DESIGN AT LARGE

real-world, large scale
...and sometimes disruptive

Scott Klemmer
UC San Diego
The Design Lab

Tiphaine Henry

Step 1 - POV

The daily amount of emails received makes it hard for the user to keep his private or professional mailbox well organized and to have a clear overview of the messages saved.

Step 2 – Experience prototypes

The app idea :

In order to tackle this issue, I wish to develop an app able to categorize directly the mails received in the mailbox and to present them in an interactive way.

The steps required for planning and executing the task :

- Ask for the user a list of the last week emails he received
- Categorize the emails
- Present the first layer of categorization
    - ask the user where he would click
    - ask whether he would delete the emails or not
- Present the 2nd layer of categorization & ask where the user would click
    - ask the user where he would click
    - ask whether he would delete the emails or not
- If a 3rd layer, repeat the operation
- Ask about what he would do to delete this or that emailFirst execution :

Context :

Our first user, Marcus, is a UCSD international student, belonging to the category of average email users. He checks his mailbox very regularly, 2 times an hour: he does not appreciate to have unread mails. Still, once the mails are read, he tends to forget to delete or classify them. His principal mailbox is Gmail, mailbox on which I had the occasion to work on for the experience.

Experience

I first collected the list of the 20 last emails he received this month and categorized them according to two criteria : the title of the mail and the name of the sender. 3 main categories appeared : *Brazil* (4 mails), *San Diego* (10) and *Management of Online Accounts* (6). I represented the amount of emails included in each category by means of proportional circles. Marcus chose to click on *Management of Online Accounts* – the smallest one - at first. This category included *Creation of Accounts* and *Password Reset Links*. He decided to delete this latter and store the first category. He then looked at *Brazil* which contained school emails he decided to keep. He finally checked San Diego. This last category would be composed of *UCSD*, *UPS* and *Financial issues* (Uber and Venmo). He looked at *UCSD* first that included *CSE 170* and *Teacher Correspondances*. He deleted unread Piazza Reviews and kept the rest. He then inspected UPS composed of a welcoming mail of the school. He decided to save it. He then looked at *Financial issues* composed of *Uber* and *Venmo* and decided to keep a record of the past transactions also.

Feedback

Marcus was very happy to see his mails sorted, claiming that it was more easy to find the mails he wanted on purpose. He also affirmed that the app could help him get rid of emails he would not have deleted otherwise. However he would prefer to see directly the unread emails coming into his server.

General impression concerning the execution :

+ :

Good interaction between the user and the interface shown (which would consist of drawings of the remaining mails)

Interesting that the first check concerns the management of the account better than school issues for example.

- :

It takes a little time to categorize the emails manually.
How to make these categories evolve over time so that they can fit best people and their lives ? Ex : when Marcus will come back to Brazil, he may not need to see the “San Diego” category directly but maybe store it into a “Trips” or “School” one once he graduates.

Adjustments for the second execution :

- Create an *Unread* category so that the user can directly see the messages coming up if he wants to.
- Propose the *Delete or Categorize* option for each of the unread mail, making it part of the process and not optional, so that the treatment of the emails can be more direct.
- Ask before the experience for the list of emails so that the execution can be more fluid.

Evan Barosay

	simplicity	Calculation accuracy	Organizational	gratuity-inclusive	Availability	Shareable	Splits the bill	Payment method built in
Evan's Tool		X		X		X	X	
TAB	X	X	X	X		X	X	X
Paper Napkin	X				X			
Venmo					X	X		X
Excel		X	X					
Calculator		X			X			

from the Lab...

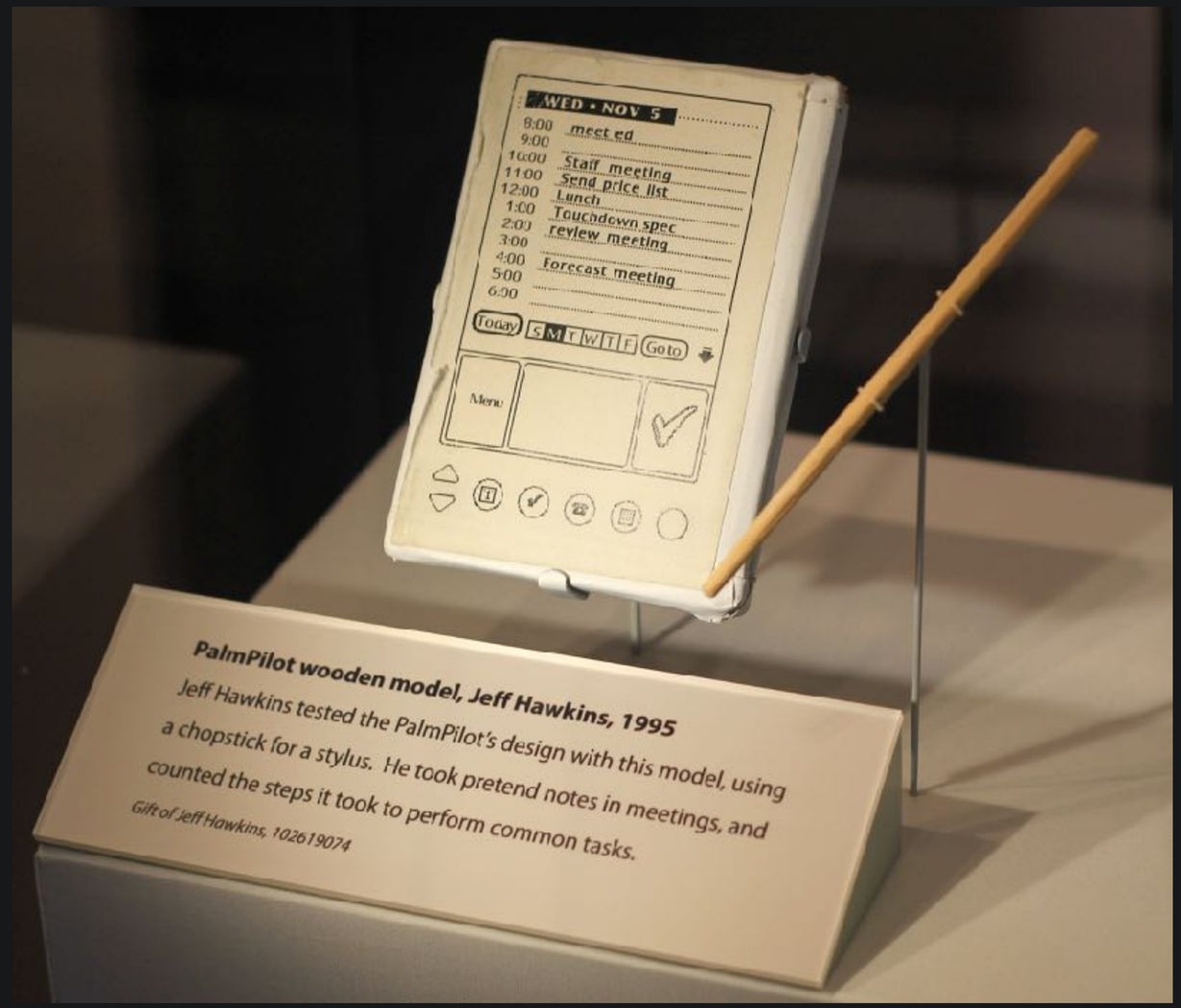
...to the Wild

An anomaly or a strategy?

The journey to 8 million users begins with a block of wood

In 2000, Palm sold nearly 8 million units and had a 76% share of the PDA market.

Plywood Experience Sampling



"If I wanted to check the calendar I'd take it out and press the wooden button."

-Jeff Hawkins

Computer History Museum in Mountain View, CA courtesy of Michael Hicks's image on flickr

Prototypes are postcards from the future

STANFORD HCI GROUP

AUTUMN 2007

CS147: Introduction to Human-Computer Interaction Design

Tuesdays & Thursdays, 1:15PM - 2:05PM, Hewlett 201

PROFESSOR
Scott Klemmer
Tues 10:15 - 12:15, Gates 384

STUDIO TIMES

Thurs: 12:00 - 12:50
Gates B02 (Bastea-Forte)
2:15 - 3:05 #1
Gates B02 (Bastea-Forte)
2:15 - 3:05 #2
Gates 392 (Cammarano)
3:15 - 4:05
Gates B02 (Wu)
4:15 - 5:05
Gates 498 (Brandt)
5:15 - 6:05
Gates 392 (Cammarano)

Fri: 9:00 - 9:50
Gates 392 (Patel)
10:00 - 10:50
Gates 392 (Patel)
11:00 - 11:50
Gates B02 (Wu)

Students will learn the fundamental concepts of human-computer i
in teams on an interaction design project that is supported by lectu

Studio: Submit homework, view others' work, and see your grades

Google Group: For technical discussions and implementation help.

Syllabus & Readings · Experimental Participation · Final P

SYLLABUS & READINGS

Week	Tuesday
1	*September 25* TOPIC: Introduction (PDF) (PPT)
	October 2

IxD HOME CALENDAR STUDIO DEVELOPMENT RESOURCES LOGISTICS SUBMIT WORK QUESTION? (PIAZZA)

Interaction Design

COGS120/CSE170 · Scott Klemmer · Winter 2017 · UCSD

Announcements

- Use Piazza to communicate with teaching staff.

In this course, you will learn how to design technologies that bring people joy, rather than frustration. To do this, you'll learn techniques for rapidly prototyping and evaluating multiple interface alternatives — and why rapid prototyping and comparative evaluation are essential to excellent interaction design. You'll learn how to conduct fieldwork with people to help generate design ideas. You'll learn how to make paper prototypes and low-fidelity mock-ups that are interactive — and how to use these designs to get feedback from teammates, clients, and users. You'll learn principles of visual design, perception and cognition so that you can effectively organize and present information with your interfaces. And you'll learn how to perform and analyze controlled experiments online.

Through a series of weekly assignments, you will complete a quarter-long project in teams of three. Each week, in small design studios, you present and discuss work with peers. The setting for the course is mobile web applications. The constraints of this small form factor set the stage for this challenge.

Weekly schedule

Lecture

Tuesdays 9:30am-10:50am
CICC 101 (Copley International Conference Center)

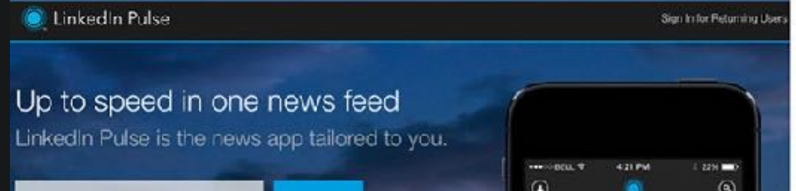
Labs

Thursdays 9:30am-10:50am
CICC 101 (Copley International Conference Center)

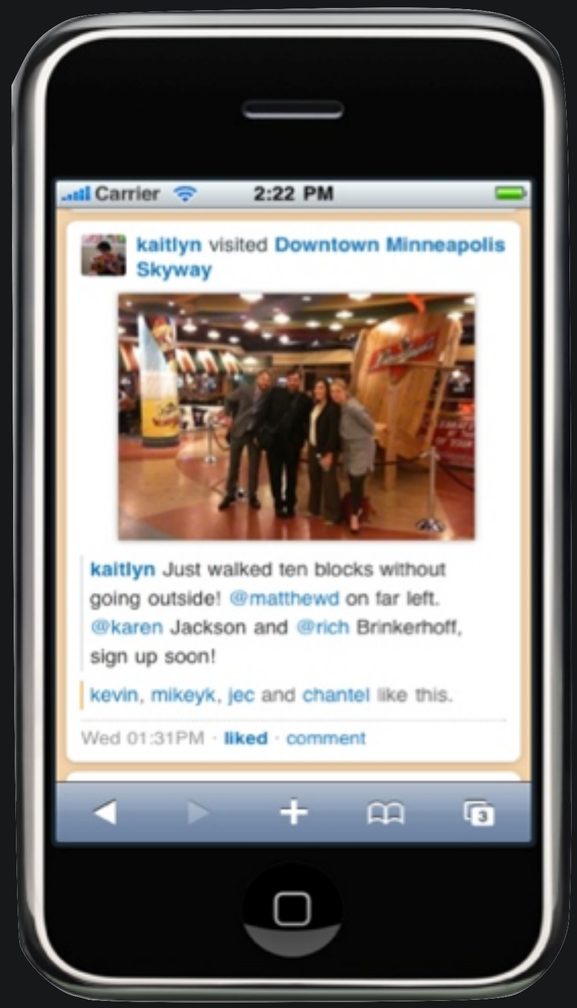
Studio Section

Fridays 8:50-10:00am, 10:10am-11:20am, 11:30am-12:40pm, 2:50pm-14:00pm, or 14:10pm-15:20pm
CSB 180, CSB 272, HSS 1346

The successes are tremendously exciting

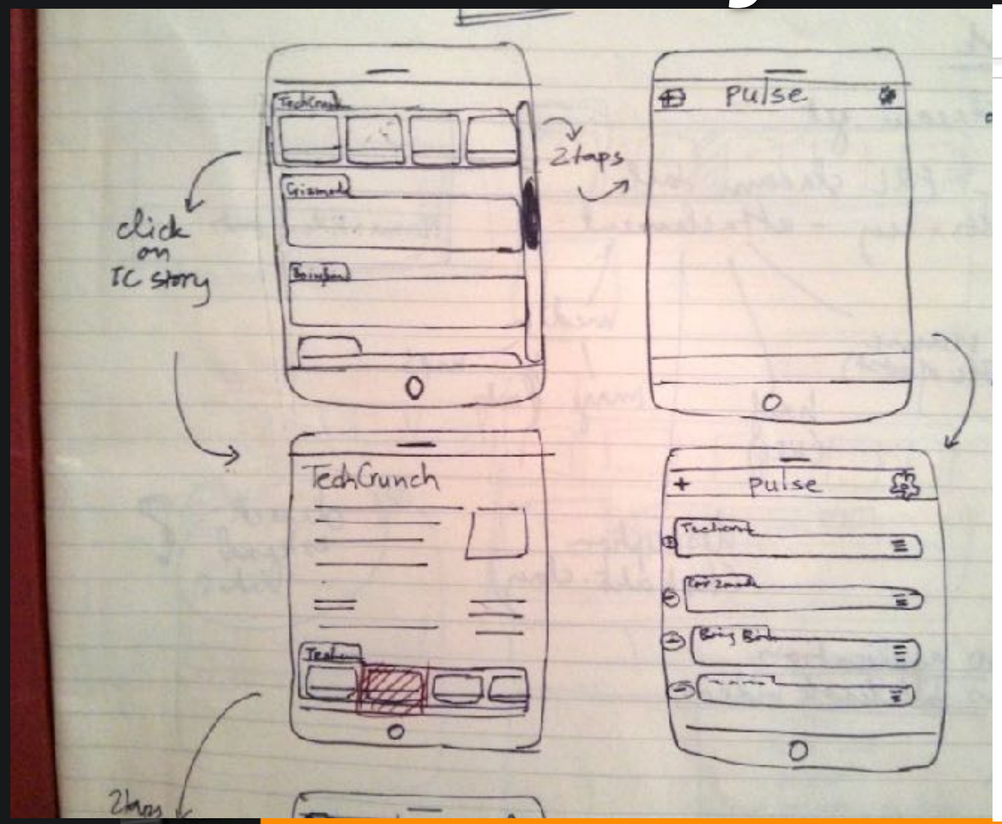


How do you ‘sketch’ a social computing environment? Mike’s Burbn experience



If at first you don't succeed…
From MVP to >400m active users

An anomaly or a strategy?



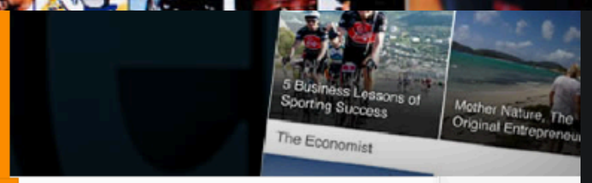
SECTIONS HOME SEARCH The New York Times

Bits

The iPad Pulse Reader Scales the Charts

By BRAD STONE JUNE 1, 2010 6:10 PM 16

Starting Up with Design Thinking: The Story of LinkedIn's Pulse



Home Interviews

Recent Popular Random

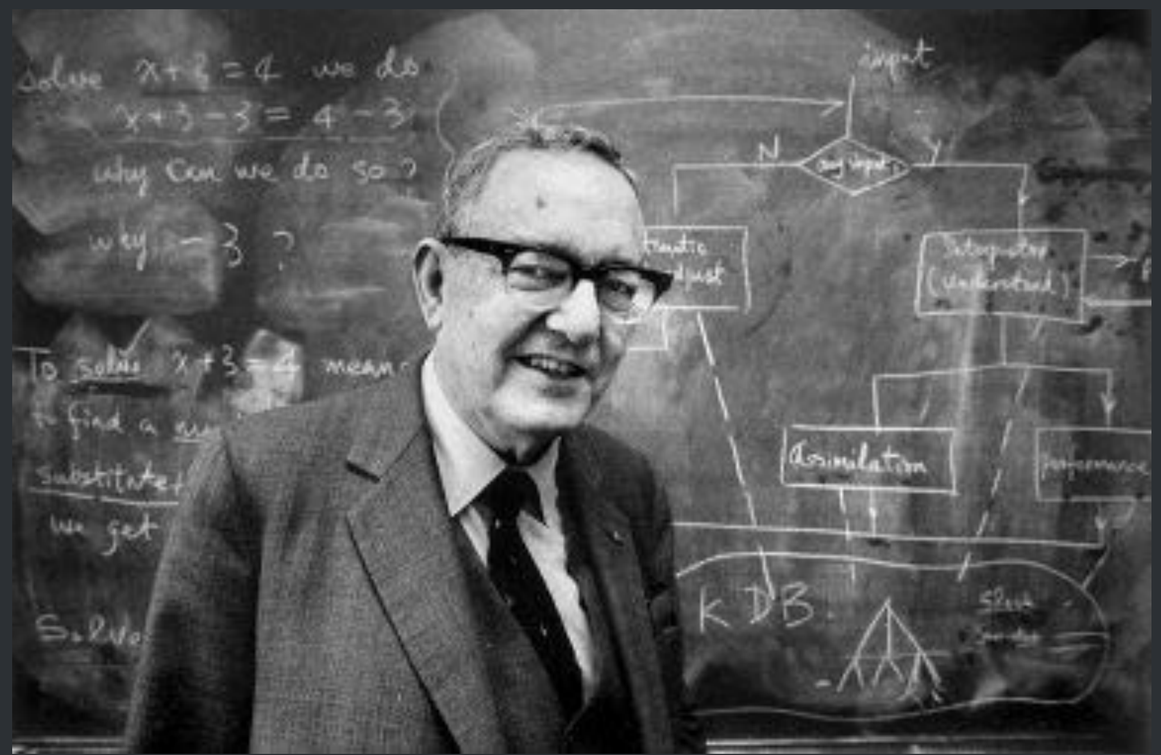
We met Akshay Kothari, one of the two co-founders of Pulse, a business reader app that was famously displayed by Steve Jobs at the Apple Worldwide Developers Conference in 2010 and was later acquired by LinkedIn for 90 Million Dollars. Akshay and his business partner Ankit

Comfortable with Feeling Uncomfortable: Innovation at the San Francisco Opera

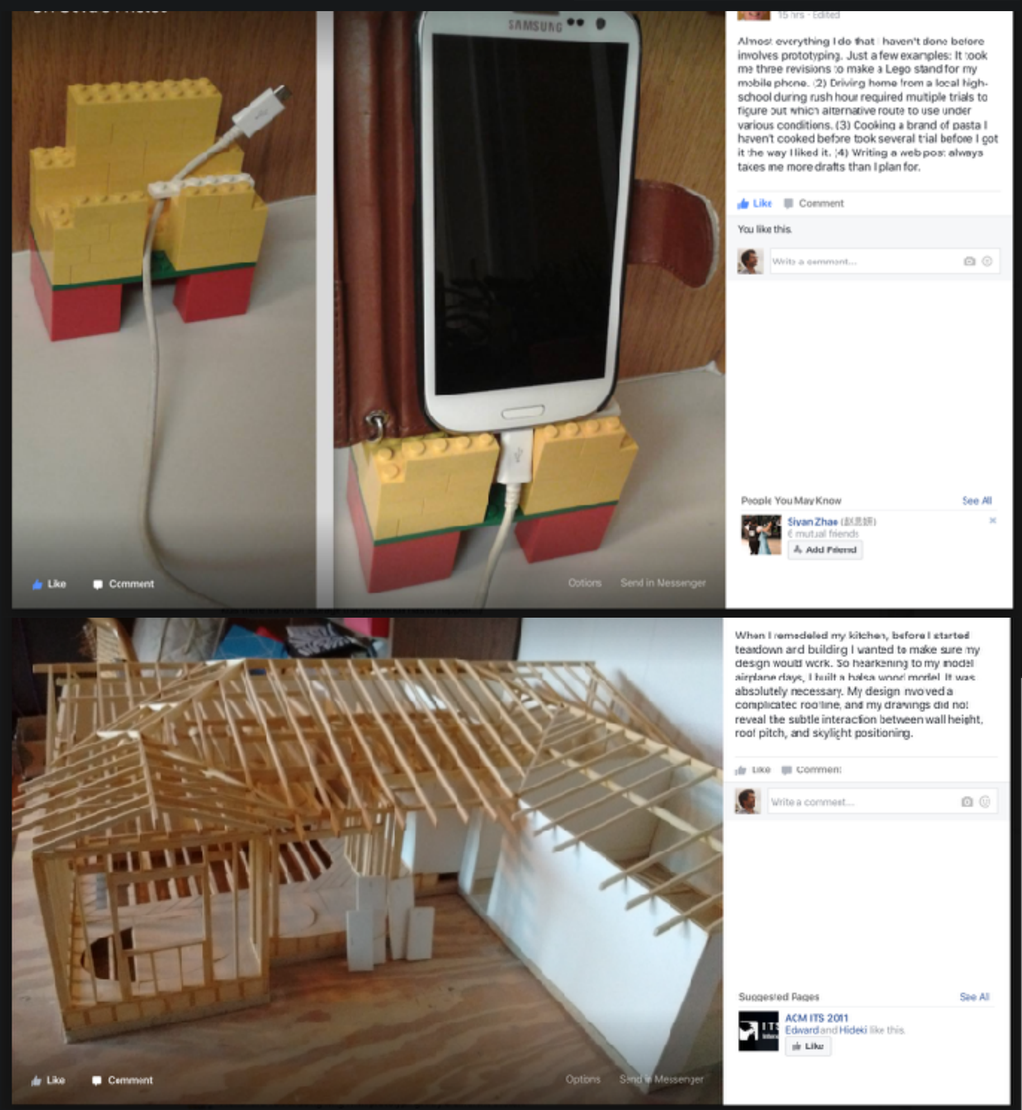
Sep 20 2016

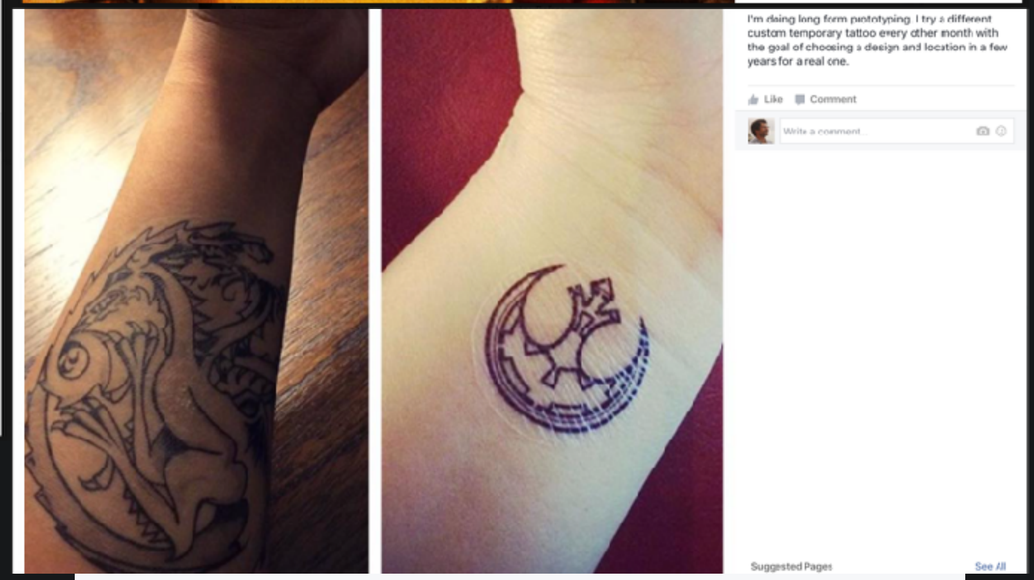
“Everyone designs who devises courses of action aimed at changing existing situations into preferred ones.”

– Herb Simon



Prototyping in real life





James Landay the prototype furnitur Johan Lundin Cooking and raising children is constant prototyping of methods
Like · Reply · 13 hrs

Maryanna Rogers Contracting as a prototype for working at a company full time

Maryanna Rogers Prototyped living with my man for 3 months (in Berlin) before moving in together 😃

Meenal Lele oh and I ikea hacked a standing desk. which taught me that i like sitting

Joined Big Sisters to prototype parenthood.

Participants picked their concept early

How can we help more people swim?

Norman & Klemmer (2014) *How design education must change*

Creating design principles

Experimentation Matters

Norman & Klemmer (2014) *How design education must change*

Thomke (2003)

Science wisdom:

"A year in the lab saves an hour in the library"

Design melds physical, digital, and social worlds

A
atoms

B
bits

C
culture

Beyond Being There

Hollan & Stornetta (1992) *Beyond Being There*

What is Design at Large? Three principles

- Traditional design makes things: teacups and brochures. With Design at Large, the thing is only a piece of the experience
- With traditional design, we don’t know what happens when it leaves the factory. With Design at Large, we do.
- With traditional design, the object stays the same forever. Design at Large is magic, creating always evolving prototypes.

- Because solutions live in the real-world,
  Start with observation (define your system)
- Because what people say is different than what people do,
  Use that observation to uncover/articulate the <u>real</u> problem
- Because no one person has all the knowledge,
  Engage diverse stakeholders (all the people)
- Because the first idea is rarely the best,
  Encourage wild ideas: brainstorm widely w/multidisciplinary teams
- Because there's no oracle for complex systems and solutions emerge from surprising places,
  Get real quick, test and iterate
- Because designs change the setting,
  Prototype fast and furious, trying ideas in real situations
- Because what we see depends on what we know,
  Embrace Practice based evidence & Evidence based practice

Announcements

- Quiz 2 next Tuesday

HEURISTIC EVALUATION

WHY AND HOW

Scott Klemmer
www.hci-class.org

Multiple ways to evaluate

Empirical	Assess with real users
Formal	Models and formulas to calculate measures
Automated	Software measures
Critique	Expertise and heuristic feedback

When to get design critique?

- **Before user testing.** Don't waste users on the small stuff. Critique can identify minor issues that can be resolved before testing, allowing users to focus on the big issues.
- **Before redesigning.** Don't throw out the baby with the bathwater. Critique can help you learn what works and what should change.
- **When you know there are problems, but you need evidence.** Perhaps you've received complaints from customers or found yourself stumbling around your own site. Critique can help you articulate problems and provide you with ammunition for redesign.
- **Before release.** Smooth off the rough edges.

http://**www.etre.com/usability/inspection**

Begin Review with a Clear Goal

Heuristic Evaluation

- Developed by Jakob Nielsen
- Helps find usability problems in a design
- Small set (3-5) of evaluators examine UI
  - independently check for compliance with usability principles (“heuristics”)
  - different evaluators will find different problems
  - evaluators only communicate afterwards
    - findings are then aggregated
- Can perform on working UI or sketches

Jakob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Ten Design Heuristics

- Show system status
- Familiar metaphors & language
- Control & freedom
- Consistency
- Error prevention
- Recognition over recall
- Flexibility & efficiency
- Aesthetic & minimalist design
- Recognize, diagnose, & recover from errors
- Help

My somewhat revised names for Jakob Nielsen's heuristics, http://www.useit.com/papers/heuristic

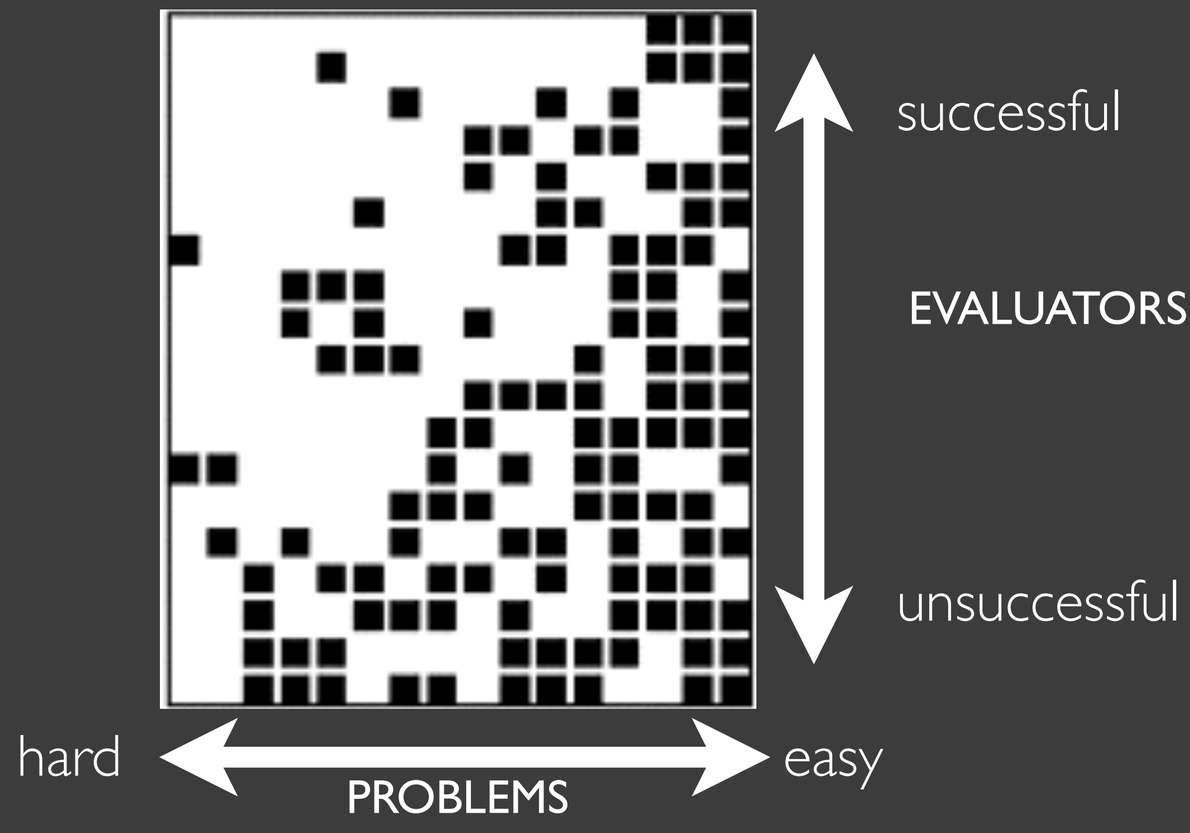
Evaluators' Process

- Step through design several times
  - Examine details, flow, and architecture
  - Consult list of usability principles
  - ..and anything else that comes to mind
- Which principles?
  - Nielsen's "heuristics"
  - **Category-specific heuristics** from e.g., design goals, competitive analysis, existing designs
- Use violations to redesign/fix problems

Jakob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Why Multiple Evaluators?

- No evaluator finds everything
- Some find more than others

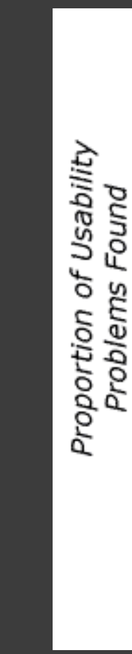


Jakob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

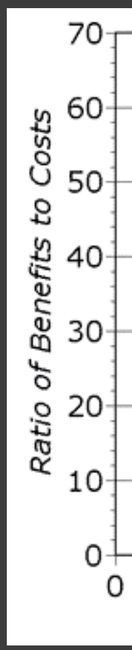
Decreasing Returns

- Caveat: graphs for a specific example

problems found



benefits / cost



Heuristic Eval: Cost-effective

- In one case: benefit-cost ratio of 48
  - estimated benefit $500,000; cost $10,500
  - value of each problem ~$15K
  - how might we calculate this value?
    - in-house -> productivity; open market -> sales
- Severe problems found more often
- Single evaluator achieves poor results
  - only finds 35% of usability problems
  - 5 evaluators find ~ 75% of problems

Jakob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Heuristics vs. User Testing

- Heuristic Evaluation often faster
  - 1-2 hours each evaluator
- HE results come pre-interpreted
- User testing is more accurate (by def.)
  - takes into account actual users and tasks
  - HE may miss problems & find "false positives"
- Valuable to alternate methods
  - find different problems
  - don't waste participants

Jakob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Phases of Heuristic Evaluation

1. **Pre-evaluation training:** give evaluators needed domain knowledge and information on the scenario
2. **Evaluation:** individuals evaluate and then aggregate results
3. **Severity rating:** determine how severe each problem is (priority). Can do first individually and then as a group
4. **Debriefing:** review with design team

Jakob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

How-to: Heuristic Evaluation

- At least two passes for each evaluator
  - first to get feel for flow and scope of system
  - second to focus on specific elements
- If system is walk-up-and-use or evaluators are domain experts, no assistance needed
  - otherwise might supply evaluators with scenarios
- Each evaluator produces list of problems
  - explain why with reference to heuristic or other information
  - be specific and list each problem separately

Jakob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

How-to: Heuristic Evaluation

- Why separate listings for each violation?
  - risk of repeating problematic aspect
  - may not be possible to fix all problems
- Where problems may be found
  - single location in UI
  - two or more locations that need to be compared
  - problem with overall structure of UI
  - something is missing
    - ambiguous with early prototypes; clarify in advance
    - sometimes features are implied by design docs and just haven't been "implemented" – relax on those

Jakob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Severity Rating

- Independently estimate after review
- Allocate resources to fix problems
- Estimate need for more usability efforts
- Severity combines
  - frequency
  - impact
  - persistence

Jakob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Severity Ratings

0 - don't agree that this is a usability problem

1 - cosmetic problem

2 - minor usability problem

3 - major usability problem; important to fix

4 - usability catastrophe; imperative to fix

Jakob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

Severity Ratings Example

- *Issue*: Unable to edit one's weight
- *Severity*: 2
- *Heuristics violated*: User control and freedom
- *Description*: when you open the app for the first time, you have to enter your weight, but you cannot update it. It could be useful if you mistyped your weight, or if one year or two after the first use of the app, your weight has changed.

Courtesy Sebastien Robaszkiewicz, cs147 2011

Debriefing

- Conduct with evaluators, observers, and development team members
- Discuss general characteristics of UI
- Suggest potential improvements to address major usability problems
- Dev. team rates effort to fix
- Brainstorm solutions

Jakob Nielsen, http://www.useit.com/papers/heuristic · Slide summary adapted from James Landay

http://designlab.ucsd.edu

Scott Klemmer

@DesignAtLarge