CS220: Human-Computer Interaction, Summer 2016

P2: Paper prototype implementation and testing

Due Thursday, July 4th at 11:59 PM.

The goal of this project milestone is to build and test the first implementation of your project, in the form of a paper prototype. Your paper prototype should be able to handle at least the 3 scenarios you described in P2. You will be testing your paper prototype with **three** (or more) users who represent your user population, and who are **not** taking CS220 this semester.

Reading

Paper prototyping as a usability testing technique:
http://usabilitygeek.com/paper-prototyping-as-a-usability-testing-technique/

Building your paper prototype

Your prototype should be complete enough to "run" a new user through your application.

Use the Balsamiq Mockup software to create your prototype- (you can use the Web App trial) (http://www.balsamiq.com/). You should print out your designs and test a **paper-based** prototype.

In class, on **Wednesday July 13**, we will run *pilot* users (your classmates) on the **first draft of your paper prototype**. Your pilot users will help you find the most obvious usability problems and help you practice running your paper prototype. Keep refining your prototype after this class session until you feel satisfied that it works. Then, on Thursday, test it on **at least 3 users from your target population, all from outside the class**.

Choosing What to Prototype and Test

You may need to adjust your scenarios so that they explore the riskiest parts of your interface. A part of your interface is risky if its usability is hard to predict, or if its usability strongly affects the usability of the whole system. Parts of your interface that are new and different are potentially risky. By contrast, a username/password form isn't risky at all, because it's a familiar and well understood. A frequently-used feature might also be risky, because the efficiency of the whole interface depends strongly on it. Finally, A feature in which user errors might be common or hard to recover from is risky.

Risky parts need the most design iterations, so they'll give you the most payoff from prototyping. If you make sure your scenarios cover the risky parts now, you'll be able to plan your subsequent (computer-based) prototypes better.

Preparing for Testing

Before testing your prototype, you should:

1. *Prepare a briefing for test users*. This should be at most a page of information about the purpose of your application and any background information about the domain that may be needed by your test users to understand it. Make your briefing short, clear and concise, it should not describe how to use the interface.

2. *Write your 3 scenario tasks on separate index cards*. Just write the concrete goals of the task (e.g. "use this mobile app to buy milk, tomatoes, and bread"). Don't write the specific steps to follow, since that's for your users to figure out. The tasks should be brief. Running your 3 scenario tasks should take roughly 5 minutes.

3. *Choose roles for your team members*. One person must play the device. The other team members will be observers. It may be useful for you to swap roles after every user, so that each of you gets a chance to try each role, but decide how you'll do it in advance.

4. *Practice running your paper prototype*. Every team member should practice playing the device, learning the steps involved in making the prototype functional, such as rearranging pieces and writing responses. It isn't important to be fast, just competent and confident. A few trials are enough. Make sure your prototype can handle the 3 scenario tasks you chose.

Running the Tests

When you run your prototype on a user, you should do the following things:

1. *Brief the user*. Use the briefing you wrote up to describe the purpose of the application and background information about the domain. Up to 1 minute briefing should be enough.

2. *Present one task*. Hand the index card to the user and let them read it. Make sure they understand the task.

3. *Watch the user perform the task*. Take notes from your observations.

4. *Repeat with the other tasks*. Run the three tasks you implemented.

5. *Bring extra materials on Testing Day*. Having extra blank Post-it notes,

correction tape, and index cards will help you improvise or make small fixes to your prototype between users.

What to Hand In

Update your team webpage so that it contains a section P2 Paper Prototyping, with the following subsections:

Prototype photos. Digital and annotated photos of the pieces of your prototype. Show the prototype in interesting states; don't just show a blank window.

Briefing. The briefing you gave to users.

Scenario Tasks. The tasks you gave to users, as you wrote them on the cards.

Observations. Usability problems you discovered from the testing. Describe what users did, but don't record users' names. Your observations should include both your pilot users, and at least 3 users from outside the class.

Prototype iteration. Describe how your prototype changed between your pilot users and the real users.

Resolutions. What did you learn about the parts of your interface from this prototype? Propose design solutions for the usability problems you found.

Video. A video clip demonstrating a ‘run’ of your prototype on at least one task.