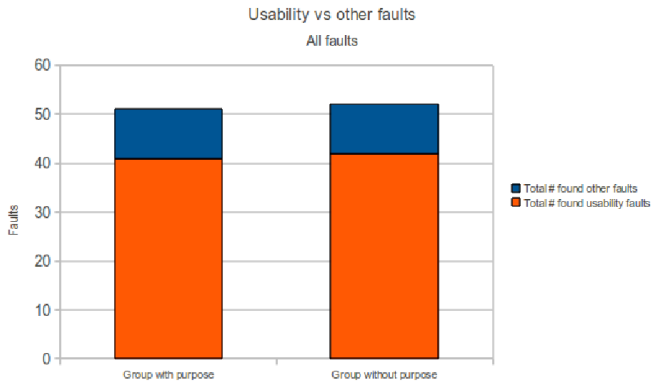

Figure 3: Number of found usability faults versus other faults

This high cognitive load can be caused by the combination of the instructions, the evaluation of the prototype and thinking aloud, the split-attention effect, the absence of extensive domain specific knowledge or the absence of extensive experience with prototyping [7, 12].

If the cognitive load exceeds the working memory resources, it is fatal to learning [7, 12]. This could explain the observations made during the prototyping sessions: most participants didn't really use the provided information and in case of frequently encountered problems schema automation occurred [12].

Threats to validity

The number of participants in this experiment was low, fourteen in total. This number is too low to make any well-grounded conclusions about the results.

The participants consisted of a specialized group, namely undergraduate students of software engineering. There is a chance that they look differently at software and user interfaces then users with a non-IT background.

Final words

In this article we investigated the effect of sharing the purpose of a prototyping session on the success of the prototyping session. Contrary to what we predicted sharing the purpose upfront did not lead to more identified faults concerning the purpose, nor did it lead to fewer identified faults of a different type. However, it is not clear how much other factors, in particular the cognitive load, have influenced the results, thus there is insufficient evidence to support or refute our hypotheses.

Future study should be directed to the influence of cognitive load when sharing the purpose.

References

[1] Hugh Beyer and Karen Holtzblatt. *Contextual design: defining customer-centered systems*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1998.

[2] Ellen Gottesdiener. *Requirements by collaboration: workshops for defining needs*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2002.

[3] Deborah Hix and H. Rex Hartson. *Developing user interfaces: ensuring usability through product & process*. John Wiley & Sons, Inc., New York, NY, USA, 1993.

[4] S. Houde and C. Hill. What do prototypes prototype? *Handbook of human-computer interaction*, 2:367–381, 1997.

[5] IEEE Computer Society. *Software Engineering Body of Knowledge (SWEBOK)*. Angela Burgess, EUA, 2004.

[6] Jeff Johnson. *GUI Bloopers 2.0: Common User Interface Design Don'ts and Dos*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2nd edition, 2007.

[7] F. Paas, A. Renkl, and J. Sweller. Cognitive load theory and instructional design: Recent developments. *Educational Psychologist*, 38(1):1–4, 2003.

[8] J. Radatz. IEEE standard glossary of software engineering terminology. *IEEE Std 610121990*, 121990, 1990.

[9] M. Rettig. Prototyping for tiny fingers. *Communications of the ACM*, 37(4):21–27, 1994.

[10] R.L. Rosnow and R. Rosenthal. *Beginning behavioral research: A conceptual primer*. Prentice Hall Upper Saddle River, NJ, 2005.

[11] R. Sefelin, M. Tscheligi, and V. Giller. Paper prototyping-what is it good for? : a comparison of paper-and computer-based low-fidelity prototyping. In *CHI'03 extended abstracts on Human factors in computing systems*, pages 778–779. ACM, 2003.

[12] J. Sweller, J.J.G. Van Merrienboer, and F.G.W.C. Paas. Cognitive architecture and instructional design. *Educational psychology review*, 10(3):251–296, 1998.

[13] A. Timmerman. Prototyping, a stepping-stone to more successful prototyping. 2008.

[14] Martijn van Welie. Interaction design pattern library. `http://www.welie.com/patterns/index.php`. Retrieved November 21, 2010.

[15] D. Verhamme. Effective prototyping in embedded systems. 2009.

[16] T.Z. Warfel. *Prototyping: A Practitioner's Guide*. Rosenfeld Media, 2009.

[17] I.Q. Whishaw and B. Kolb. *Fundamentals of human neuropsychology*. Worth Publishers, 1995.