

# Designing through testing: a tool for finding usable mobile services

John Sören Pettersson, Malin Wik  
*Centre for HumanIT, Karlstad University, Karlstad, Sweden*  
[johnpett@kau.se](mailto:johnpett@kau.se), [malin.wik@kau.se](mailto:malin.wik@kau.se)

Being able to demonstrate solutions for non-technical prospective users has great advantages compared to any other method, at least if the demonstration technique itself is not too costly. By interactive demonstrations prospective users will get a grasp of the purported system. It is also a way for system developers to see if the whole concept is flawed or if individual steps in the interaction are causing problems.

Such a demonstration technique is incorporated in the system called Ozlab. For many years, the Ozlab system was used by Karlstad University researchers to explore user interface concepts and paradigms for computer applications. The idea was that of *Wizard of Oz*, i.e., one puts in graphics and text in advance but retains control over how things appear on test subjects' screen and loudspeakers during each test/demo session. The test/demo leader manipulates the computers responses. As operating systems have changed through the years, our tool needed updating. Ozlab is now redeveloped, making it possible to run in web browsers – that is, making a cloud solution for Wizard of Oz-based development of interaction design.

For running a test or demo via a web browser, the control buttons of the web browser will have to be removed. This is easily done in many browsers. Left is only the window frame so that whatever content the prototype contains it will appear as an ordinary GUI window with only the prototype's content displayed and not any of the web browser used to connect to the Ozlab server. For the test leader (the 'wizard' secretly manipulating the test subject's screen) it is of course not urgent to hide the web browser. Likewise, when building a prototype – an 'interaction shell' in the Ozlab terminology – one is using an ordinary web browser. Any graphics of the interaction shell has to be made to fit the device where the demo or test is to be run; this to ensure that the prototype is made quickly and that due consideration is taken to design for the device where the idea is going to be tested or demonstrated. Making a responsive-design solution already in the rapid-prototyping tool would belie the

purpose of testing the design for a particular device (many alternatives might be tested, so it should not be calculated from a prototypical prototype).

Noteworthy, if the server is installed to be accessible by Wi-Fi, wizard and test subjects could be in whatever location. The test subject's device might be a mobile phone with web browser. In addition, running Ozlab locally on a laptop, broadcasting by Wi-Fi to a mobile phone, enable experiments in the field where no internet connection is available. Thus, a mock-up of a prospective mobile application (or mobile-adapted web site) can be run interactively regardless of the present status of mobile connections.

With our previous experience in rapid prototyping with Ozlab we know that this tool will speed up user-oriented development considerably. When demonstrating and testing this way, not only the interaction design is possible to develop (turn-taking, graphics, menu structure, evocative sounds, etc.). Rather, because this is such a concrete demonstration, participants normally come up with comments on the service as such, both on the content and the purpose of it. That is, "testing" can start long before functional requirements are settled.

**For updates**, see [www.kau.se/en/ozlab](http://www.kau.se/en/ozlab); **for old works**, see e.g.

Molin, L. (2004) "Wizard-of-Oz Prototyping for Cooperative Interaction Design of Graphical User Interfaces", in *Proceedings of the Third Nordic Conference on Human-Computer Interaction*, 23-27 October, Tampere, Finland, pp. 425-428.

Nilsson, J & J. Siponen (2006) Challenging the HCI concept of fidelity by positioning Ozlab prototypes. Presented at ISD'2005, Karlstad. Published in *Advances in Information Systems Development*, eds. A.G. Nilsson et al., Springer-Verlag, 2006.

Pettersson, J.S. & J. Nilsson (2011) Effects on software quality of early user-testing (testing before programming). Data from a case study. *Information Systems Development. Asian Experiences*. Springer, New York. Pp 499-510.

Thorén, C., J.S. Pettersson & S. Fischer-Hübner (2003) Making Privacy Protocols Usable for Mobile Internet Environments. Presented at HCI International 2003, June 23-25, Crete Greece, *Human-Centered Computing, Vol. 3*, eds. Harris & al., pp. 839-843.

See also <http://www.kau.se/en/ozlab/Publications>

### **Early Wizard of OZ:**

Kelley, J.F. (1983) An empirical methodology for writing user-friendly natural language computer applications. *Proc. of the SIGCHI Conference on Human Factors in Computing Systems*, 1983, pp. 193-196.

### **For related works within this field:**

Alce, G., K. Hermodsson & M. Wallergård (2013) WozARd: A Wizard of Oz Tool for Mobile AR. (Poster.) *Mobile HCI 2013*, 27-30 August, Munich, Germany, pp. 600-605.

Ardito, C., P. Buono, M.F. Costabile, R. Lanzilotti & A. Piccinno (2009) A tool for Wizard of Oz studies of multimodal mobile systems. [About MuMoWOz.] *IEEE HIS 2009*, 21-23 May, Catania, Italy, pp. 344-347.

Grill, T., O. Polacek & M. Tscheligi (2012) ConWIZ: A tool supporting contextual Wizard of Oz simulation. *MUM'12*, 4-6 December, Ulm, Germany, Article no. 21.

Linell, N., R. Bareiss & K. Pantic (2012) A Wizard of Oz tool for Android. *Mobile HCP12*, 21-24 September, San Francisco, USA, pp. 65-70.

Seguro, V.C.V.B. & S.D.J. Barbosa (2013) UISKEI++: Multi-device Wizard of Oz prototyping. *ECIS'13*, 24-27 June, London, UK, pp. 171-174.