Task-Based Open Card Sorting: Towards a New Method to Produce Usable Information Architectures

Christos Katsanos(✉), Vasileios Christoforidis, and Christina Demertzi

Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece
{ckatsanos,chrvaskos,demertzi}@csd.auth.gr

Abstract. Open card sorting is the most widely used HCI technique for designing user-centered Information Architectures (IAs). The method has a straightforward data collection process, but data analysis can be challenging. Open card sorting has been also criticized as an inherently content-centric technique that may lead to unusable IAs when users are attempting tasks. This paper proposes a new variant of open card sorting, the Task-Based Open Card Sorting (TB-OCS), which considers users' tasks and simplifies data analysis. The proposed method involves two phases. First, small groups of participants perform classic open card sorting. Then, each participant performs findability tasks using each IA produced by the rest participants of the same group and their first-click success is measured. Analysis of the collected data involves simply calculating the first-click success rate per participants' IA and selecting the one with the highest value. We have also developed a web-based software tool to facilitate the conduction of TB-OCS. A within-subjects user testing study found that open card sorting produced IAs that had significantly higher first-click success rates and perceived usability ratings compared to the IAs produced by TB-OCS. However, this may be due to parameters of the new method that require finetuning, thus further research is required.

Keywords: Card Sorting · Information Architecture · IA · Task-Based Open Card Sorting

1 Introduction

1.1 Card Sorting and Information Architecture

Card sorting is an established technique used to discover how participants might arrange and organise information that makes sense to them [1]. Many studies in the literature have used the method in a variety of contexts, such as exploring participants' mental models for mental wellness [2], programming [3], cybersecurity [4] and haptic devices [5]. Open card sorting has been also used to group HCI design guidelines [6–8] or validate HCI tools that support the design of interactive systems [9–11]. Card sorting is most frequently employed, however, to assist in the creation or assessment of Information Architectures (IAs) [1, 12, 13].

H. Mori and Y. Asahi (Eds.): HCII 2023, LNCS 14015, pp. 68–80, 2023.
https://doi.org/10.1007/978-3-031-35132-7_5

The IA of an interactive system specifies how its content is structured and labeled [13]. Open card sorting asks participants to organize a set of provided labels that describe content items (the cards), written on paper or on any card sorting software tool, using their own groupings and category names [1]. Other variants that have been proposed in the literature include closed card sorting [1], hybrid card sorting [14] and modified Delphi card sorting [15]. In a closed card sort, participants organize a set of provided cards into a set of provided named categories. In a hybrid card sort, participants can place the provided cards into provided categories or make their own categories. In a Modified-Delphi card sort, the first participant performs an open sort, and each subsequent participant modifies it until consensus is reached.

1.2 Open Card Sorting

Open card sorting is the most widely-used variant to design or evaluate the IA of interactive systems [1, 16]. Previous research has explored various questions related to open card sort data collection and analysis.

Research has shown that a sample size of 15 to 20 participants is enough for open card sorts [17, 18]. Participants may complete an open card sort in anywhere between 20 and 60 min, depending on the number of cards [14]. There shouldn't be less than 30 cards to sort since it can be difficult to establish groups, and there shouldn't be more than 100 cards because participants might become tired or lost [1]. For a large set of cards, a sub-sample sorting approach has been proposed: if each participant sorts 60% of the total set and there are 30–40 participants involved, then the obtained data are highly similar to sorting done on the full set of cards [19]. Recent research provides support for the validity and reliability of open card sorts [20–22], but has also found that the results are significantly affected by participants' characteristics, such as sense of direction and self-efficacy [23, 24].

Studies [25, 26] have also compared manual card sorting with physical cards against electronic card sorting using software. No differences have been found in the obtained results. However, the participants' time spent sorting cards using software was significantly longer than their time spent sorting cards physically, especially for those who did not speak English as their first language [26]. Research [27] has also examined the usability of software tools for card sorting. It was found that researchers and participants preferred different card sorting tools. However, more current study is needed in this area since new card sort tools, such the open source CardSorter [28], have appeared and most tools in the existing study [27] are no longer supported or have substantially evolved.

There is a rather large body of research on analysis of open card sort data, which is the most challenging part [29–31]. Various open card sort data analysis methods have been proposed in the literature, including tabulations [1] and graph visualizations of the data [32], factor analysis [33], general purpose clustering algorithms (e.g. hierarchical clustering, k-means clustering, multidimensional scaling) [1, 12, 14, 16, 31, 34, 35] and specialized algorithms developed for clustering open card sort data [29, 30, 36]. Righi and colleagues [31] present best practices for card sort analysis. Spencer [1] argues that both qualitative and quantitative analysis should be employed. Nawaz reports that analyzing the same data using different approaches results in varied IAs [35].

1.3 Research Motivation

This paper presents a new variant of open card sorting, the Task-Based Open Card Sorting (TB-OCS). Our motivation for proposing this new variant was twofold.

On the one hand, card sorting has been criticized for not considering users' tasks, which may lead to unusable IAs [37]. Participants may sort the cards without considering what the content is about or how they would use it to complete a task [37]. TB-OCS considers users' tasks. On the other hand, open card sort data analysis remains a rather challenging task [29–31]. TB-OCS simplifies data analysis, but this comes at a cost of an increase in the complexity of running the card sort.

In the following, we first present the TB-OCS method. Next, we present a two-phased study that compares the proposed method with the open card sorting method. In the first phase, the same participants sort the same cards following both the open card sorting and the TB-OCS approach. In the second phase, a different sample of participants interacts with two functional prototypes, one for the IA created by the open card sort and one for the IA created by the proposed method, and usability metrics are compared. The methodology and results of these two studies are reported in the following two sections, and the paper concludes with a discussion of the findings and future research directions.

2 Task-Based Open Card Sorting

2.1 Procedure

The TB-OCS method involves the following steps:

1. Participants are divided into small groups (e.g., 3–5 participants per group). A small number of participants per group is required for practical purposes so that the total number of tasks to be performed in the final step of the method is manageable (see in the following).
2. Each participant of the group performs an open card sorting. Each such sorting is considered the participant's proposal for the IA (participant's IA). No data analysis is required for the open card sort data as each set of participant's groupings corresponds to one IA candidate.
3. Each participant of the group performs tree testing [38, 39] in each IA created by the rest participants of the group. In a tree test, also known as reverse card sorting or card-based classification evaluation [40], participants are presented with an IA and are asked what they would select in order to accomplish a task. For TB-OCS, the study facilitator notes the total number of successfully completed tasks and the total number of tasks attempted per participant's IA.

2.2 TB-OCS Software Tool

We have developed a web-based software tool to facilitate the conduction of TB-OCS. This tool is implemented in React and is freely available as an open-source project at https://github.com/chrvaskos/card-sorting.

Figure 1 presents the user interface of the TB-OCS tool while a user performs an open card sort (second step of the TB-OCS method). As shown in the figure, the tool provides a list of the cards to sort (Fig. 1, left), which can be easily inserted as simple text in the corresponding file of the tool. Each card can be dragged and dropped into either a new category (plus symbol) or an existing one (represented as a box). This drag and drop moving of cards can be repeated as many times as the participant wants. Naming a category is done by clicking on the top of the category box and editing its title; the default title is “New category”.

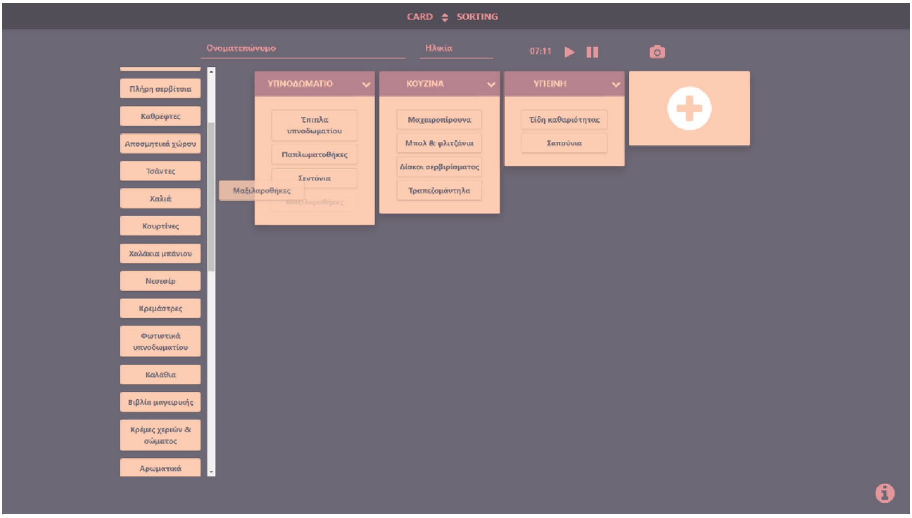

Fig. 1. The user interface of the TB-OCS tool while a user performs an open card sort. The cards and categories are in Greek, the native language of our study participants.

The TB-OCS tool includes functionality to also support the third step of the TB-OCS method, which involves tree testing the participants’ IAs. First, there is a down arrow symbol to the right of the category names which can be pressed to hide or unhide the cards placed in the corresponding category. Hiding the content items of the categories is required for tree testing. In addition, there is a camera icon (Fig. 1, top-right) which downloads a screenshot of the tool user interface when pressed. This enables easy creation of an IA screenshot for the needs of tree testing. The tool also provides a text field for optionally adding the participant’s name or id (Fig. 1, top-left) so that an IA screenshot can be easily associated with a specific participant.

Furthermore, there is an icon for instructions to the user (Fig. 1, bottom-right), which are displayed as a tooltip. These text instructions can be easily inserted as text in the corresponding file of the tool. Finally, the TB-OCS tool provides an embedded timer, which can be used by the participant in order to measure the session time for the open card sort. This timer provides the typical controls for starting, pausing and stopping it. Such functionality can be particularly useful when the open card sort is performed asynchronously (e.g., participants are instructed to time their session and send a screenshot with the session time value embedded).

It should be noted that the TB-OCS tool can also be used to facilitate the collection of data in a typical open card sort. The collected data can then be exported and analyzed with other tools, such as Casolysis 2.0 [41], which is also what we did for the needs of the study described in the following (see Sect. 3.4).

2.3 Data Analysis

The data produced by TB-OCS is a list with all the IAs produced by participants' open card sorting and the total number of successful and attempted tasks per participant's IA. Analysis of the collected data involves simply calculating the first-click success rate per participant's IA (i.e., the number of successfully completed tasks for the IA divided by the number of tasks attempted). The proposed IA is the one with the highest first-click success rate.

3 Card Sorting Study

The card sorting study employed both the open card sort method and the proposed TB-OCS method on the same group of cards with the same group of participants. In practice, only the TB-OCS method was employed for the data collection part given that open card sorting is a step of the proposed method. However, data analysis was separate per method. In the following, we describe the methodology and results of the card sorting study.

3.1 Participants

A total of 20 participants, 10 females and 10 males, with mean age 28.4 years (SD = 9.7) were involved in the card sorting study. All the participants were native speakers of Greek, the language used in the cards. They were volunteers recruited by the authors and they were not compensated for their participation.

3.2 Cards Selection

The card sort was about an existing eshop retailing various housewares. We selected a website domain that does not require any specialized knowledge so that no screening criteria would be required for participants and their recruitment would be easier.

Following Spencer's [1] recommendations, a total of 46 cards were chosen from the website. All cards were provided in Greek and were items selected from the lowest level of the website menu as it was available at the time of the study preparation. Examples of these cards translated into English are the following: "Cutlery", "Bathroom curtains", "Carpets", "Knobs", "Hangers", "Photo frames", "Candles", "Pillowcases", "Mirrors", "Blankets".

3.3 Instruments and Procedures

First, the participants were split into five groups. Each group had four participants. Due to the COVID-19 restrictions, the sessions had to be performed from a distance. We used a Discord server to coordinate communication with participants and our custom-built TB-OCS tool to facilitate the card sorting.

Next, each group of participants was invited to connect to a Discord voice channel on a pre-agreed time per group. This voice channel was used for communication between the study facilitators (i.e., two of the authors) and the participants as a group. The participants were welcomed, provided their consent for study participation and then were asked to read the study instructions. These instructions were available in a Discord text channel and included a hyperlink to the TB-OCS tool used to mediate the card sorting. Subsequently, participants were instructed to mute their microphones and perform a typical online open card sorting. At any point, the participants could privately talk with the study facilitators using Discord direct messaging or a one-to-one voice call in case they needed technical help, or they had a question.

After completing the open card sorting, each participant used Discord direct messaging to send to the study facilitators two screenshots of their groupings. Screenshot1 presented only the categories created by each participant, whereas the Screenshot2 showed the full groupings (i.e., categories and cards placed in each category). These two screenshots were easily captured by participants through the functionality provided by the TB-OCS tool (Fig. 1, camera icon) and were used in the final step of the proposed TB-OCS method, as described in the following.

After a brief break of five minutes, each participant had received through Discord direct messaging the following: a) four images showing only the categories created by the rest members of the group (Screenshot1), b) three task descriptions, each of which asked for locating a specific product on the eshop. For each Screenshot1, participants were asked to select the category that they would click to find each product and write it to the study facilitators using Discord direct messaging. The study facilitators used the corresponding Screenshot2 to decide whether the correct category had been selected. In addition, they entered in a spreadsheet the following data per participant's IA: a) total number of successful tasks, b) total number of attempted tasks. Administration of the images with the categories and the tasks was counterbalanced to minimize order effects.

3.4 Data Analysis and Results for Open Card Sorting

Open card sorting data were collected from 20 participants who performed individual sortings. According to research [17, 18], open card sorts require at least 15 users to produce reliable data, therefore our sample was sufficient.

The collected card sorting data were analyzed combining exploratory and statistical analysis [1]. Our analysis was mediated by Casolysis 2.0 [41], a free software tool that supports a variety of methods for analyzing card sort data. First, the open card sort data were exported from TB-OCS and imported into Casolysis 2.0 as a csv file. Next, the visualization produced by Casolysis 2.0 using multidimensional scaling (MDS) was inspected in order to get a first understanding of the data. Subsequently, we explored the dendrogram produced by average-linkage hierarchical clustering. A dendrogram, a tree diagram that shows a hierarchy of groupings based on the dissimilarity of content items, is the main result of hierarchical cluster analysis. Then, we used the tool functionality to define standardized labels for every group. Each standardized label was the one that "had been used by most participants or represented the idea most clearly" [1]. Subsequently, we reinspected the MDS visualization and average-linkage dendrogram. This process was iterated and was greatly facilitated by the Casolysis 2.0 hold functionality that enables fixing individual card groups that are no longer considered in the next processing step. The latter made it possible to flexibly explore the solution space without losing intermediate results.

3.5 Data Analysis and Results for Task-Based Open Card Sorting

The first step in analyzing the data produced by TB-OCS was to calculate the first-click success rate per participant's IA. To this end, we used the spreadsheet produced by the study facilitators and simply added a column that divided the total number of successful tasks by the total number of attempted tasks for each participant's IA. The mean IA success rate was 57% (SD = 22%) and ranged from 17% to 100%. According to the TB-OCS method, we selected as the proposed IA for the eshop the one with the highest success rate, which in our case was 100%.

4 User Testing Study

The within-samples user testing study compared usability metrics for the IA produced by the open card sorting method (hereafter OCS eshop) against the IA produced by the proposed TB-OCS method (hereafter TB-OCS eshop). In the following, we describe the methodology and results of this user testing study.

4.1 Participants

The user testing study involved 30 participants, 14 females and 16 males, with mean age 31.8 years (SD = 13.3). All the participants were native speakers of Greek, the language of the provided prototypes and questionnaires. They were recruited as volunteers by the authors, and they did not receive any payment for taking part.

4.2 Prototypes

Two functional prototypes were created for the eshop. The prototypes shared the same overall appearance and feel and featured a top navigation menu. They differed only in their IA. One eshop implemented the IA produced by the open card sorting method and the other eshop the IA produced by the proposed TB-OCS method. The prototypes were created using HTML5, CCS3 and JavaScript, and were made available online through a web server.

4.3 Instruments and Procedures

Due to the COVID-19 pandemic, participants were asked to attend a Discord video-conferencing call with the study facilitator (one of the authors). There they were first welcomed and provided their consent for study participation.

Next, the participants received the hyperlinks for each eshop functional prototype and performed five tasks in both prototypes. Each task asked participants to find a specific item to buy: a) cookbook, b) mirror, c) stationery, d) fragrant cards, and e) vanity bag. These items were selected because they were categorized differently in the two eshop versions. To reduce order effects, the order of both the eshop versions and tasks was counterbalanced.

Participants used screen sharing so that the facilitator could observe their interactions. The facilitator recorded whether they made the right choice with their first click (first-click success) and how long each task required (time on task). After performing all the tasks in an eshop version, participants received a hyperlink to complete the System Usability Scale (SUS) [42] in Greek (SUS-GR) [43, 44]. SUS is a standardized scale that measures perceived usability. It has 10 questions and yields a final score between 0 and 100, the higher the score the more usable the system. In agreement with previous studies [43, 44], SUS-GR was found to have high internal reliability in our dataset; Cronbach's $\alpha = 0.811$, $N = 10$ items.

The Google Forms service was used to create and distribute the study questionnaire. IBM SPSS Statistics 27 was used for the statistical analysis of the collected data.

4.4 Data Analysis and Results

Table 1 presents descriptive statistics of the dependent variables measured in the user testing study. In all subsequent statistical analyses, the effect size r was calculated according to the formulas reported in [45].

Table 1. Descriptive statistics of the dependent variables measured in the user testing study.

Eshop IA version	Variable	Mean	Mdn	SD	95% C.I
Open card sorting	First-click success (%)	72.67	80.00	17.01	(66.32, 79.02)
Task-based open card sorting	First-click success (%)	52.00	60.00	19.37	(44.77, 59.23)
Open card sorting	Time on task (sec)	13.91	12.90	4.99	(12.05, 15.77)
Task-based open card sorting	Time on task (sec)	15.48	14.63	6.41	(13.09, 17.87)
Open card sorting	SUS score (0–100)	90.00	92.50	11.03	(85.88, 94.12)
Task-based open card sorting	SUS score (0–100)	76.42	78.75	17.81	(69.77, 83.07)

First-Click Success. A Shapiro-Wilk test found that the distribution of the differences in the first click success for the OCS eshop and the TB-OCS eshop did not deviate significantly from a normal distribution; W(30) = 0.933, p = 0.060. Given that the p value was rather close to the 0.05 threshold, the histogram, and skewness and kurtosis values were also studied. They were found to support the Shapiro-Wilk finding. Thus, a parametric test was used to compare participants' first-click success between the two conditions. A two-tailed dependent t-test found that participants were significantly more successful with their first click in the OCS eshop (M = 72.67%, SD = 17.01%) compared to the TB-OCS eshop (M = 52.00%, SD = 19.37%); t(29) = 4.356, p< 0.001, r = 0.629.

Time on Task. Shapiro-Wilk analysis showed that the assumption of normality was violated for the differences in the task times of the two conditions; W(30) = 0.919, p = 0.026. Thus, a non-parametric test was used to compare participants' time on task between the OCS eshop and the TB-OCS eshop. A two-tailed Wilcoxon signed-rank test found that the eshop IA version did not significantly affect participants' time on task; z = 1.512, p = 0.131. Participants average time on task in the OCS eshop (Mdn = 12.90 s) and the TB-OCS eshop (Mdn = 14.63 s) was similar.

SUS Score. A Shapiro-Wilk test found that the assumption of normality was violated for the differences in the SUS score of the two conditions; W(30) = 0.879, p = 0.003. Thus, a non-parametric test was used to compare participants' SUS score between the OCS eshop and the TB-OCS eshop. A two-tailed Wilcoxon signed-rank test found a significant effect of the eshop IA version on participants' SUS score; z = 4.258, p < 0.001, r = 0.550. Participants provided significantly higher SUS score for the OCS eshop (Mdn = 92.50) compared to the TB-OCS eshop (Mdn = 78.75).

5 Discussion and Conclusion

Open card sorting is an important method for HCI research and practice, and thus there are many publications on how to conduct it and analyze its data. However, open card sorting does not consider users' goal-directed behaviour when interacting with IAs and thus might result in unusable IAs [37]. Additionally, analysis of open card sorting data remains the main challenge of the method [29–31].

This paper proposed TB-OCS, a new card sorting variant that attempts to address these two limitations of open card sorting. A software tool, named TB-OCS tool, has been also developed to facilitate TB-OCS data collection. A two-phase study investigated whether the new method produces more usable IAs compared to open card sorting. In the first phase, 20 participants were involved in a card sorting study with 46 cards from an eshop. The study was mediated by our TB-OCS tool. This phase produced two IAs for the eshop: one based on analysis of open card sort data, and one based on analysis of TB-OCS data. In the second phase, 30 participants were involved in a within-samples user testing study that compared two functional prototypes, one per aforementioned IA. Results showed that users interacting with the IA produced by the TB-OCS method made significantly less correct first clicks and provided significantly lower perceived usability ratings compared to when interacting with the IA produced by open card sorting method. No significant difference was found for the time required to find products.

On the one hand, the proposed method greatly simplified the analysis of the collected data, which is the main challenge in open card sorting. And it did so by increasing the session time by only 8 min and 10 s on average; 5 min break between TB-OCS step2 and step3 plus 3 min and 10 s on average for TB-OCS step3. Of course, the new method also increased the complexity of running the sort for the facilitators. However, we believe that this can be alleviated with some experience and/or pilot testing of the study.

On the other hand, the open card sorting method was found to produce more usable IAs compared to the proposed method. Although we did not expect this finding, additional studies are required to investigate if it is generalizable. If it is indeed generalizable, then this would provide support against the critique to the open card sort method that it is too content-centric and may lead to unusable IAs. If it is not generalizable, then we need to explore why the new method works well is some cases and not in some others. We speculate that it might be related to parameters of the new method that need finetuning, such as the number of groups of participants (5 in this study), the number of participants per group (4 in this study), and the number of tree testing tasks (3 in this study). For example, increasing the number of tree testing tasks could provide an IA that has increased overall findability, but this would also increase the session time.

In conclusion, this research found that the classic open card sorting technique leads to more usable information architectures compared to the proposed technique. However, this may be due to parameters of the new technique that were not explored in this paper. Therefore, additional research is required in order to draw safe conclusions about the proposed technique.

Acknowledgments. We would like to thank the anonymous participants that volunteered to participate in our studies and thus made this research possible.

References

1. Spencer, D.: Card Sorting: Designing Usable Categories. Rosenfeld Media, Brooklyn (2009)
2. Kelley, C., Lee, B., Wilcox, L.: Self-tracking for mental wellness: understanding expert perspectives and student experiences. In: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems, pp. 629–641. ACM, New York (2017). https://doi.org/10.1145/3025453.3025750
3. Dorn, B., Guzdial, M.: Learning on the job: characterizing the programming knowledge and learning strategies of web designers. In: Proceedings of the 28th International Conference on Human Factors in Computing Systems (CHI 2010), Atlanta, Georgia, USA, pp. 703–712 (2010). https://doi.org/10.1145/1753326.1753430
4. Jeong, R., Chiasson, S.: "Lime", "open lock", and "blocked": children's perception of colors, symbols, and words in cybersecurity warnings. In: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, 14 pp. ACM, New York (2020). https://doi.org/10.1145/3313831.3376611
5. Seifi, H., Oppermann, M., Bullard, J., MacLean, K.E., Kuchenbecker, K.J.: Capturing experts' mental models to organize a collection of haptic devices: affordances outweigh attributes. In: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems, 13 pp. ACM, New York (2020). https://doi.org/10.1145/3313831.3376395
6. Adamides, G., Christou, G., Katsanos, C., Xenos, M., Hadzilacos, T.: Usability guidelines for the design of robot teleoperation: a taxonomy. IEEE Trans. Hum. Mach. Syst. **45**, 256–262 (2015). https://doi.org/10.1109/THMS.2014.2371048

7. Kappel, K., Tomitsch, M., Költringer, T., Grechenig, T.: Developing user interface guidelines for DVD menus. In: CHI 2006 Extended Abstracts on Human Factors in Computing Systems, pp. 177–182. ACM, New York (2006). https://doi.org/10.1145/1125451.1125490
8. Zaphiris, P., Ghiawadwala, M., Mughal, S.: Age-centered research-based web design guidelines. In: CHI 2005 Extended Abstracts on Human Factors in Computing Systems, pp. 1897–1900. ACM, New York (2005). https://doi.org/10.1145/1056808.1057050
9. Katsanos, C., Tselios, N., Avouris, N.: AutoCardSorter: designing the information architecture of a web site using latent semantic analysis. In: Proceedings of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems, CHI 2008, pp. 875–878. ACM, Florence (2008). https://doi.org/10.1145/1357054.1357192
10. Katsanos, C., Tselios, N., Avouris, N.: Automated semantic elaboration of web site information architecture. Interact. Comput. **20**, 535–544 (2008). https://doi.org/10.1016/j.intcom.2008.08.002
11. Katsanos, C., Tselios, N., Goncalves, J., Juntunen, T., Kostakos, V.: Multipurpose public displays: Can automated grouping of applications and services enhance user experience? Int. J. Hum. Comput. Interact. **30**, 237–249 (2014). https://doi.org/10.1080/10447318.2013.849547
12. Albert, W., Tullis, T.S.: Measuring the User Experience: Collecting, Analyzing, and Presenting Usability Metrics. Morgan Kaufmann (2013)
13. Rosenfeld, L., Morville, P., Arango, J.: Information Architecture: For the Web and Beyond. O'Reilly Media, Sebastopol (2015)
14. Hudson, W.: Card sorting. In: Soegaard, M., Dam, R.F. (eds.) The Encyclopedia of Human-Computer Interaction, 2nd edn. Interaction Design Foundation (2013)
15. Paul, C.: A modified Delphi approach to a new card sorting methodology. J. Usability Stud. **4**, 7–30 (2008)
16. Wood, J., Wood, L.: Card sorting: current practices and beyond. J. Usability Stud. **4**, 1–6 (2008)
17. Tullis, T., Wood, L.: How many users are enough for a card-sorting study? In: Usability Professionals Association (UPA) 2004 Conference, Minneapolis, MN (2004)
18. Nielsen, J.: Card Sorting: How many users to test. http://www.useit.com/alertbox/20040719.html. Accessed 10 Feb 2023
19. Tullis, T., Wood, L.: How can you do a card-sorting study with LOTS of cards? In: Usability Professionals Association (UPA) 2004 Conference, Minneapolis, MN (2004)
20. Katsanos, C., Tselios, N., Avouris, N., Demetriadis, S., Stamelos, I., Angelis, L.: Cross-study reliability of the open card sorting method. In: Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems, pp. LBW2718:1–LBW2718:6. ACM, New York (2019). https://doi.org/10.1145/3290607.3312999
21. Pampoukidou, S., Katsanos, C.: Test-retest reliability of the open card sorting method. In: Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems. pp. Article330:1–Article330:7. Association for Computing Machinery, New York (2021). https://doi.org/10.1145/3411763.3451750
22. Ntouvaleti, M., Katsanos, C.: Validity of the open card sorting method for producing website information structures. In: CHI Conference on Human Factors in Computing Systems Extended Abstracts, pp. Article374:1–Article374:7. Association for Computing Machinery, New York (2022). https://doi.org/10.1145/3491101.3519734
23. Zafeiriou, G., Katsanos, C., Liapis, A.: Effect of sense of direction on open card sorts for websites. In: CHI Greece 2021: 1st International Conference of the ACM Greek SIGCHI Chapter, pp. Article6:1–Article6:8. Association for Computing Machinery, New York (2021). https://doi.org/10.1145/3489410.3489416

24. Katsanos, C., Zafeiriou, G., Liapis, A.: Effect of self-efficacy on open card sorts for websites. In: Proceedings of the 24th International Conference on Human-Computer Interaction, HCI International 2022, pp. 75–87. Springer International Publishing, Gothenburg (2022). https://doi.org/10.1007/978-3-031-06424-1_7
25. Harper, M.E., Jentsch, F., Van Duyne, L.R., Smith-Jentsch, K., Sanchez, A.D.: Computerized card sort training tool: is it comparable to manual card sorting? In: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, pp. 2049–2053. SAGE Publications Inc. (2002). https://doi.org/10.1177/154193120204602512
26. Petrie, H., Power, C., Cairns, P., Seneler, C.: Using card sorts for understanding website information architectures: technological, methodological and cultural issues. In: Campos, P., Graham, N., Jorge, J., Nunes, N., Palanque, P., Winckler, M. (eds.) INTERACT 2011. LNCS, vol. 6949, pp. 309–322. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-23768-3_26
27. Chaparro, B.S., Hinkle, V.D., Riley, S.K.: The usability of computerized card sorting: a comparison of three applications by researchers and end users. J. Usability Stud. **4**, 31–48 (2008)
28. Melissourgos, G., Katsanos, C.: CardSorter: Towards an open source tool for online card sorts. In: Proceedings of the 24th Pan-Hellenic Conference on Informatics, pp. 77–81. ACM, New York (2020). https://doi.org/10.1145/3437120.3437279
29. Paea, S., Katsanos, C., Bulivou, G.: Information architecture: Using k-means clustering and the best merge method for open card sorting data analysis. Interact. Comput. **33**, 670–689 (2021). https://doi.org/10.1093/iwc/iwac022
30. Paea, S., Katsanos, C., Bulivou, G.: Information architecture: using best merge method, category validity, and multidimensional scaling for open card sort data analysis. Int. J. Hum. Comput. Interact. Forthcoming, 1–21 (2022). https://doi.org/10.1080/10447318.2022.2112077
31. Righi, C., et al.: Card sort analysis best practices. J. Usability Stud. **8**, 69–89 (2013)
32. Paul, C.: Analyzing card-sorting data using graph visualization. J. Usability Stud. **9**, 87–104 (2014)
33. Capra, M.G.: Factor analysis of card sort data: an alternative to hierarchical cluster analysis. In: Proceedings of the Human Factors and Ergonomics Society 49th Annual Meeting, pp. 691–695. HFES, Santa Monica (2005)
34. Dong, J., Martin, S., Waldo, P.: A user input and analysis tool for information architecture. In: CHI 2001 extended abstracts on Human factors in computing systems, pp. 23–24. ACM, Seattle (2001). https://doi.org/10.1145/634067.634085
35. Nawaz, A.: A comparison of card-sorting analysis methods. In: Proceedings of the 10th Asia Pacific Conference on Computer-Human Interaction, APCHI 2012, pp. 583–592. ACM Press (2012)
36. Paea, S., Baird, R.: Information Architecture (IA): Using multidimensional scaling (MDS) and K-Means clustering algorithm for analysis of card sorting data. J. Usability Stud. **13**, 138–157 (2018)
37. Usability Body of Knowledge: Card Sorting. https://www.usabilitybok.org/card-sorting. Accessed 21 Oct 2022
38. Whitenton, K.: Tree testing: fast, iterative evaluation of menu labels and categories. https://www.nngroup.com/articles/tree-testing/. Accessed 10 Feb 2023
39. Hanington, B., Martin, B.: Universal Methods of Design, Expanded and Revised: 125 Ways to Research Complex Problems, Develop Innovative Ideas, and Design Effective Solutions. Rockport Publishers, Beverly (2019)
40. Spencer, D.: Card-based classification evaluation. https://boxesandarrows.com/card-based-classification-evaluation/. Accessed 10 Feb 2023

41. Szwillus, G., Hülsmann, A., Mexin, Y., Wawilow, A.: Casolysis 2.0-Flexible auswertung von card sorting experimenten. In: Mensch und Computer 2015–Usability Professionals, pp. 444–455. De Gruyter Oldenbourg (2015)
42. Brooke, J.: SUS: a "quick and dirty" usability scale. In: Jordan, P.W., Thomas, B., Weerdmeester, B.A., and McClelland, A.L. (eds.) Usability Evaluation in Industry. Taylor and Francis, London (1996)
43. Katsanos, C., Tselios, N., Xenos, M.: Perceived usability evaluation of learning management systems: a first step towards standardization of the System Usability Scale in Greek. In: Proceedings of the16th Panhellenic Conference on Informatics (PCI 2012), pp. 302–307 (2012)
44. Orfanou, K., Tselios, N., Katsanos, C.: Perceived usability evaluation of learning management systems: Empirical evaluation of the System Usability Scale. Int. Rev. Res. Open Distrib. Learn. **16**, 227–246 (2015)
45. Field, A.P.: Discovering Statistics Using IBM SPSS Statistics. SAGE, London (2013)