Interactive Prototype (Group)

Due: Thursday, November 29, 2007

Goals

The goal of this assignment is to learn how to build prototypes of user interface ideas using an interactive user interface design tool. You will revise your user interface ideas based on the low-fi eval and then use interactive tools to build a prototype of the design.

Interface Redesign

Use the results of your low-fi prototype tests to design a revised user interface. Develop new and/or revised scenarios for your tasks by storyboarding your ideas. The tasks that most of you used in the low-fi assignment should be sufficient for this, but some were simple or partial tasks that did not adequately cover your proposed functionality. Make sure to revise those tasks. If you are changing your tasks, make an appointment with us to present your new tasks, design ideas, and storyboards for discussion.

Prototyping

You will use a prototyping tool to create an interactive prototype of your application. For most applications, we would like you to use either DreamWeaver[1] or DENIM (http://dub.washington.edu/denim/). If your project is especially suited to Topiary (location-based at http://dub.washington.edu/topiary) or ActivityDesigner (activity-based at http://dub.washington.edu/activitydesigner), you may use those – but even then it might be good to first see what you can do with DreamWeaver/DENIM. If you go the research tool route, you will run into bugs and missing features, but with less complexity – that is part of the deal on the research tool path. So, decide what you'd prefer.

Your prototype should implement the three scenarios that you developed for your tasks. You can import images into DENIM/DreamWeaver to start to account for the size and resolution of your target device by creating an appropriate background (e.g., for a phone case & buttons). Since DENIM is an informal tool these aspects of the design will still be rough and at best might be considered a **medium-fidelit**y prototype (so, lines don't have to be straight, etc. – in fact that is hard to do in DENIM – you can use typed text).

The underlying functionality does not have to be fully implemented. For example, applications requiring large databases of information can instead have a sufficient number of hard-coded data points for supporting the three tasks.

You have a very short period of time to complete this prototype, so you should focus on showing only what is essential. You will likely have to make some difficult choices!

[1] Download a free trial of DreamWeaver at Adobe's site and there are lots of good mobile tips via "google dreamweaver mobile"). **If you'd like to use a different commercial tool, you must get our approval.**

Deliverables

1. **Prototype**
   Your prototype must be accessible and/or executable by everyone in the class from your web site. It must be accompanied by a README file that describes the tool that it runs with and operating instructions, including any limitations in the current implementation. **If this is not working on the due date, you will get a *zero* on this assignment (**why? The other team needs to run a heuristic evaluation on your assignment in the next 4 days after the due date).
2. **Report**
   You will submit two copies of a printed report of no more than four pages of text in class (images free and required). You must also put a copy of the report online on your course web site (on our server).

Report

The report should follow this outline with separate sections for the top-level items.

1. **Problem and solution overview (1 paragraph)**
2. **Tasks (1/4 page)**
   - 3 representative tasks to test your interface (labeled easy, medium, hard)
3. **Revised interface design (1 page plus screenshots or scripts for speech UIs)**
   - Changes as a result of low-fi testing and rationale behind the changes (refer to screenshots or scripts)
     - Sketches or scripts for unimplemented portions of the interface
   - Scenarios for 3 tasks
     - Storyboards of scenarios (annotated DENIM screenshots, web pages, etc.)
4. **Prototype overview (2 pages)**
   - Tools
     - How the tools helped
     - How the tools did not help
   - Overview of implemented UI (reference figures or scripts from next section)
   - What was left out and why
   - Any wizard of Oz techniques required to make it work
5. **Prototype screenshots or scripts (as many as needed)**

Grading

Here is the grading criteria for the report and prototype (100 pts total):

Design (40 Points)

- Tasks
    - Do the tasks cover the interesting features of the project?
    - Do the tasks have an appropriate difficulty/complexity specified?
    - Do the tasks altogether form a compelling story for the project?
- Changes
    - Were appropriate changes made to address the important problems discovered during the low-fi testing?
    - Are these changes well illustrated with screenshots or scripts?
- Transition from low-fi to interactive prototype
    - Were some of the limitations of the low-fi addressed?
    - Were appropriate constraints from the final target platform considered?
    - Were any non-standard interactions described and justified?

Prototype (30 pts)

- Is the prototype accessible and *fully* working?
- Can users complete the three tasks with the prototype?
- Were appropriate tradeoffs made between functionality and completeness?
- Are the limitations and tradeoffs described and justified in the report?
- Does the README file summarize these limitations and any other details needed?

Report (30 pts)

- Writing
    - Does the report cover all the topics in the outline?
    - Does the organization follow the outline?
    - Are sub-sections used for easy scanning of important parts?
- Screenshots and Storyboards or Scripts
    - Are important figures referenced and placed inline with the text? *
    - Is there a complete set of screenshots or scripts in the appendix?
    - Are they clearly annotated?

* Use Relevance-Enhanced Image Reduction to create effective thumbnail images.