Evaluating PiP: optimising user acceptance testing via heuristic evaluation

George Macgregor
University of Strathclyde

12 February 2012

Those that follow the PiP project will be aware that PiP entered a phase of evaluation in late 2011. A formal evaluation plan was finalised in early November and, since then, we have been working hard to execute the numerous evaluative strands detailed in the plan. The overall purpose of the evaluation is to examine core project deliverables, to assess their fitness for purpose and their impact on wider institutional systems and processes. This involves - among other things - systems testing, the gathering and analysis of user data using a variety of research techniques in order to identify opportunities for system and process enhancements, interpreting the perceptions and reactions of primary and secondary stakeholders, and assessing the overall institutional impact of the project. An additional objective is to use the findings from testing to share lessons with the HE sector about ways of improving curriculum design and approval processes.

Our evaluation identifies four distinct evaluative strands (or activities), each containing numerous subtasks:

1. Evaluation of system pilot (C-CAP system).
2. Evaluation of pilot implications for other institutional systems and processes.
3. Evaluation of impact on - and re-engineering of - business processes.
4. Project evaluation.

Many of these evaluative strands are interdependent and feed into other strands, as illustrated in the diagram below. We have recently finished the first evaluative strand in its entirety, which is pleasing since it is by far the largest and most complicated, and has implications for the success for strands #2 and #3.

Evaluative strand #1 was focussed primarily on evaluation of the pilot curriculum design and approval system (C-CAP). This included detailed user acceptance testing, entailing special "think aloud" curriculum design tasks which were assigned to academic participants. Their experiences with C-CAP were grabbed using screen capture software for subsequent protocol analysis (i.e. "think aloud protocols"). Post-task stimulated recall was also gathered for qualitative analysis, and a pre- and post-session questionnaire instrument was deployed. All of the aforementioned was designed to elicit data on system efficacy, participants' perceptions of the system, and the extent to which it could improve the curriculum design and approval process from a business process perspective, but also a pedagogical perspective. I intend to blog about the user acceptance work in a future post. My story for this blog post, however, begins before the user acceptance testing and focuses on a usability engineering technique: heuristic evaluation.

Heuristic evaluation is an established method of usability testing and is most commonly deployed in Human-Computer Interaction (HCI) research (e.g. to test user interface designs, technology systems testing, etc.). Heuristic evaluation techniques enable a suitably trained evaluator to examine the object of study (e.g. interface or system) and assess its compliance (or lack of) with recognised heuristic evaluation principles, thereby testing its usability. Results of the evaluation are then used to inform system modifications. The approach is favoured in incremental design methodologies, which is convenient because PiP is using such a design methodology. Over the past 15 years a variety of heuristic frameworks have been proposed, the most popular of which was developed (incrementally, funnily enough) by Jakob Nielsen. We lack the space to explain these heuristics in detail (Nielsen provides a useful summary at Useit); suffice to say, Nielsen has 10 heuristic principles, each of which can be assessed for their severity using a specially devised severity ratings scale (e.g. 0 = This is not a usability problem, 4 = Usability catastrophe).

Heuristic evaluation functions as an informal and relatively rapid means of engaging in usability engineering and debugging, and is often used as a precursor to user testing. This need to debug as a precursor to user testing is especially true of a project like PiP (and probably other projects from the Institutional Approaches to Curriculum Design programme). The curriculum design process is one that – in HCI terms - can be considered as cognitively onerous. It is a creative process demanding a high level of cognitive resource from users, for example, to adequately design a high-impact assessment task which simultaneously demonstrates constructive alignment with the stated learning objectives. Users (i.e. curriculum designers) can ill afford to devote unnecessary cognitive resource on interpreting the interface of an unusable or confusing curriculum design system. Any system has to ensure a high level of usability if it is to truly support and inspire academics in the curriculum design process, and ergo the approval process. From an evaluation perspective it is also imperative that any analysis of curriculum designers' interaction with the system focuses on deeper system and curriculum design issues, rather than on trivial or careless interface issues, or system errors that could easily be debugged prior to user exposure.

Almost 10 years ago I was involved in a research project that developed an innovative distributed information retrieval system. This retrieval system demonstrated extraordinary complexity in the way it retrieved and aggregated results for users; unfortunately – from a user perspective – it looked terrible and was close to unusable. A series of user testing sessions was arranged in order to gather rich data on users' perceptions of results relevance, precision and recall. Sadly, the data we eventually gathered from these user testing sessions failed to satisfy our requirements. Our users were too preoccupied with decoding the unusable interface and complaining about the front end aesthetics to comment meaningfully on all the wonderful back-end tricks. And, of course, it was in the latter we were most interested. On reflection I can see that if we had invested time on a pre-study heuristic evaluation most usability and interface issues could have been addressed prior to user testing, thus optimising the system experience and setting the stage for useful data collection.

Basic usability of any system should therefore be ignored at ones peril. This is especially true when the system concerned provides the basis for exploring "bigger issues", whether this is users' evaluation of results relevance or – within the PiP context - the design and approval of curricula. Any subsequent data collection is likely to be compromised if the user has to devote an above average amount of cognitive resource to understand aspects of a system which should, where possible, be self-explanatory. It fails to create a valid data collection environment and it skews data towards superficial system problems which are often not indicative of a project's wider raison d'être.

The heuristic evaluation of the C-CAP system was performed in late November using a specially devised methodology. The C-CAP system performed generally well under heuristic evaluation. The system demonstrated good use of short cuts and accelerators. User control and freedom was generally very good, partly owing to the provision of familiar rich-text editors enabling incorrect actions to be "undone", and a minimalist and uncluttered interface design also ensured essential page elements were clearly visible. The use of rich-text editors also provides a degree of consistency and demonstrated adherence to the de facto standard of the word processing dashboard. However, a total of 27 heuristic violations were found. Full details of these violations and their severity can be found in the associated report (to be made available on the PiP website soon). Of the violations found, 67% (n = 18) were classified at a mean severity rating of ≤ 2.67, and of these 11% (n = 3) were classified at severity rating 1 (Cosmetic problem only). Only 33% (n = 9) were classified at a mean severity rating ≥ 3.

Nevertheless, this systems-based evaluative approach has proven invaluable in optimising the value of our user acceptance testing. Findings from the heuristic evaluation – and the solutions it proposed - were immediately implemented in C-CAP. This ensured user acceptance testing could better capture data on the true potential of the C-CAP system to facilitate curriculum design and approval, and on its ability to aid process and pedagogy. It would be disingenuous to suggest that no usability issues were recorded in our user acceptance testing; but it wouldn't be to state that such comments accounted for a minuscule fraction of the data collected. A lesson well learned from 10 years ago.