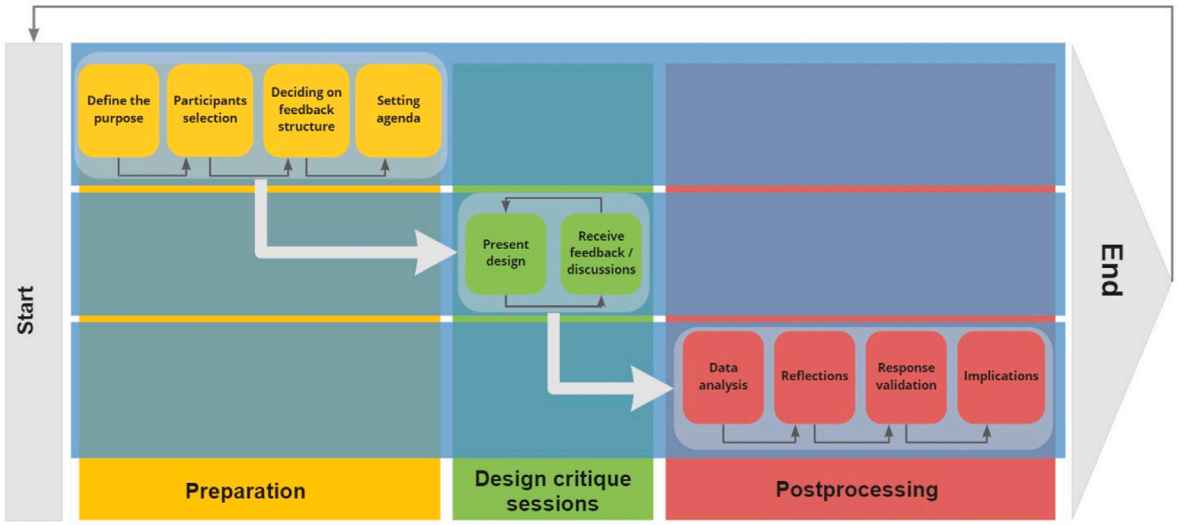


Fig. 5. Proposed steps for a design critique process.

6.3. Postprocessing

Some papers reported conducting a reflection step prior to the data analysis. This approach aligns with most qualitative data analysis methods where 'familiarize yourself with data' is highly encouraged before conducting any analysis. Using recorded notes can be very helpful in bringing back memories from the DC session. Other studies start with qualitative data analysis and then reflect on the findings before deciding on the implications.

Before analyzing the data, decisions about the analysis methods must be made. We found that script coding is the most recommended method of data analysis. The validation step allows a researcher/designer to decide what feedback will be addressed and which will not. Validating the feedback can be done internally among the design/research team. Bias would be a concern in this case. Alternatively, feedback validations could be externalized by including other participants in the validation process (5-Likert scale, up–down voting). Finally, design changes can be made at this point based on the validation phase outcome.

7. Design critique and Agile UX

DC is an approach to receiving productive feedback based on design objectives [4,5]. As a result, DC feedback often targets the conceptual level of a system, while usability testing often deals with a more detailed level. The DC steps can be different depending on the project goals and context. One of the advantages of DC is that it is flexible in nature, where the entire process can be conducted in a single meeting [9] or a sequence of meetings [40] to improve a design. Moreover, it can be synchronous and conducted at a specific time for all participants [9], or asynchronous [5]. It can be an individual effort [63,64], collaborative work [71], or a mix [20]. This flexibility makes DC well suited and adaptable for Agile environments. In this regard, DC can be utilized in the following:

- as part of the design process that runs one iteration ahead of the development iteration by conducting a design critique on low-fidelity prototypes
- as an alternative to Nielsen's discount usability testing approach following the development of the first version(s) of a user interface

The literature discusses that integrating the DC into the UCD process can save time, effort, and cost, specifically during the ideation and prototyping phases. In other papers, it was shown that incorporating UCD into Agile processes is helpful [11]. Our paper proposed a generic DC process based on ten steps and 3 phases. Our paper shows that DC is a lightweight process that can be conducted in one or multiple sessions with various participant categories. Sometimes, critiques can be recruited from within the same organization, and at other times participants can be drawn from the pool of beta testers. The DC process can be executed within the one-iteration-ahead design process or after a UI has been implemented. Design critiques generate ideas and actionable results even when few participants provide feedback. Scheduling synchronous or asynchronous feedback DC sessions with a few people can fit into the short iteration cycle of agile teams. Conducting a DC session can reduce efforts, time and cost spent on usability testing at the end of the system. This is aligned with arguments for discount usability testing that argues that some feedback is better than no feedback, but a key difference is that DC often provides feedback on a conceptual level while (discount) usability testing feedback is often on details of the workflow and UI design. Both kinds of feedback are important, and they can augment each other. More formal usability testing with more participants could likely enhance overall usability, but the increased costs and effort needed to schedule feedback sessions will often conflict with fast, agile iterations. Due to its flexibility and adaptability to a local context, we believe that our proposed DC process has great potential if appropriately integrated into the Agile UX design.

8. Limitations

The design critique process can be beneficial when gathering collaborative feedback for a specific problem. However, Lowgren states that [72]. Therefore, design critiques feedback is subjective to the participants and cannot be recreated, which comes with limitations. Stolterman et al. [18] state that "design critiques are not an objective exercise. Instead, it is highly subjective where each critic comments based on their own experiences and judgements". Because open discussion can be used to evaluate and achieve a solution to a problem, it is easy to be sidetracked by other issues that may arise.

The DC process presented in this paper is based on a sample of literature studies retrieved via search engines which in turn, come with technical limitations as we stated earlier. Although we supplemented our search results by conducting further searches on ACM Digital Library, IEEE Xplore, and Scopus, we might have missed papers published on other databases which not appear in our Google Scholar searches. This search strategy may impacted our sample size leading to an external threat of validity. During the paper selection process, we implemented a double-blind screening protocol to reduce selection

bias. However, this practice does not completely eliminate bias which may cause an internal threat to validity. We followed Cruzes and Dyba's guidelines to conduct thematic analysis on the selected papers. Our approach of coding and generating themes was inductive and reflexive, which is believed to reduce the impact of personal beliefs on the analysis [73]. However, the trustworthiness of our findings are still limited to our interpretation of the data. Finally, since our proposed DC process is entirely based on previous literature studies, thus, its effectiveness has not been empirically tested yet.

9. Conclusion

The design critique is a process used in creative design fields to collect feedback from experts and potential end-users before development. The DC method is emerging in the field of UX due to the need for evaluation methods for new systems where guidelines are not well established. DC studies are not meant to replace usability studies. However, they are considered a different method of gathering structured and free-from feedback in an iterative process at an early stage of development. In this paper, we conducted a systematic literature review study of the DC in the context of HCI and UX. Then, a thematic analysis was conducted on the selected papers. The focus of the thematic analysis was to extract common DC steps for literature studies and discuss the following attributes: participant categories, data collection methods, and data analysis methods. Based on the analysis results, we presented a generic DC process consisting of 10 steps breakdown into three phases. We found that domain experts represent the majority of participants in the DC literature. Conducting one-on-one interviews with participants or combining interviews with other data collection methods, such as recorded interactive sessions, were the most common methods of collecting DC data. Among all the papers we analyzed, script coding analysis, such as thematic analysis, grounded theory, and textual analysis, were the typical data analysis methods in the DC literature. The DC method has great potential for integration into the AgilUX as it aligns with the small, lightweight steps approach in the Agile methods. Finally, we recommend that future researchers further investigate the DC utilization better to explore its impact on usability and systems development progress.

CRediT authorship contribution statement

Lorans Alabood: Conceptualization, Introduction, Research methodology, Screening, Data curation, Formal analysis, Discussions, Conclusion, Writing and editing. **Zahra Aminolroaya:** Introduction, Conceptualization of proposed process, Reference, Figures and tables management, Writing and editing. **Dianna Yim:** Conceptualizing the original draft, Screening, Original analysis, Discussions, Writing and editing. **Omar Addam:** Conceptualizing the original draft, Writing introduction, Analysis, Discussions, Conclusion, Writing. **Frank Maurer:** Conceptualization, Supervision, Writing – reviewing and editing.

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.

References

[1] D.P. Dannels, K.N. Martin, Critiquing critiques: A genre analysis of feedback across novice to expert design studios, J. Bus. Tech. Commun. 22 (2) (2008) 135–159.

[2] B. Hokanson, The design critique as a model for distributed learning, in: The Next Generation of Distance Education, Springer, 2012, pp. 71–83.

[3] C. Hundhausen, D. Fairbrother, M. Petre, The" prototype walkthrough" a studio-based learning activity for human-computer interaction courses, in: Proceedings of the Seventh International Workshop on Computing Education Research, 2011, pp. 117–124.

[4] E. Blevis, Y.-k. Lim, D. Roedl, E. Stolterman, Using design critique as research to link sustainability and interactive technologies, in: International Conference on Online Communities and Social Computing, Springer, 2007, pp. 22–31.

[5] M. Nebeling, User experience and interaction design for AR/vr/MR/XR, 2021, https://www.coursera.org/learn/user-experience-interaction-design-augmented-virtual-mixed-extended-reality (Accessed: 10 Dec 2021).

[6] S. Charleer, J. Klerkx, E. Duval, T.D. Laet, K. Verbert, Towards balanced discussions in the classroom using ambient information visualisations, Int. J. Technol. Enhan. Learn. 9 (2–3) (2017) 227–253.

[7] S. Harrison, D. Tatar, More than a method inventivity: teaching theory, design and innovation, in: Inventivity: Teaching Theory, Design and Innovation in HCI, Proceedings of of HCIEd2006-1 First Joint BCS/IFIP WG13, Vol. 1, 2006, pp. 61–66.

[8] E. Rose, A. Davidson, E. Agapie, K. Sobel, Designing our future students: Introducing user experience to teens through a UCD charette, in: Proceedings of the 34th ACM International Conference on the Design of Communication, 2016, pp. 1–6.

[9] M. Billinghurst, M. Nebeling, Rapid prototyping of XR experiences, in: Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems, 2021, pp. 1–3.

[10] S. Somanath, A. Rocha, H. Hamdi, E. Sharlin, M.C. Sousa, Reservoirbench: An interactive educational reservoir engineering workbench, in: IFIP Conference on Human-Computer Interaction, Springer, 2015, pp. 340–348.

[11] T.S. Da Silva, A. Martin, F. Maurer, M. Silveira, User-centered design and agile methods: a systematic review, in: 2011 AGILE Conference, IEEE, 2011, pp. 77–86.

[12] C. Frauenberger, J. Good, A. Alcorn, H. Pain, Conversing through and about technologies: Design critique as an opportunity to engage children with autism and broaden research (er) perspectives, Int. J. Child-Comput. Interact. 1 (2) (2013) 38–49.

[13] T.S. Da Silva, M.S. Silveira, F. Maurer, F.F. Silveira, The evolution of agile UXD, Inf. Softw. Technol. 102 (2018) 1–5.

[14] D. Fox, J. Sillito, F. Maurer, Agile methods and user-centered design: How these two methodologies are being successfully integrated in industry, in: Agile 2008 Conference, IEEE, 2008, pp. 63–72.

[15] S. Chamberlain, H. Sharp, N. Maiden, Towards a framework for integrating agile development and user-centred design, in: International Conference on Extreme Programming and Agile Processes in Software Engineering, Springer, 2006, pp. 143–153.

[16] J. Ferreira, J. Noble, R. Biddle, Agile development iterations and UI design, in: Agile 2007, AGILE 2007, IEEE, 2007, pp. 50–58.

[17] J. Forlizzi, C. DiSalvo, J. Bardzell, I. Koskinen, S. Wensveen, Quality control: A panel on the critique and criticism of design research, in: CHI'11 Extended Abstracts on Human Factors in Computing Systems, 2011, pp. 823–826.

[18] E. Stolterman, A. Croon Fors, Critical HCI research: a research position proposal, Des. Philos. Pap. 1 (2008).

[19] L. Alabood, F. Maurer, An IoT-based immersive smart home system for seniors with neurocognitive disorders, in: Proceedings of the 3d International Workshop on Empowering People in Dealing with Internet of Things Ecosystems, Vol. 3172, Co-Located with AVI, Rome, Italy, CEUR-WS, 2022, pp. 10–15.

[20] Z. Aminolroaya, S. Dawar, C.B. Josephson, S. Wiebe, F. Maurer, Virtual reality for understanding multidimensional spatiotemporal phenomena in neuroscience, in: Companion Proceedings of the 2020 Conference on Interactive Surfaces and Spaces, 2020, pp. 85–89.

[21] B. Kitchenham, S. Charters, Guidelines for performing systematic literature reviews in software engineering, Citeseer, 2007.

[22] D.S. Cruzes, T. Dyba, Recommended steps for thematic synthesis in software engineering, in: 2011 International Symposium on Empirical Software Engineering and Measurement, IEEE, 2011, pp. 275–284.

[23] E. Laukkanen, J. Itkonen, C. Lassenius, Problems, causes and solutions when adopting continuous delivery—A systematic literature review, Inf. Softw. Technol. 82 (2017) 55–79.

[24] J. Saad, S. Martinelli, L.S. Machado, C.R. de Souza, A. Alvaro, L. Zaina, UX work in software startups: A thematic analysis of the literature, Inf. Softw. Technol. 140 (2021) 106688.

[25] W.M. Watanabe, K.R. Felizardo, A. Candido Jr., É.F. de Souza, J.E. de Campos Neto, N.L. Vijaykumar, Reducing efforts of software engineering systematic literature reviews updates using text classification, Inf. Softw. Technol. 128 (2020) 106395.

[26] J.L. Coram, R. Iverson, A. Ackerman, Astrotouch: a multi-touch digital desktop for astrodynamics, in: Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces, 2013, pp. 11–14.

[27] N. Bonnardel, T. Sumner, Supporting evaluation in design, Acta Psychol. 91 (3) (1996) 221–244.

[28] L. Hallnäs, Design process: design rationale the affective diary, in: Proceedings of the 5th Nordic Conference on Human-Computer Interaction: Building Bridges, 2008, pp. 565–566.

[29] S. Greenberg, B. Buxton, Usability evaluation considered harmful (some of the time), in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2008, pp. 111–120.

[30] S.R. Herring, B.R. Jones, B.P. Bailey, Idea generation techniques among creative professionals, in: 2009 42nd Hawaii International Conference on System Sciences, IEEE, 2009, pp. 1–10.

[31] R. Macefield, An overview of expert heuristic evaluations, UX Matters 2 (2014) 2.

[32] A. Xu, H. Rao, S.P. Dow, B.P. Bailey, A classroom study of using crowd feedback in the iterative design process, in: Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing, 2015, pp. 1637–1648.

[33] S. Dow, J. Fortuna, D. Schwartz, B. Altringer, D. Schwartz, S. Klemmer, Prototyping dynamics: sharing multiple designs improves exploration, group rapport, and results, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2011, pp. 2807–2816.

[34] C.D. Hundhausen, D. Fairbrother, M. Petre, An empirical study of the ‘‘prototype walkthrough’’ a studio-based activity for HCI education, ACM Trans. Comput.-Hum. Interact. 19 (4) (2012) 1–36.

[35] C. Hundhausenm, D. Fairbrother, M. Petre, The ‘‘prototype walkthrough’’: a studio-based learning activity for the next generation of HCI education, 2010.

[36] A.Z. Asha, F. Anzum, P. Finn, E. Sharlin, M. Costa Sousa, Designing external automotive displays: VR prototypes and analysis, in: 12th International Conference on Automotive User Interfaces and Interactive Vehicular Applications, 2020, pp. 74–82.

[37] M.W. Easterday, D. Rees Lewis, C. Fitzpatrick, E.M. Gerber, Computer supported novice group critique, in: Proceedings of the 2014 Conference on Designing Interactive Systems, 2014, pp. 405–414.

[38] K. Luther, J.-L. Tolentino, W. Wu, A. Pavel, B.P. Bailey, M. Agrawala, B. Hartmann, S.P. Dow, Structuring, aggregating, and evaluating crowdsourced design critique, in: Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing, 2015, pp. 473–485.

[39] H. Schneider, K. Frison, J. Wagner, A. Butz, CrowdUX: a case for using widespread and lightweight tools in the quest for UX, in: Proceedings of the 2016 ACM Conference on Designing Interactive Systems, 2016, pp. 415–426.

[40] A. Martin-Hammond, S. Vemireddy, K. Rao, Engaging older adults in the participatory design of intelligent health search tools, in: Proceedings of the 12th EAI International Conference on Pervasive Computing Technologies for Healthcare, 2018, pp. 280–284.

[41] T.M. Amabile, The social psychology of creativity: A componential conceptualization, J. Personal. Soc. Psychol. 45 (2) (1983) 357.

[42] M. Antona, A. Savidis, C. Stephanidis, A process-oriented interactive design environment for automatic user-interface adaptation, Int. J. Hum. Comput. Interact. 20 (2) (2006) 79–116.

[43] J. Harris, J. Young, N. Sultanum, P. Lapides, E. Sharlin, M. Costa Sousa, Snakey: A Tangible User Interface for Well Path Planning in the Context of Reservoir Engineering, Tech. rep., University of Calgary, 2011.

[44] M. Tozser, N. Sultanum, E. Sharlin, K. Rutherford, C. Foster, Re-collision: a collision reconstruction forensics tabletop interface, in: Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces, 2012, pp. 81–84.

[45] T. Wright, J. Noble, S. Marshall, Using a system of tutorials and groups to increase feedback and teach user interface design, in: Proceedings of the 7th Australasian Conference on Computing Education-Volume 42, 2005, pp. 187–192.

[46] Y.-C. Yen, S.P. Dow, E. Gerber, B.P. Bailey, Social network, web forum, or task market? Comparing different crowd genres for design feedback exchange, in: Proceedings of the 2016 ACM Conference on Designing Interactive Systems, 2016, pp. 773–784.

[47] N. Sultanum, E. Sharlin, M.C. Sousa, D.N. Miranda-Filho, R. Eastick, Touching the depths: introducing tabletop interaction to reservoir engineering, in: ACM International Conference on Interactive Tabletops and Surfaces, 2010, pp. 105–108.

[48] Y.J. Reimer, K. Cennamo, S.A. Douglas, Emergent themes in a UI design hybrid-studio course, in: Proceedings of the 43rd ACM Technical Symposium on Computer Science Education, 2012, pp. 625–630.

[49] R.I. Clarke, From Default to Design: Design-Based Assessment for Libraries and Librarianship.

[50] J. Li, E. Sharlin, S. Greenberg, M. Rounding, Designing the car iwindow: exploring interaction through vehicle side windows, in: CHI’13 Extended Abstracts on Human Factors in Computing Systems, 2013, pp. 1665–1670.

[51] M.A. Dunlap, Science Caching: Applying Geocaching to Mobile Citizen Science, (Master’s thesis), University of Calgary, 2013.

[52] A. Arakelyan, I. Shmorgun, S. Sousa, Incorporating values into the design process: The case of e-textbook development for estonia, in: International Conference on Web-Based Learning, Springer, 2014, pp. 274–281.

[53] H. Wang, Y.-P. Ma, H.-C. Lin, T. Jeng, S. Liou, Methodology for Ideation of Interaction Design, CUMINCAD, 2009.

[54] B.T. Christensen, L.J. Ball, Dimensions of creative evaluation: Distinct design and reasoning strategies for aesthetic, functional and originality judgments, Des. Stud. 45 (2016) 116–136.

[55] Y. Kou, C.M. Gray, Supporting distributed critique through interpretation and sense-making in an online creative community, Proc. ACM Hum.-Comput. Interact. 1 (CSCW) (2017) 1–18.

[56] A.E. Mostafa, PetroVis & FractVis: Interactive Visual Exploration of High-dimensional Oil and Gas Data, (Master’s thesis), University of Calgary, 2013.

[57] C. Clark, D. Ayotte, A. Basman, J. Treviranus, About us, with us: The fluid project’s inclusive design tools, in: International Conference on Universal Access in Human-Computer Interaction, Springer, 2016, pp. 172–182.

[58] R. Grasser, Experiences in implementing a studio component into a course for novice web developers, in: Proceedings of the 16th Annual Joint Conference on Innovation and Technology in Computer Science Education, 2011, pp. 23–27.

[59] O. Shaer, D. Tosca, Teaching tangible interaction remotely during COVID-19: Transcending physical boundaries, IEEE Pervasive Comput. 20 (2) (2021) 49–53.

[60] S. Harrison, M. Back, D. Tatar, " It’s just a method!" a pedagogical experiment in interdisciplinary design, in: Proceedings of the 6th Conference on Designing Interactive Systems, 2006, pp. 261–270.

[61] S.R. Miller, B.P. Bailey, A. Kirlik, Exploring the utility of Bayesian truth serum for assessing design knowledge, Hum.–Comput. Interact. 29 (5–6) (2014) 487–515.

[62] Y. Li, D. Hicks, W.S. Lages, S.W. Lee, A. Sharma, D.A. Bowman, Arcritique: Supporting remote design critique of physical artifacts through collaborative augmented reality, in: 2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops, VRW, IEEE, 2021, pp. 585–586.

[63] M. Conley, How to critique your own design projects: 5 questions to ask, 2017, https://blog.hubspot.com/agency/critique-your-own-des (Accessed: 10 Dec 2021).

[64] K. Liez, How to self-critique your design projects, 2013, https://naldzgraphics.net/self-critique-design-projects (Accessed: 10 Dec 2021).

[65] A. Garcia, T.S.d. Silva, M.S. Silveira, Artifact-facilitated communication in agile user-centered design, in: International Conference on Agile Software Development, Springer, 2019, pp. 102–118.

[66] J. Nielsen, Usability engineering at a discount, in: Proceedings of the Third International Conference on Human-Computer Interaction on Designing and using Human-Computer Interfaces and Knowledge Based Systems, second Ed., 1989, pp. 394–401.

[67] C. Remy, O. Bates, A. Dix, V. Thomas, M. Hazas, A. Friday, E.M. Huang, Evaluation beyond usability: Validating sustainable HCI research, in: Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems, 2018, pp. 1–14.

[68] O. Ayalon, E. Toch, Crowdsourcing privacy design critique: An empirical evaluation of framing effects, in: Proceedings of the 51st Hawaii International Conference on System Sciences, 2018.

[69] J. Czar, Mastering design critiques, 2019, https://uxdesign.cc/design-critique-design-better-products-through-team-collaboration-4c4f26a7be5f (Accessed: 10 Dec 2021).

[70] R. Sens, UX design 101: Articulating a viewpoint — Communicating design decisions effectively, 2019, https://blog.prototypr.io/ux-design-101-design-critiques-presenting-discussing-your-work-5b2dcb26e52f (Accessed: 10 Dec 2021).

[71] T. Seyed, M. Costa Sousa, F. Maurer, A. Tang, SkyHunter: a multi-surface environment for supporting oil and gas exploration, in: Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces, 2013, pp. 15–22.

[72] J. Löwgren, Applying design methodology to software development, in: Proceedings of the 1st Conference on Designing Interactive Systems: Processes, Practices, Methods, & Techniques, 1995, pp. 87–95.

[73] V. Braun, V. Clarke, Reflecting on reflexive thematic analysis, Qualit. Res. Sport, Exercise Health 11 (4) (2019) 589–597.