Overview of scheduled topics

This is a plan, not a map

Things are still slowly evolving, but this is the high-level plan. You can think of class as having five main “chunks”. Those chunks are laid out below.

Weeks 1-2: overview of design and HCI

The first two weeks are an overview of design in general, as well as a broad introduction to HCI. The main goal is to get you thinking like a designer: adopting designerly attitudes, noticing good and bad design, becoming comfortable talking about it, and starting to develop the vocabulary that interaction designers use for these conversations.

Readings and notes:

- RND Aug 26, Course intro and design attitudes
- RND Aug 31, Design is everywhere
- RND Sep 2, Miniproject and working in groups

Weeks 3-5: Design as Communication

This part of the course asks you to think about design, not just as the creative activity many of us have notions of when we say design, but as an act of communication. This has a number of facets:

When designing for people, it helps if you can learn about them, either through direct techniques like interviewing and participant-observation, or through indirect techniques like surveys, data/log analysis, and observing the things they already use.

This learning needs to be translated into forms that you and your team – and possibly also the people you design for – can hold onto and use. Info about prototypical users, their goals, their tasks, and desirable interactions with the system is important to capture.

Likewise, we need to communicate our design ideas, though sketching and storyboards and other tools that allow us to get feedback both from designers and clients.

We'll practice these skills, and refine skills from the first two weeks, through your blogs and through a project you'll do over the course of these three weeks.

Readings and notes for design as communication

- RND Sep 7, Design processes and goals
- RND Sep 9, Learning about people
- RND Sep 14, More learning about people!
- RND Sep 16, Remembering what you learned
- RND Sep 21, Ideation, sketches, storyboards, and fidelity
- RND Sep 23, Justifying your design (Hao-Chuan Wang, guest lecture)
- RND Sep 28, Miniproject I critique + Maxi-project intro

Weeks 6.5-10: Design and constraint

In this part of the course, we'll be focusing on how the design constraints involved in interaction and interface design lead to design moves that are likely to be better or worse for the people involved. Of course, we've talked about some constraints already: basic ideas around psychology, as well as the detailed kinds of constraints that arise from looking at people in particular contexts with particular needs and practices that should inform your thoughts throughout the design process.

Here, we look much more closely at specific constraints. In particular, I make a bold claim: just like any other kind of engineering, interaction designers are constrained by the materials involved. It's just that here, the primary limiting material is not the computer, it's the people involved.

So we focus on that: what aspects of human cognition and physical capability are important for designers to recognize in designing interfaces for people? Memory, attention, and perception are three useful principles that guide both interface and information design, both informally in the sense of guidelines and principles (such as CRAP, actually), and formally in the guise of “Laws” like Fitts’ Law and models of task performance that allow us to estimate how long it will take people to use various interfaces. This is not the only dimension of usability – more on that in the second half of the semester – but it's an important one.

We also have a number of guest lecturers for this part of the course, experts on cognitive psychology, input and output beyond the WIMP (windows, icons, menus, pointers) paradigm, and game interfaces. It should be fun.

Notes and readings for design and constraint.

- RND Sep 30, Constraints and opportunities
- RND Oct 5, Input and output
Weeks 10.5-13: Instantiation, iteration, evaluation, and issues

One of the key issues as you go through design processes is knowing whether the ideas you're generating are any good. Are they "usable", for whatever definitions of usable your project has adopted? Do they meet users' needs, suit their goals, help them accomplish their tasks?

Some of the tools you've already looked at help answer these questions. You can use personas, tasks, and scenarios to help you, when you make choices, remember important details of the people you're designing for, and these data help you make decisions. Likewise, knowing the physical and social constraints that being human imposes on designs can help you make choices.

But not always. Sometimes you don't know what to do. Sometimes you have several plausible ideas. One option is to argue endlessly over abstractions. I don't recommend that. Instead, I recommend a combination of techniques. First, develop lots of low-fidelity versions of the interfaces. These are the sketches, the storyboards, and other paper/less functional versions of the interface. Seeing them realized in physical form is very useful.

Second, use some of the techniques we'll be talking about in this part of the course. One excellent option is to just test your interface using some common tasks with only a few real users. This will uncover a surprising number of interface issues quickly, and often cheaply, and we will talk about that later in the semester. Sometimes, however, users are unavailable or the prototype isn't quite ready to show or you just have no money or time to set up user testing. Then, you can use the "discount usability" techniques we will discuss. Applying guidelines and standards, doing cognitive walkthroughs, and conducting heuristic evaluations are all techniques that can give you lots of information about your interface in a very short time.

Third, use those techniques often, and rapidly change the interface as you learn about it. Once you know that option X doesn't work, or that you need to change element Y, you make a new version of the interface. (This is another virtue of low-cost, easily-modifiable prototypes.) There's no sense in letting ten people see the same error, or having to work around it while you apply some of the discount techniques. Try to fix errors early and often; you'll be glad you did.

More on this in lectures.

Readings and notes for iteration and evaluation

- RND Oct 28, "Usability", discount style
- RND Nov 2, Which widgets, and why?
- RND Nov 4, Heuristic evaluation
- RND Nov 9, Awareness, interruption, and eyetracking (Jeremy Birnholtz, guest lecture)
- RND Nov 11, Prototyping tools
- RND Nov 16, TBD, depending on Nov 11 lecture
- RND Nov 18, Testing with people (Hrönn Brynjarsdóttir, guest lecture)

Weeks 14-final: Putting it all together

The main goal of these two weeks is to think about where you might go with HCI in the future, to ponder all the things we didn't get to talk about, and to try to put together the ideas of design being everywhere, design as communication, design under constraint, and design as iteration and evaluation.

Readings for putting it all together

- RND Nov 23, Designing for experience (Walker White, guest lecture)
- Nov 25: Thanksgiving break
- RND Nov 30, Wrapup
- RND Dec 2, Prototype demos and testing