Exploring the use of Concept Mapping to facilitate Interaction Design Processes

Ilya Shmorgun
Center for Educational Technology
Tallinn University
Narva mnt 25, 10120 Tallinn, Estonia
ilya.shmorgun@gmail.com

David Lamas
Center for Educational Technology
Tallinn University
Narva mnt 25, 10120 Tallinn, Estonia
david.lamas@tlu.ee

ABSTRACT

This work focuses on the field of interaction design in the context of software development. The research question focuses on assessing whether concept mapping is an adequate technique to support interaction design processes. The results indicate that concept mapping can indeed be applied in the field of interaction design and can successfully deliver on the established requirements.

Keywords

Concept mapping, diagramming techniques, interaction design

INTRODUCTION

The need to create usable and understandable products can be seen as inevitably leading to a necessity of utilizing an established process with concrete principles and methodologies. Such a process is interaction design [1].

Existing techniques to support the design process

In his article, [2] states that for a time there has been a "search for a single holy Grail of design techniques" to support interaction design processes.

According to [3], a common problem in interaction design is that the concepts, tasks and terminology of a product environment need to be established. The multidisciplinary teams, participating in interaction design processes, quite often lack a common language to communicate product development knowledge to each other. Knowledge extraction and understanding of technological constraints is also very difficult.

In [4] it is stated that scenarios and prototypes are commonly used techniques for supporting communication yet quite often projects lack an integrated view, which may lead to issues and a general shortage of focus.

In [2] it is mentioned the use of conceptual specifications to facilitate agreement of team members in regards to the system, which is being developed. State charts can also be used to produce representations of interfaces after a designer has created them. Cognitive walkthroughs can be used to evaluate features of an interface. A User Action Notion can be utilized to capture behavioral information.

At the time of this writing, the most common techniques used for supporting interaction design processes are also UML (unified modeling language), state transition and decomposition diagrams [4, 5].

Still [2] remarks that at one point the methods of software engineering were sufficient, but the introduction of cognitive psychology to the field of interaction design lead to a requirement in using many techniques to support the interaction design process.

And finally [4] summarizes the issue by stating, that many models and representations have been proposed to support design processes, however these deal only with specific fragments of the designed product. This can lead, for example, to difficulties in communicating design decisions.

Problem statement

A problem can be seen in the absence of a technique, which can be used to support interaction design processes through all of their stages. Such a technique should be easy to understand yet flexible and powerful enough to be adapted to specific context of a particular design stage. It is important for the technique not to be limited by any particular field.

Research question

A research question is put forth whether there exists such a technique, which could be used to support interaction design processes through all of their stages. Although [3] presents an answer in the form of pattern languages, which turned out to be one of the most commonly proposed techniques, and [6] suggest the use of MoLIC (Modeling Language for Interaction as Conversation), the fact that, as of the time of this writing, none of these techniques have been widely adopted by the interaction design community leads to a conclusion, that the formulated research question has not been answered by other researchers.

Research methodology

Design research has been chose as the methodology for validating the hypothesis with the goal of improving the way the proposed design would operate in practice and producing a contribution of knowledge. Several case studies were conducted, which focused on designing various prototypes of mobile applications. Concept mapping has been applied in each of those studies with the goal of researching its effectiveness as a form of support of interaction design processes. The technique was continuously refined from case study to case study in order to revise the design based on experience and work out all possible issues. Real-life projects were used as case studies in order to avoid distortions of resulting findings. At times,

certain elements of the design were not working and needed to be modified to better fit the context.

CONCEPT MAPPING

A concept map is essentially a diagram, which enables the representation of relationships between ideas, images and words [7, 8, 9, 10]. Concept mapping encourages logical thinking by illustrating connections and helping visualize how individual ideas form a larger whole. In [11] it is stated that a concept map is a hierarchical network, which is comprised of concept terms. These terms also act as nodes, which are connected by labeled linking lines. The labels of the links explain relationships between nodes. Concept maps can be used to organize pieces of wideware [12]. An example [10] puts forward is of teams using Post-its to create concept maps and later transferring them into a computer.

Structuring a concept map in a hierarchical way means dividing concepts into two large groups. One of them is the higher order group, which consists of concepts that are more general and the lower order group, which is usually populated with concepts that are more specific. It is also a possible to add cross-links, which show relationships between ideas in different segments of the diagram. Thus, certain concepts can form loops and not just straightforward linear connections. An exception might be a more hierarchical concept map with very little or no cross-links at all [11].

An important value of concept maps is the ability to add a focus question, which helps the author use only the necessary ideas in order for the diagram to answer a formulated question [11]. And also according to [9], concept maps are flexible enough to show relationships among concepts of any type of a complex and structured domain. The maps can be built for any type of subject and can facilitate creative work in every discipline.

A way of structuring knowledge

The goal of concept mapping is to facilitate meaningful learning, which is a process characterized by a learner being in control of the acquired new knowledge. The person is then able to retain the knowledge longer and use it in creative thinking and as scaffolding in future learning.

Concept maps help establish an integrated framework of propositions and concepts, which are organized hierarchically to support problem solving.

Because creating a concept map for a particular domain of knowledge requires a person to structure and organize his ideas in a comprehensive and thorough way, concept maps can also easily surface both valid and invalid ideas. As a result, the creator of a concept map is able to see clearly, which ideas do not simply make sense and need reformulating or expanding [10].

Facilitating teamwork

Novak's research suggests that concept mapping is a great tool to be used in teams. Concept maps help groups move towards solutions that are more creative. Different members are able to share concept meanings for words and symbols and reach an agreement on how the information depicted in the diagram relates to other concepts from a particular domain of knowledge in regards to a certain question of interest to the group.

With the help of concept maps team members are able to reach a consensus and capture ideas collectively, thus reaching a new level of understanding and cooperation [19]. This can be especially important in interaction design processes, were there is a need to clearly communicate ideas between stakeholders and members of the design and development team.

Concept mapping as a visualization tool

An important aspect of concept maps is that they enable visualization by creating visual artifacts, which can amplify cognition and reflect the process of internal construction in the mind. During the creation of a visual map of abstract data, the mind establishes certain mental models, helping a person to understand and expand that data. Through the benefit of visualization, it is possible to produce better information [13].

Concept mapping in interaction design

Humans live in a world of information. We can argue that the ability to learn, communicate and think or essentially to cognize is the main characteristic, which makes us human. However, the constraints of our physical brain do not necessarily limit cognition because we are gifted with the ability to offload our cognition onto the external environment, effectively modifying it in the process. This offloading can take several forms and occur at different times and during different processes. As a result, various tools and outlets are used and, quite often, disjoined pieces or scraps of information are created. These scraps need to be flexible, mobile and easily transferable because at some point we might decide to reuse and structure them in a completely different context. Of course, as thinkers and scholars have known for ages, nothing in the world is separate and everything is connected in one way or the other. The difficult part is to identify that connection.

This is where concept mapping can help. The method can support establishing relationships between pieces of data and help us see connections, which are not that obvious at a first glance. The process of creating a concept map also enables us to create a hierarchical and thorough structure in our mind, which we can later use as a springboard for new and exciting ideas to emerge. All of this leads to innovative problem solving, and that is what human beings are best at doing. Being a variation of a graph, concept maps have certain drawbacks in that they are not very scalable. At a certain point, a concept map can become too complex and not readable. Too many nodes and relationships can be introduced in the creation process and as a result, it is not possible to visualize the concept map in a clear and understandable way [13].

And [13] also proposes several solutions to the problem of visualizing graphs and these methods can be applied to

concept maps as well. A good starting point may be the reduction of the diagram. This can be achieved for example by hiding certain relationships, which are of lesser interest to the person using the concept map.

A second option is to make use of a new layout or redistribution of the existing concept map. Finally, interactivity can be introduced, where only the points of interest to the user are represented. This would enable the creation of a dynamic concept map, which would encourage manipulation and exploration according to the needs of the user.

DOCUMENTATION TECHNIQUES IN INTERACTION DESIGN

According to [14] functional requirements of a system have been traditionally analyzed and documented using such tools as data-flow diagrams, state and work-flow charts and others. It is also possible to use state and sequence diagrams. In [5] it is further mentioned that tools such as state charts and flow diagrams can be used for dialogue design, which is similar to a script of a play in interaction design. Finally [15] mentions transition diagrams as having wide applicability in user-interface design. Techniques such as user action notation are applied for describing user behavior and certain aspects of system responses.

Decomposition diagrams

According to [16], decomposition diagrams can be used to create various components, which can be later developed separately. Such diagrams also enable the author to divide the higher-level view of a system into smaller and more manageable blocks. With the help of decomposition diagrams one is able to manage various abstractions in a more productive way as well as the complexity in design of software.

State transition diagrams

As stated in [17, 18], a transition diagram can specify the dynamic behavior of individual objects by representing a sequence of operations in classes. State diagrams can be used for modeling of control and sequencing views of a particular system. Further [19] report that transition diagrams can be used in interface design to "quantify and control surface level complexity." It is also possible to assess if the interface possesses any states, which are unstable or undefined. According to [18] it is also possible to identify states, which are missing or unnecessary. Test cases can be generated, if a transition diagram is transformed into an extended finite state machine.

Unified modeling language

In [21] state that the unified modeling language or UML has become the "de-facto standard for object-oriented analysis and design." It provides a common language, which helps to specify, visualize and document software intensive systems. It is important to note, that UML enables interoperability of tools at the semantic level. UML can provide a framework of notations, which enables the integration of modeling of user interfaces with software engineering. Although native UML components do not have built-in support mechanisms for UI modeling, due to inherent extensibility certain additions might be introduced to address that specific issue [22].

Techniques for software development

In [23], the author's state that, although software developers have applied certain techniques consistently, they tend to be very poor when applied in the field of human-computer interaction in general and interaction design in particular for studying who the users are, what they need and how to create suitable interfaces for them

After reviewing several sources [5, 14, 15, 23] covering the foundations of human-computer interaction and interaction design an assumption can be made that at the time of writing this thesis wide-ranging documentation techniques, which could be used throughout the process of interaction design, have not been yet established. Tools, such as UML, decomposition and state transition diagrams originate from software engineering and do not serve very well in the context of interaction design.

Concept mapping revisited

The research and examples described in the previous sections of this document aim to establish the current state of things in the field of cognitive science and interaction design. We are able to see the way humans gather and structure information from various bits and pieces and how it is crucial to establish a solid foundation in order to successfully build new knowledge as well as facilitate creative thinking and problem solving. We have also seen that the techniques, used for creating those diagrams in the field of interaction design originate from the software engineering discipline and thus might be applicable only in specific contexts of the interaction design process. This brings us to a conclusion that there is a need to look someplace else to discover a technique, which could be used to support interaction design processes through their various stages.

We have seen that such a potential technique does indeed exist, originating from the field of learning and education, being created by a team under the leadership of Joseph Novak. This technique is concept mapping and it has been applied and refined in various fields since the 1970s. Novak's research shows that concept mapping has been successfully applied by both schoolchildren and researchers and scholars with promising results and that the basics can be understood by both a child, attending the first grades at school as well as a scientist trying to figure out a new solution to a complex problem.

The undertaken case studies illustrate that concept mapping can indeed be applied in the field of interaction design having benefits such as: Establishing a common language between different members of the team; Bridging the communications gap between people with different backgrounds, who are not necessarily familiar with tools being used in software engineering; Being able to replicate the cognitive structure, which is established in the human mind during the process of gathering and ordering

information during research and problem solving; and Being flexible and powerful yet easy to use and understand.

Also, once a solid theoretical foundation is established it is possible to move on to the next stages, such as creating sketches, mockups and prototypes, testing and modifying them according to user feedback, writing code and eventually shipping the product. Yet the core technique always stays in place, being refined depending on the requirements and context of a particular stage. This approach ensures that all the subsequent activities are based on a clear and well-defined understanding.

FINAL REMARKS

Future developments and research directions could include additional research in to the potential of concept mapping in order to understand whether certain modifications to the technique need to be made in order for the tool to be more successfully applied in particular contexts. Another interesting opportunity would be to explore edge-cases where concept mapping cannot simply be utilized and to identify which alternative techniques could be employed instead.

REFERENCES

1. Norman, D. (2002). Design of Everyday Things: Basic Books.

2. Karat, J., & Dayton, T. (1990). Taking design seriously: exploring techniques useful in HCI design. SIGCHI Bull., 22(2), 26-33.

3. Borchers, J. O. (2000). A pattern approach to interaction design. Proceedings of the 3rd conference on Designing interactive systems.

4. Paula, M. G. d., Diniz, S., & Barbosa, J. (2004). Using an Interaction Model to Support Communication among HCI Design Team Members from Multidisciplinary Background. VI Simposio sobre Fatores Humanos em Sistemas Cumputacionais, Curitiba.

5. Dix, A., Cairncross, S., Cockton, G., Beale, R., Amant, R. S., & Hause, M. (1993). Human-Computer Interaction: Prentice Hall International UK Limited.

6. Paula, M. G. D., Barbosa, S. D. J., José, C., Lucena, P., Marquês, R., & Vicente, S. (2003). Relating Human-Computer Interaction and Software Engineering Concerns: Towards Extending UML Through an Interaction Modeling Language.

7. Bernstein, M. S., Kleek, M. V., schraefel, m. c., & Karger, D. R. (2007). Management of personal information scraps. Paper presented at the CHI '07 extended abstracts on Human factors in computing systems.

8. Bernstein, M., Kleek, M. V., Karger, D., & Schraefel, M. C. (2008). Information scraps: How and why information eludes our personal information management tools. ACM Trans. Inf. Syst., 26(4), 1-46.

9. Novak, J. D., & Cañas, A. J. (2006). The Theory Underlying Concept Maps and How to Construct and Use Them: Florida Institute for Human and Machine Cognition.

10. Novak, J. D. (2010b). Learning, Creating, and Using Knowledge: Concept maps as facilitative tools in schools and corporations. Journal of e-Learning and Knowledge Society, 6(3).

11. Novak, J. D. (2010a). Learning, Creating, and Using Knowledge: Concept Maps as Facilitative Tools in Schools and Corporations (2 ed.): T & F Books US.

12. Clark, A. (1998). Where Brain, Body and World Collide. Daedalus : Journal Of The American Academy Of Arts And Sciences (Special Issue on The Brain), 127(2), 257-280.

13. Mazza, R. (2009). Introduction to Information Visualization (1st ed.). London: Springer-Verlag.

14. Preece, J., Rogers, Y., & Sharp, H. (2007). Interaction Design: Beyond Human-Computer Interaction (2 ed.): Wiley.

15. Shneiderman, B. (1998). Designing the User Interface: Strategies for Effective Human-Computer Interaction (3 ed.): Addison-Wesley.

16. Chang, H.-F., & Lu, S. C.-Y. (2009). Decomposition and Traceability in Software Design. Proceedings of the 2009 33rd Annual IEEE International Computer Software and Applications Conference - Volume 02.

17. Wasserman, A. I. (1985). Extending State Transition Diagrams for the Specification of Human–Computer Interaction. IEEE Transactions on Software Engineering, SE-11(8), 699 - 713.

18. Bae, J. H., & Chae, H. S. (2009). An Automatic Approach to Generating State Diagram from Contract-Based Class. 16th IEEE International Conference and Workshop on the Engineering of Computer Based Systems.

19. Maple, C., French, T., & Conrad, M. (2003). A visual formalism for graphical user interfaces based on state transition diagrams. Seventh International Conference on Information Visualization.

20. Nunes, N. (2003). Representing User-Interface Patterns in UML Object-Oriented Information Systems (Vol. 2817, pp. 142-151): Springer Berlin / Heidelberg.

21. Jardim Nunes, N., & Falcão Cunha, J. (2000). Towards a UML Profile for Interaction Design: The Wisdom Approach «UML» 2000 — The Unified Modeling Language (Vol. 1939, pp. 101-116): Springer.

22. Kovacevic, S. (1999). UML and User Interface Modeling. Selected papers from the First International Workshop on The Unified Modeling Language: Beyond the Notation.

23. Shneiderman, B., Plaisant, C., Cohen, M., & Jacobs, S. (2010). Designing the user interface: Strategies for effective human-computer interaction (5 ed.): Addison-Wesley.