

BasKet Note Pads Usability

Frank Ploss (frank.ploss@informatik.uni-hamburg.de)

The basic idea for my diploma thesis is to find and adapt an existing approach to usability engineering for an open source development project, carry out the approach and evaluate the process and its results. The working title of the thesis is “Scenario-based Usability Engineering in an Open Source Project—A Methodical Experiment”. Most usability engineering approaches assume that they are used to develop individual software. In contrast, most OSS projects, and BasKet too, develop standard software with no defined context of use—existing approaches have to be especially adapted to work for such cases.

BasKet is a desktop software not precisely aimed at a defined group of users. Therefore one cannot walk up to the users in their specific context, ask them how they do their work and derive from that which tasks in “real work” to support by software and how. Instead, there is a rather open space of what BasKet can be used for now and in the futures. From this, two questions for the development of BasKet arise:

- In which direction should development of BasKet go?
- Where are the limits? How can be decided to include or not to include a certain feature? (e.g., would it make sense to extend the To Do list functionality or better leave this to a specialized planner tool?)

To answer these questions, it is important to know who uses BasKet for what and in what context. But because BasKet is a generic desktop tool, it can be assumed that the users’ contexts and uses of the tool vary widely. So how can it be achieved to integrate the different views and decide where to put emphasis on? My idea is to make the descriptions of scenarios, personas and similar documents open via a website and to handle them as a public good like source code. Interested users can then read the scenarios and see if they recognize themselves in them, understand how other people work with BasKet, comment on them, correct misunderstandings, develop new ideas, and even write new scenarios. The great advantage of this will hopefully be the possibility to exploit the users’ creativity and their unique insight into their own work, and also that they might get a feeling of having influence on BasKet’s development (which otherwise is difficult to create for non-programmers), which again may lead to more participation. Of course, all of this is a theoretical idea by now, but you can’t know until you’ve tried (a “methodical experiment”).

I would like to work on BasKet for several reasons: 1. I use it and I like it myself, 2. the project has an affinity to usability. 3. it is a general-purpose tool and therefore is a good candidate for trying out my “experiment”.

Here comes my preliminary plan. It looks different from the project plan you have already set up, but the intention is the same: to analyze who the users are, what they do with BasKet and how BasKet can be enhanced based on this data. The differences are in the way I formulate the plan and in the emphasis participation gets.

Analysis step. In this step, the current situation is analyzed with an online survey and Like-Back data. Personas (stories about typical users) and scenarios (stories about current use) are used to document this in a way accessible to everyone involved (users and developers). These will be published and be opened for discussion and change. Define what BasKet is and shall be and the limits thereof (the vision).

Design step. Different types of scenarios will be used to put down stories about improved use of BasKet and about ideas for new feature ideas. These again will be published and opened for discussion and change.

Evaluation step. Ideas and improvements will be evaluated using paper prototypes (or even prototypes in code, but that's a problem of limited time), observations will be evaluated and used for refinement of the prototypes. I know many people using GNU/Linux, so there is no problem finding test persons (5 to 10 should be ok). Also, the prototypes will be published again, and could be commented on.

I've got half a year for writing my thesis, and the practical part (the above) will take half of that, that's three months. I have not much practical experience in usability engineering (never done a whole cycle, only tried out some techniques), but I think I've got a good overview about methods and techniques. My practical experiences until now have been in developing and system administration (using GNU/Linux for 10 years).

November 8, 2006