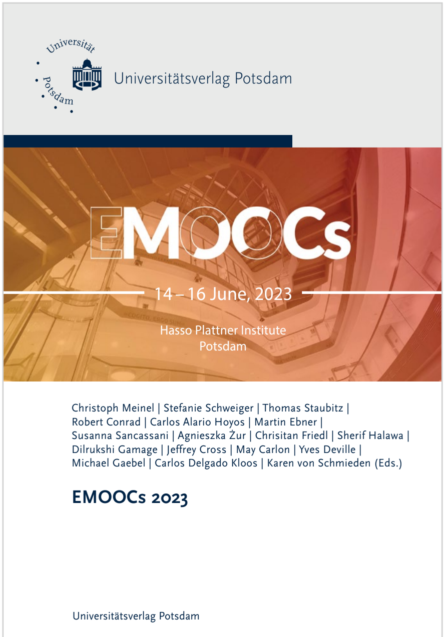
Christoph Meinel, Stefanie Schweiger, Thomas Staubitz, Robert Conrad, Carlos Alario Hoyos, Martin Ebner, Susanna Sancassani, Agnieszka Żur, Christian Friedl, Sherif Halawa, Dilrukshi Gamage, Jeffrey Cross, May Kristine Jonson Carlon, Yves Deville, Michael Gaebel, Carlos Delgado Kloos, Karen von Schmieden (Eds.)

EMOOCs 2023

2023 – vii, 350 p.
DOI https://doi.org/10.25932/publishup-57645



Student-Centered Re-Design of an Online Course with Card Sorting

How to quickly get a mental model of students

Michael Dietz and Dennis Roth

Lehr- und Kompetenzentwicklung – Technische Hochschule Nürnberg
michael.dietz@th-nuernberg.de | dennis.roth@th-nuernberg.de

"How can a course structure be redesigned based on empirical data to enhance the learning effectiveness through a student-centered approach using objective criteria?", was the research question we asked. "Digital Twins for Virtual Commissioning of Production Machines" is a course using several innovative concepts including an in-depth practical part with online experiments, called *virtual labs.* The teaching-learning concept is continuously evaluated. *Card Sorting* is a popular method for designing information architectures (IA), "a practice of effectively organizing, structuring, and labeling the content of a website or application into a structuref that enables efficient navigation" [11]. In the presented higher education context, a so-called *hybrid card sort* was used, in which each participants had to sort 70 cards into seven predefined categories or create new categories themselves. Twelve out of 28 students voluntarily participated in the process and short interviews were conducted after the activity. The analysis of the category mapping creates a quantitative measure of the *(dis-)similarity* of the keywords in specific categories using hierarchical clustering (HCA). The learning designer could then interpret the results to make decisions about the number, labeling and order of sections in the course.

1 Introduction and Motivation

The course is provided for master students in mechanical engineering at Nuremberg University of Applied Sciences since 2019 by the lead author. Further details of the course concept are described in broad terms in [1]. The teaching-learning concept includes an in-depth practical part where students learn about virtual commissioning and how to interact with the digital twin in online experiments, called *virtual labs*. Course material in the learning management system (LMS) *Moodle* accompanies practical exercises with problem-based learning as the teaching method.

section 1 introduces the context of our work. In section 2, we shortly describe the background of the method and provide further sources. In section 3, details of the data collection process will be provided and in section 4 the analysis and interpretation is described comprehensibly and in detail. Results and conclusions will be given in section 5. Finally, we provide suggestions for future work.

1.1 Present Structure of the Course

The challenges with the course are a wide range of topics to be covered, interdisciplinary skills necessary to achieve the learning objectives, and ultimately, mastery of a sophisticated industrial software environment being used. The initial course material in the LMS was split into two parts, with the first part providing basic theoretical knowledge and the second practical part, in which students would use the dedicated software environment and instruction materials to work on their practical exercises independently. After the initial execution, the course was restructured for the first time based on students' feedback. We changed two main sections with subsections to become *weekly* sections. However, *this* first restructuring was not satisfactory, as further student feedback revealed in the next semester.

The course started with an organizational part and progressed week by week through theoretical parts and accompanying practical parts. The theoretical part covers topics such as automation technology, system modeling, programmable logic controllers (PLCs), fieldbuses and network topologies. In addition to the text documents, there are also videos and Internet links to support the students and to serve as an aid to acquire the learning material autodidactically. The practical parts contains many exercises that the students are supposed to work on and usually their work is to be submitted as a written report or assignment based on a template. Also, with the help of so-called mastery tests, students should quickly become familiar with the course structure to find information themselves and upload their assignments in a timely manner. Figure 1 shows an overview for theoretical and practical parts and the order in which they are presented.

2 Background

"Everyone knows the phenomenon: some products can be used intuitively, with others you are constantly searching for the right functions and they never behave as you expect them to... in many cases, however, it is not because someone has not put ***enough thought into them****, but because the information architecture does not correspond to the understanding from the user's point of view"*, as phrased by [7].

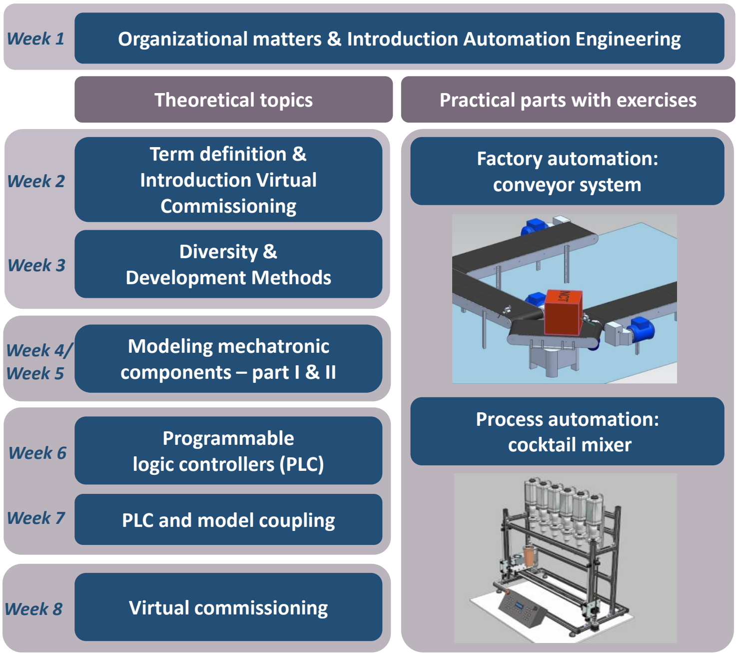


Figure 1: Theoretical content is divided in weeks, practical exercises run in parallel.

Card sorting (CSort) is particularly useful for learning designers creating courses in a LMS, where it is utilized to match designs to users' mental models, as described in detail along with more user-centered design and evaluation techniques in [12]. Since the users themselves are not able to describe their mental model, attempts must be made to obtain the necessary information indirectly as also stated in [7]. *CSort* is for Front-End-Analysis and a user experience (UX) research technique that has been used for several years to determine an information architecture. It was used already used in 1985 to organize menu structures [13] or user interface designs [9]. It involves asking participants to group and categorize information or concepts written on cards. This technique can help researchers to understand how people/users think about and organize information, as well as how they prioritize and label information. To conduct a card sort study, the researcher first creates a set of cards, each containing a single piece of information. These cards can be physical cards or digital cards in a software program. Participants are then asked to group the cards into categories based on their own logic or criteria. Once the cards are sorted, participants may be asked to label the categories they created and explain their thought process. In [3, 11] various card sorting research techniques are explained in detail, among them physical or digital card sorting, closed, open and hybrid and unmoderated and moderated testing. For further method description, please refer to the given literature sources. Figure 2 shows the process based on Mosers's User Experience Design [7].

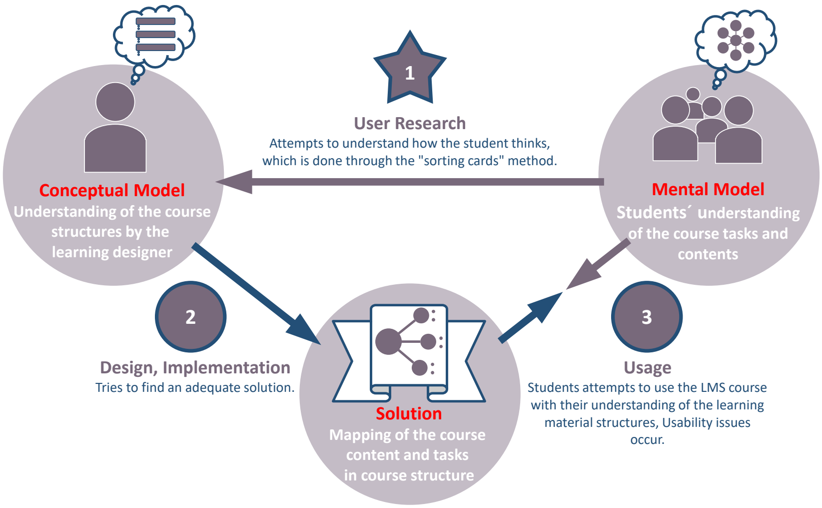


Figure 2: Simplified process in accordance with [7]

3 Methodology

A typical CSort dataset is structured as illustrated schematically in Table 1. Our approach uses several sources, firstly the related literature from section 2 and descriptions on the web, see [2] for a good start. It is suggested for beginners to

Table 1: Example of a nominal dataset with multiple participants in card sorting

Cards	Participant_1	Participant_2	Participant_3	...
Card 1	Category A	Category X	Category A	...
Card 2	Category A	Category X	Category X	...
Card 3	Category B	Category Y	Category Z	...
...	...	...	...	...

dive deeper into the following topics: *Hierarchical cluster analysis* [8] and *Distance methods* [4] and *Linkage methods*. Next, we will mention some critical aspects in brief. For our analysis, we used *complete linkage*, as shown in Figure 4 compared to other methods.

3.1 Performing an Hierarchical Cluster Analysis

We used the software *RStudio* to conduct a Hierarchical Cluster Analysis (HCA) using specific packages and functions which allows the calculation of the distance of nominal factors. The raw dataset has to be formatted to a CSV (comma separated values) file. By importing the dataset each card is represented by a number, which will later be used in the graphical output. Each step is shortly described next and in detail in the appendix on page 348.

A so-called dissimilarity matrix or item-by-item matrix with specified columns of the dataset is generated first, please note the different linkage methods further described in section 4. The dissimilarity matrix is used to execute the HCA. Next, the cluster analysis has to be visualized: a basic dendrogram is generated. A dendrogram is obtained by plotting the results of the HCA, which should further be graphically prepared. In Figure 4 in section 4, we show the resulting dendrograms generated using the different linkage methods.

4 Analysis and Results

This section is based on the recommended best practices for *"interpreting cluster analysis data matrices and dendrograms"* as shown in Righi et al. [11]. However, we only apply some aspects of their work that we believe are essential and helpful to the processing of our dataset.

4.1 Analysis of the item-by-item matrix

During the analysis, a number of data matrices and charts are created automatically or manually, depending on the tool chain used. The first one is the *item-by-item matrix (ii-mx)*, called disimilarity matrix earlier, which helps researchers to quickly find the most important relationships between each keyword in the card set. The *ii-mx* shows a percentage value representing how many participants have grouped each individual card with each other card in the set. One may look at the strongest relationships and ask *"What is the connection between these items?"*

First example: We start with two cards named *VPN connection (No. 8)* and *dongle license (No. 69)*. The dataset shows that 92 % of the participants put both cards in the same category. In fact, there is a close link here, as students need to connect to their institution's virtual private network (*VPN*) to activate their IP-over-USB *dongle license*. Every time students want to work on their exercises, these two steps are necessary. The second example shows two cards allocated to the same category by *all students*, giving a value of 100 %. These are, the so-called *PTn-Element (No. 65)*, a technical term from the field of control engineering and systems theory, and the *non-linear system element (No. 64)*, another term from the same field. As an important note, the underlying calculation process for the ii-mx starts with the highest similarity value and all other cards are calculated on this baseline. The given examples therefore explain the correlation of the items and thus also verifies the correctness of the underlying calculations that should lead to a high degree of similarity as suspected, further interrelations will be described in sections 4 and 5.

Similarity in percentages

Card no.	Labels	WinMOD	State machine	Timing functions, bit linkage, counters, memory functions	PTn-Element	Non-linear system element	VPN connection	Dongle license	Industry 4.0
61	WinMOD	–							
57	State machine	17	–						
55	Timing functions, bit linkage, counters, memory functions	17	58	–					
65	PTn-Element	33	42	42	–				
64	Non-linear system element	33	42	42	100	–			
8	VPN connection	25	8	17	8	8	–		
69	Dongle license	33	17	25	17	17	92	–	
28	Industry 4.0	0	0	0	0	0	0	0	–

Figure 3: Excerpt of an item-by-item matrix. The similarity indicates how many of the participants have sorted both cards into one category. Note that card no. are intended to identify items in the dendrogram in Figure 4

4.2 Analysis of the dendrogram

The *dendrogram*, also *dendrite* or *tree diagram* is "a visual representation of item relationships [and] is similar to that of a tree in that a large branch subdivides into smaller branches, each of which subdivides into still smaller branches, [...]

resulting in a hierarchy of categories and items" as phrased in [11]. As outlined in Kim and Bayes [14], *average*, *single*, and *complete linkage* have specific features such that e.g. *complete linkage* "...can resolve large clusters though it is highly influenced by outliers" or *single linkage* is "...sensitive to outliers but impervious to differences in the density of clusters". A first sight of the item-by-item and dendrogram representation is helpful for the process of creating the top-level categories for the redesigned course structure. Righi et al. refer to a slider function generated by card sorting tools for defining boundary lines to create categories.

In our work, we created the cut-off line manually, initially obtaining five top-level categories, see dendrogramm at far right and imagine the boundary line at roughly 90 %. Five categories are also highly consistent with the *"magical number seven plus or minus two"* by Miller [6]. Righi et al. now recommend returning to the ii-mx representation after defining the top-level categories and check that at least half to two-thirds of the rows have a high correlation with your categories, if this is true, one can proceed to the next step of our analysis. We used a potential top-level category *automation pyramid (AP)* that could best represent the card items. An AP depicts different levels in industrial production and classifies systems in control engineering. Unfortunately, there is no consensus on the naming and number of levels, so there are about 25 variants in the literature, as described in Meudt et al. [5] and it is challenging for students (and for researchers) which topics are related to the AP. Further, cards like *Cyber-physical systems (CPS)* or *hierarchical structure of productions systems (HSP)* where allocated in the same category. Table 2 does not show all percentage values, as it is intended to be understood only to illustrate the returning step to the ii-mx in the analysis. It was found that less than half of the similarity values were with high agreement (set to 50 %), Table 2 below. In that case Righi et al. state that participants may have very different mental models of how the content should be organized.

Table 2: ii-mx cells table representation indicating relationships

Card items	Field	Control	Process control	...	CPS	HSP
Field	–	50 %	67 %	...	25 %	17 %
Control	50%	–	42 %	...	25 %	8 %
Process control	67%	42 %	–	...	25 %	25 %
...	...	...	...	–	...	...
CPS	25 %	25 %	25 %	...	–	25 %
HSP	17 %	8 %	25 %	...	25 %	–

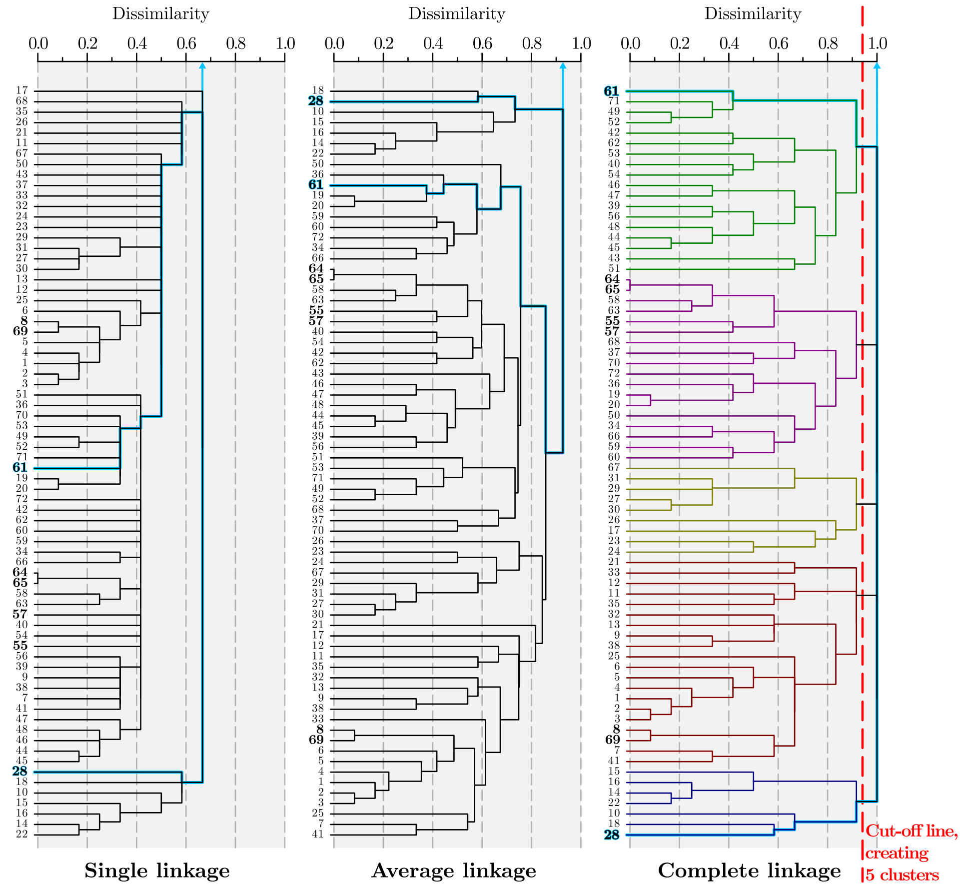


Figure 4: Dendrograms with different linkage methods, boldface card numbers correspond to cards from Figure 3. Depending on the method, the representation of the of the dissimilarity of two cards may be different, as shown in cards no. 28 and 61. Clusters are produced where the cut-off line intersects the dendrogram.

4.3 Label the Top-Level Categories

So far, it was decided to have *five* potential top-level categories, yet the names of the categories are still an open issue. This seems to be an extremely critical step, as it clearly affects the results and subjective naming can lead to non-optimal structure or usability of the course. To avoid this, we again followed Righi et al. [11], namely *Approach 1: Review the Items in Categories*. We started with seven predefined categories: Control Engineering, Automation Engineering, Modeling, Digital Factory, Digitization, Industry 4.0 and Other, working directly on raw data and focusing to:

1. Items that some participants grouped together but others kept separate
2. Cases where participants took different strategies for grouping items
3. *Miscellaneous* categories contain items participants didn't know where to put

Each case enforces specific actions for category labeling. Next, a selection of items is shown as an example how to find and label *suitable* top-level categories.

Case [1] will be demonstrated using items such as *Cyber-Physical System, Process control and Industry 4.0*. A simple text search in several tables of the raw data was used for this step. It could be found that, participants assigned these items often to different categories like *Other, Digitization or Digital Factory*. Yet, the categories ***Digital Factory and Industry 4.0*** were used more frequently. It was therefore decided to combine these two category names together for a single top-level category.

Case [2] focuses on cards about programming. Several cards were available for grouping and participants showed different strategies. One strategy was to combine cards related to the industrial software used in the course, namely for programming a PLC and for graphical programming in the software WinMOD. But eventually the idea came up to create a superior category where all other category assignments related to programming seem to fit in. A participant named it *Programs/Languages/Properties* and a simple internet search for wording options resulted then in the top-level category ***Types and Features of Programming Languages***, which covers to a large extent the required mastery of textual *and* graphical languages. For case [3] the search was for items that were assigned to the category *Other*. Items therein were e.g. *learning goals of the course, creating a screencast, write reports and use technical terms accurately, deepen understanding of technical terms through technical articles, definition of terms, use mindmap method*. Since these elements are clearly practice-based exercises, a category ***learning objectives*** to include all in one place was created. Lastly, brief interviews were conducted with participants after the CSort activity. Each was asked to describe considerations for category assignment and their individual course category sequence. Comments were also made on card items, such as cards that did not match well because they would fit into any or no category at all. Some suggested to divide the content between *physical world – the real production plant* from the *virtual world – the digital twin*. Lastly participants suggested combining *Control* and *Automation Engineering* or *Digital Factory* and *Industry 4.0* and *Digitization* categories together, which we also assumed when looking at the data as in case [2].

5 Discussion and Conclusion

This publication provided insights into an example of redesigning an online course based on empirical learner data. Our work shows how the method *Card Sort* can be used to create a mental model of students' information architecture to find meaningful clusters for learning content. The aim, generally speaking, is to derive a set of divisions and subdivisions of elements that lead to a reasonable information architecture. In addition, several valuable lessons were learned during the process, noting that the same result can be achieved by surveying students, but also that the use of data can contribute to objective structuring. A student who participated both in the course and card sort stated that *"the ability to create your own categories is extremely handy, as students can present their own opinions/contexts even better."* Regarding repeatability restrictions it is stated in [10] *". . . two card sort trials performed by the same participants. . . "* showed that the test-retest reliability was between 81 % and 95 % and that the open card sorting method has *"high test-retest reliability"*. These findings will be used for future work where further analysis is planned to improve the current results, to adjust for any inaccuracies, and to find and evaluate appropriate subcategories. This research is funded as part of the program "Strengthening University Teaching through Digitization" via the "Stiftung Innovation in der Hochschullehre" of the German federal and state governments [FBM2020-EA-2700-07250].

Appendix

First install the packages `cluster` for the `daisy()` function with the command in the software *RStudio.*

```
install.package('cluster')
```

To activate the package use `library()`:

```
library(cluster)
```

Importing the CSV file is done by the `read.csv()` function. With the option `stringsAsFactors=TRUE` we are able to use nominal data for the cluster analysis.

```
data <- read.csv('C:/My Documents/card-sorting.csv',
stringsAsFactors=TRUE)
```

By importing the dataset each card is represented by a number, which will later be used in the graphical output. The `daisy()` function is used to generate the dissimilarity matrix from the specified columns of the dataset, which we call `dm`.

```
dm <- daisy(data[,c('Participant_1','Participant_2',
'Participant_3', ... )])
```

The dissimilarity matrix `dm` is now used to execute the HCA with the function hc, which we defined as `hca`. Which linkage method is to be used in the cluster analysis is specified with the `method ='*'` option.

```
hca <- hclust(dm, method = 'average')
```

A basic dendrogram is generated from `hca` via the `plot()` function. With the option `hang` the vertical position, with `cex` the font size of the labels can be changed.

```
plot(hca, hang = -1, cex = 0.6)
```

By plotting `hca`, a dendrogram is obtained, which should further be graphically prepared.

References

[1] M. Dietz, A. Abebe, J. Lederer, T. Michl, and R. Schmidt-Vollus. “Case Study of a Virtual Lab Environment Using Virtualization Technologies and a Desktop as a Service Model”. In: Springer International Publishing, 2022. DOI: 10.1007/978-3-030-96296-8_72.

[2] A. Evans. *Remote Unmoderated Card Sorts: For Free!* URL: https://uxpamagazine.org/remote-unmoderated-card-sorts-for-free/ (last accessed 2023-03-22).

[3] E. Greifeneder and P. Bressel. “Hybrid Digital Card Sorting: New Research Technique or Mere Variant?” In: *Information for a Better World: Shaping the Global Future*. Edited by M. Smits. Cham: Springer International Publishing, 2022, pages 50–67. ISBN: 978-3-030-96960-8.

[4] A. Janusz. “Algorithms for Similarity Relation Learning from High Dimensional Data”. In: *Transactions on Rough Sets XVII*. Edited by J. F. Peters and A. Skowron. Berlin and Heidelberg: Springer, 2014, pages 174–292. ISBN: 978-3-642-54756-0. DOI: 10.1007/978-3-642-54756-0_7.

[5] T. Meudt, M. Pohl, and J. Metternich. *Die Automatisierungspyramide – Ein Literaturüberblick*. Technical report. Darmstadt, June 2017.

[6] G. A. Miller. “The magical number seven plus or minus two: some limits on our capacity for processing information.” In: *Psychological review* 63.2 (1956), pages 81–97.

[7] C. Moser. *User Experience Design. Mit erlebniszentrierter Softwareentwicklung zu Produkten, die begeistern*. Berlin and Heidelberg: Springer, 2012. ISBN: 978-3-642-13362-6.

[8] F. Nielsen. “Hierarchical Clustering”. In: Feb. 2016, pages 195–211. ISBN: 978-3-319-21902-8. DOI: 10.1007/978-3-319-21903-5_8.

[9] J. Nielsen and D. Sano. “SunWeb: User interface design for Sun Microsystem’s internal Web”. In: *Computer Networks and ISDN Systems* 28 (Dec. 1995), pages 179–188. DOI: 10.1016/0169-7552(95)00109-7.

[10] S. Pampoukidou and C. Katsanos. “Test-Retest Reliability of the Open Card Sorting Method”. In: *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*. CHI EA ’21. Yokohama, Japan: Association for Computing Machinery, 2021. ISBN: 9781450380959. DOI: 10.1145/3411763.3451750.

[11] C. Righi, J. James, M. Beasley, D. L. Day, J. E. Fox, J. Gieber, C. Howe, and L. Ruby. “Card Sort Analysis Best Practices”. In: *Journal of User Experience* (2013), pages 69–89.

[12] M. Schmidt, Y. Earnshaw, A. Tawfik, and I. Jahnke. “Methods of User Centered Design and Evaluation for Learning Designers”. In: Edtech Books, Oct. 2020.

[13] T. S. Tullis. “Designing a Menu-Based Interface to an Operating System”. In: *SIGCHI Bull.* 16.4 (Apr. 1985), pages 79–84. ISSN: 0736-6906. DOI: 10.1145/1165385.317471.

[14] O. Yim and K. Ack Baraly. “Hierarchical Cluster Analysis: Comparison of Three Linkage Measures and Application to Psychological Data”. In: *The Quantitative Methods for Psychology* 11 (Feb. 2015), pages 8–21. DOI: 10.20982/tqmp.11.1.p008.