Social Computing Systems

Walter S. Lasecki

EECS 498, Fall 2018

Today

- Basic design process
- Project ideation / team formation

Why are we covering this?

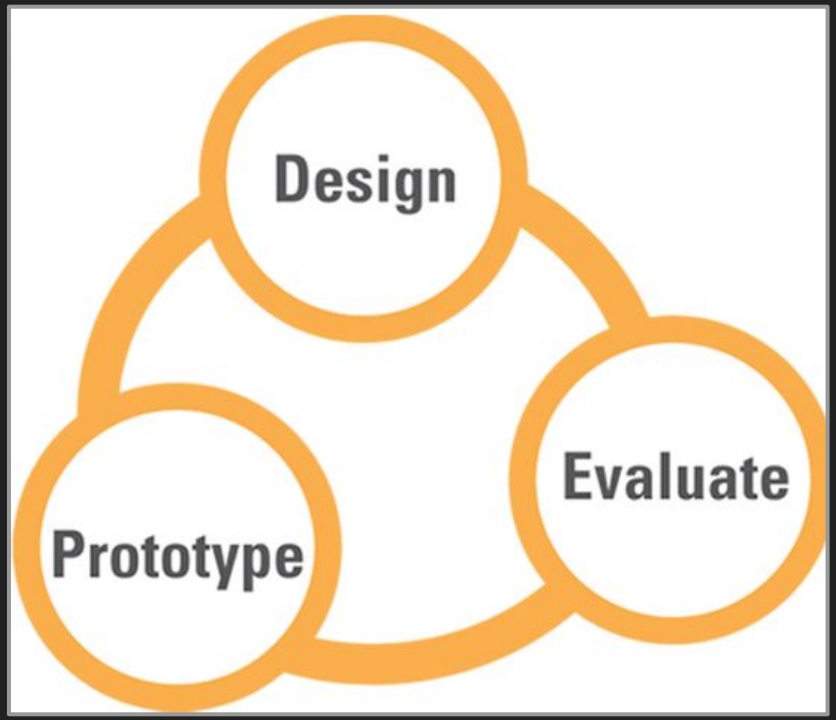
Building without design is aimless wandering

The design process can help guide good design

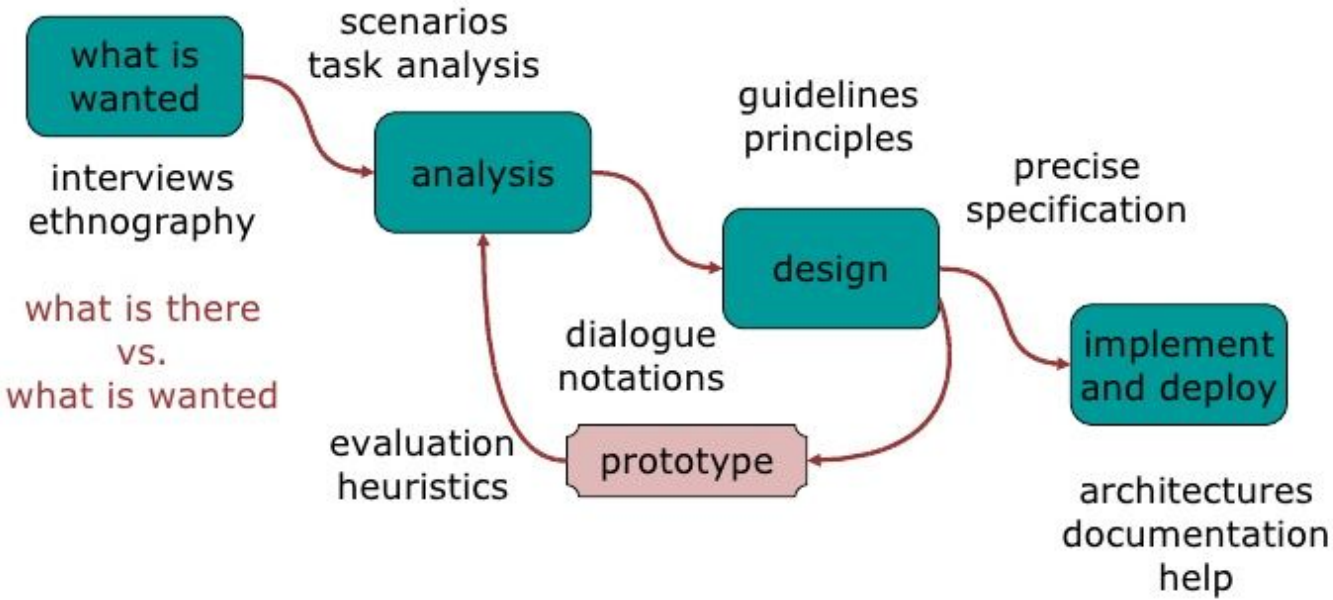
We want to build useful things

Designing Real Systems

System Design



The process of design



Motivation and Initial Design

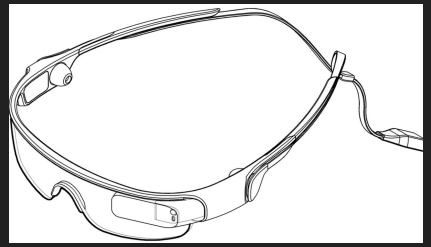
- Study the context of the problem, and how a solution would fit
- Discover + articulate a problem
- List stakeholders / needs / constraints
- Consider what tech is available and feasible
- Decide what to do first

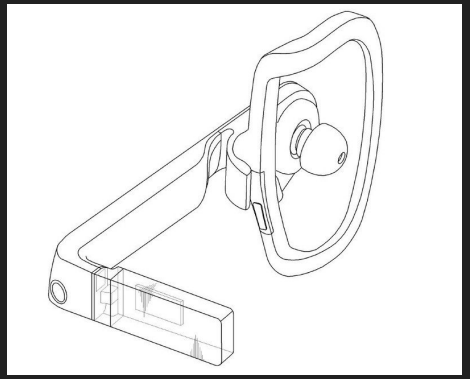
Prototyping

- Start with a low-fidelity model
- Iterate
- Test with ‘example’ users
  - e.g., within-team tests, hypothetical user profiles, etc.
- Iterate
- Test with real users
- Iterate
- More iterating
- Keep iterating...

Types of Prototypes

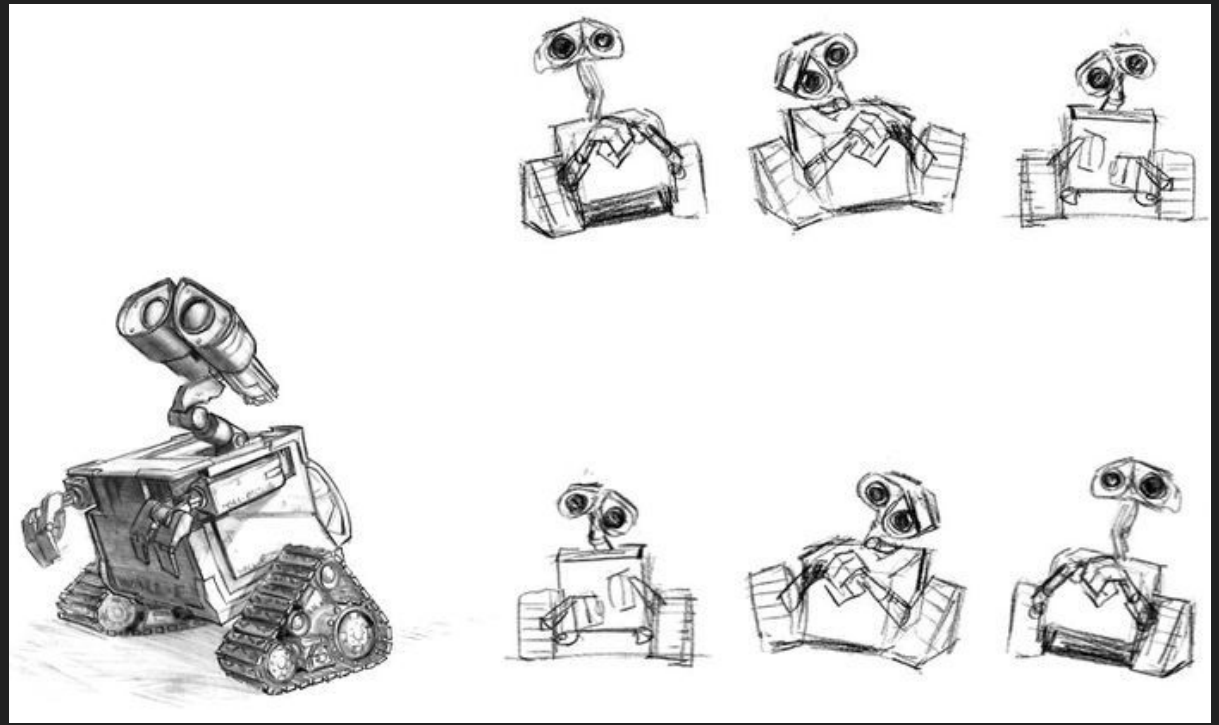
- Sketches
- Paper prototypes
- Physical mock-ups
- ‘Workbench’ build
- One-off production
- Small-batch production
- Release-ready





What Should a Prototype Be?

- Quick
- Cheap
- Explanatory
- Better than nothing

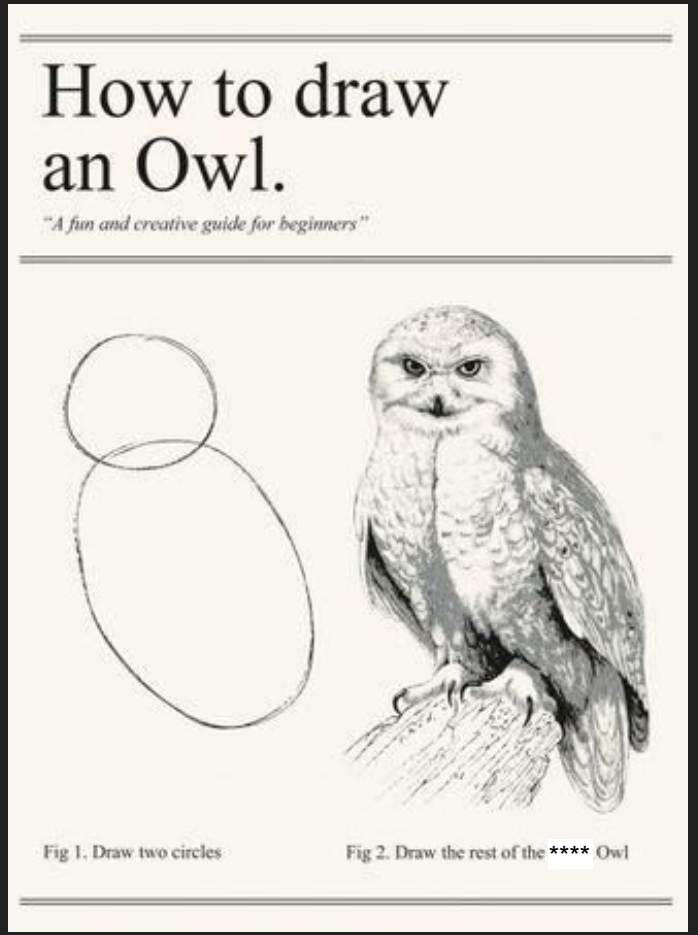


Prototyping Example: Google Glass

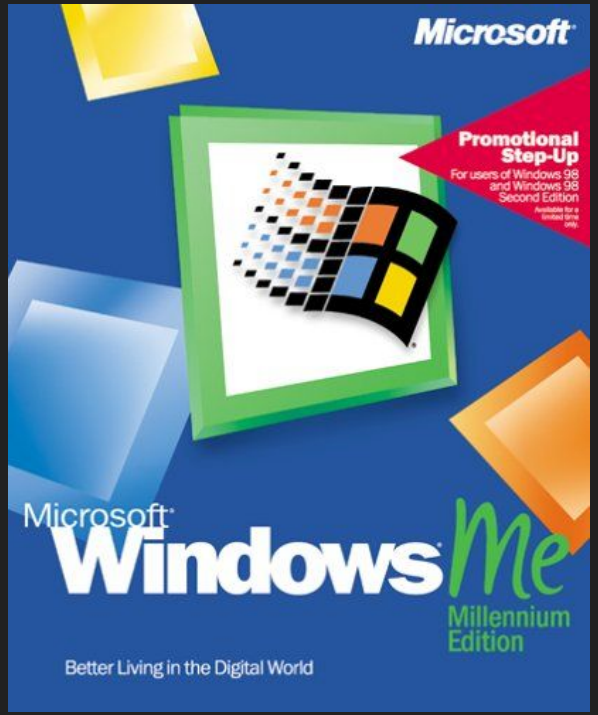
Thousands of dollars + months of effort

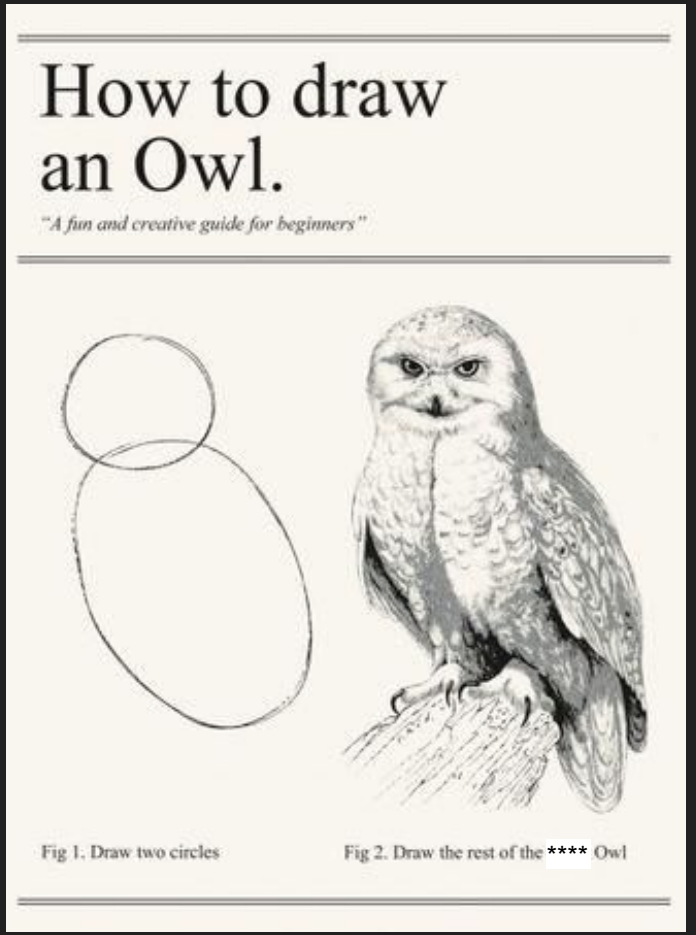
Millions of dollars + years of effort

How Not to Prototype



How Not to Prototype





User Testing

- Qualitative (understand experiences)
  - Observation (e.g., ‘Think Aloud’ study)
  - Survey and interview
  - Preference evaluation

- Quantitative (understand performance)
  - Task-based metrics (e.g., completion time)
  - Data-centric (e.g., data mining of user interaction traces)

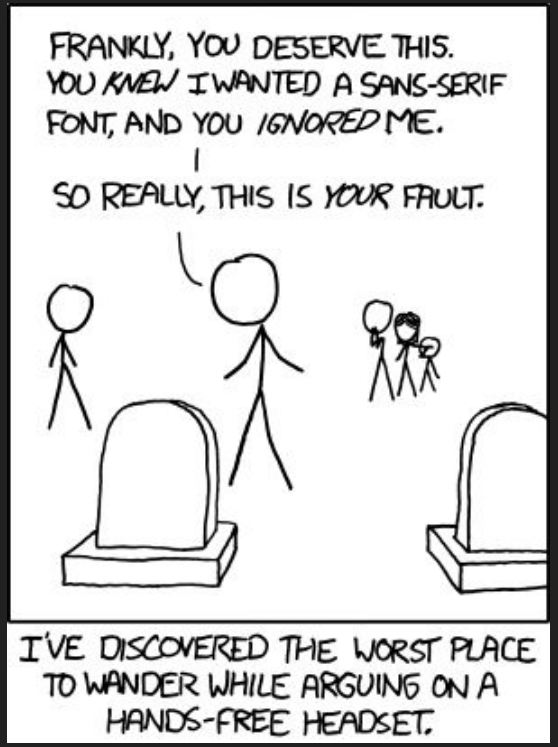
User Testing

https://twitter.com/designuxui/status/576432203560685568

Results

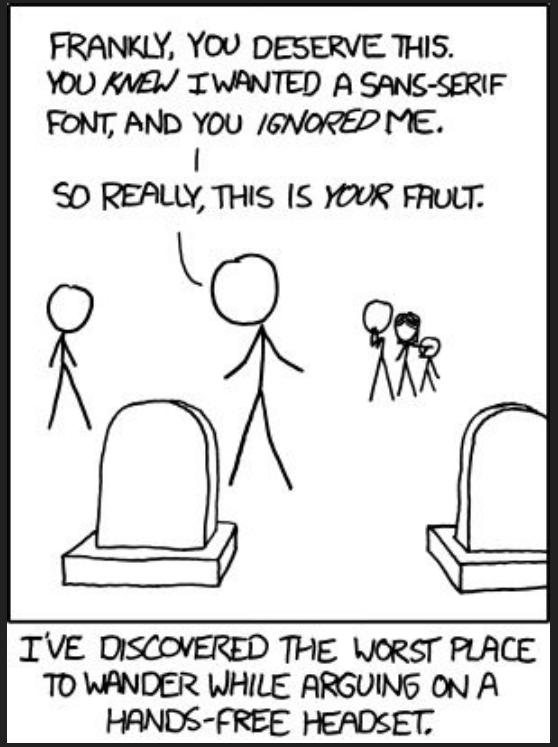
Benefitting From User Testing

- What went wrong?
- What went right?
- Is there a ‘gulf of understanding’?
- How could these issues be addressed?
  - Functionality
  - Instructions / priming
  - Experience… (e.g., for ‘expert’ features)
- Are the users the right stakeholder to address?



Benefitting From User Testing

- Key point: user tests identify intermediate problems
- ...which requires more understanding



Example Design Process Walkthrough

- We work in a hospital, and see that a large % of the mistakes made are made because one MD doesn’t know what the MD from the last shift did.

- Where do we start?

Example Process: Contextual Inquiry (CI)

- User-centered design method
- Observe the processes at the hospital. Watch an MD work.
- Understand the ‘pain points’, their causes, and the current state of the art
- Interview participants to understand the ‘why’ behind the ‘what’
- Understand broader context: stakeholders, organizational structure, etc.

Example CI Findings

- Doctors check patient charts at the start of their shifts to see history
- Doctors are overloaded, and often forget to take notes on the charts
- Nurses don’t always double-check charts until the end of their shift
- Nurses’ and doctors’ shifts are staggered to prevent high turnover
- Doctors know they sometimes miss recording information in general
- However, they can’t recall when exactly after the fact

Example Process: Example Profiles

- Create example profiles for relevant stakeholders.

Example Process: Brainstorming Solutions

- What could address the problems we see?

Example Process: Refining Solutions

- What technology could we leverage to help achieve our ideas?
- What is too far off in the future?

Example Process: Prototyping

- What could we build quickly to try out our top ideas?

Example Process: Refinement

- Using a wood-block prototype, we ask doctors to pretend it’s a real tool
- We find that MDs remember to use the device when they’re far from the chart
- What should we do next?

Example Process: User Study

- We’ve built a ‘real’ version of our device. Now we deploy it in the hospital.
- How do we measure success?
- How else can we find ways to improve?

Questions?

Readings

Extra-credit reading (last week):

- Beyond Being There by Hollan and Stornetta

Next up:

- The Intellectual Challenge of CSCW: Socio-Technical Gap by Mark Ackerman

Ideation and Pre-design Process

< studio >

Finding a team

Group by question area

Brainstorm in pairs

Brainstorm in groups

Think about target outcomes

Present your ideas

[Start to] form groups

A reminder on brainstorming

Brainstorming is a creative “branch out” process. Refine later.

Having more ideas is better than having better ideas.

For problem brainstorming, this creativity is focused on “possibly-possible futures”

Get crazy. What is the thing you really wish could exist in the world?

Topic areas (these are also research problems I'm thinking about)

Teaching and learning

- Better teaching and/or learning in groups. Could be human-human, or human-AI (e.g., better teaching an ML system using human-human interaction).

Collective reasoning

- Groups think differently than individuals. How can we improve this (e.g., to better understand falsehoods and shoddy reasoning)?

Collective control

- How can people better control a collective process or physical object/token (e.g., drive a car or robot, or play a game)?

QUESTIONS

Why do this?

What is the most evident signal that the problem is “getting better”?

Why is it new to solve this (AKA what is the current gap tech can close)?

>> DUE Monday (11:59pm)

Two project ideas. One paragraph each, describing the problem you want to solve, including answers to the questions on the last slide.

Submit one per team (with all members listed).