medium-fidelity prototyping, and usability testing

prototypes part II: medium and high fidelity prototyping;
choosing prototyping tools
medium and high fidelity prototyping

where else we’re covering it

by now W09 pre-reading – various
• tutorials – Axure, HTML, CSS

upcoming:
• last part of project part I (in planning for part II)
• project part II

today

• strategic overview
  • medium fidelity to high fidelity
  • iteration; prototype only what you need
• examples
  • from past 344/544 projects
  • demo of one type (what you can do with ppt)
• case studies + activity
learning goals

- list dimensions of prototyping fidelity and explain how these dimensions may vary;
- explain how these dimensions might differ in low to med to high fidelity prototypes, and give examples of when/why you may use each type
- make strategic choices about prototyping tools given your goals and constraints; be able to justify your choice.

what is a prototype?

prototypes take many forms:
- cardboard, foam, software, video,
- clay, paper, hidden people, website,
- sketches/skits, scripts, etc.

the point: **make ideas real**

4 designs: image-enhanced planner
types of prototypes

think of prototyping techniques as tools in your bag of tricks

• have lots so that you have appropriate one
• just like evaluation methods
• should be fast, effective and targeted to the issues
  ➞ don’t waste time implementing something that won’t teach you anything!

fidelity ranges from low to high

when to use different types of prototypes?

early design

Choose a representation
Rough out interface style
Task walkthrough & redesign
Fine tune interface, screen design
Heuristic evaluation and redesign
Usability testing and redesign
Limited field testing
Alpha/Beta tests

Low-fidelity prototypes
Medium-fidelity prototypes
High-fidelity prototypes / partially-working systems
Working systems

late design
**low fidelity prototypes**

meant to be rough, quick to build, easy to throw away

purposes
• proof of concept(s)
• rough (but flexible) interface design
• facilitate communication with users early on
  – can be useful for generating and narrowing requirements

**med to high fidelity prototypes**

increasing in completeness and detail:
• more aspects being prototyped at same time
• higher degree of functionality
• higher degree of polish
• etc. . .

fidelity is a spectrum
• not always a firm line between low/med or med/high
many dimensions of “fidelity”

what are ways a prototype can be ‘true to life’?

- **visual realism**: how real it looks. polish, graphic imagery
- **physical realism**: shape and form for 3D objects; feel
- **scope**: how many functions included; horizontal vs. vertical
- **functionality**: what actually works? e.g. web app: links live?
- **data**: operates on real vs. faked data
- **autonomy**: operates alone vs. requires “supervision”
- **platform**: interim vs. final implementation
- ???
matching game:
What medium makes most sense for each dimension?

Prototyping dimension:
• how real it looks (polish)
• **scope** how many functions included; horizontal vs. vertical
• real vs. faked functionality
  how much of it is faked?
• operates on real vs. faked data
• operates alone vs. requires “supervision”
• for 3D products: physical aspects, or just images?
• interim vs. final platform

prototyping medium:
• paper
• Balsamiq
• Axure
• PowerPoint
• HTML
• Java/Swing
• modeling foam & hot-melt glue
• Flash
• Visual Basic
• Photoshop
• Arduino
• found objects
• Python

what’s the difference between “low” and “medium”?

it has become confusing …

*used to be obvious! paper vs. nearly anything else.*

in last ~10 years: many powerful tools that:

1. **make it very easy** *(a low-fi trait)* to generate mockups
2. **look real** and are at least **somewhat interactive** *(usually a “medium fidelity” trait)*

  e.g.: balsamiq, axure – low or medium; usually not high

→ many prototyping platforms can be low or medium!
  *(depends on how you use it.)*

  but, FEW are all three (low to high)
what’s the difference between “medium” and “high”?

- fidelity is partly a matter of **completeness**: as you get more hi-fi: may become more restricted in platform (*you’re trying to do more at once*).

medium and high-fidelity prototyping
what can you use?

**many things:**
- drag-and-drop GUI toolkits for standard UI mockups
  - e.g. Axure, Visual Basic
- scripting languages & interface libraries for add’l flexibility
  - e.g. python, tcl/tk, java script libraries (e.g., jQuery)
- graphical languages for visualization & novel interface creation
  - VB, Java, Flash; Processing; d3;
- special purpose tools and environments
  - e.g. toolkits for integrating speech, haptics, I/O devices

→ a prototyping platform can be medium- OR hi-fi; depends on how you use it.
the situation today for prototyping tools
(vs. developing on final platform)

for simple prototyping:
• balsamiq, axure, html, powerpoint

other popular tools:
• InVision, Sketch, POP (digitizing paper prototypes) + lots more!

advanced UIs still require (scripting) language + libraries
• HTML + javascript
• Tool Command Language/Tool Kit (TCL/TK)
• Python
• Processing (Java based, but way more accessible; good for sketching, no good for larger code projects)
• still a need for C++, C#, Objective C, Java

live example: powerpoint

• link screenshots together (Windows, OSX)
• some additional functionality you can add, like with prototyping templates you can purchase for Keynote and PowerPoint (e.g. Keynotopia)
examples from past 344/444 projects

balsamiq: low to medium

Quickly **mock up** images and hyperlinked interactivity. But - real functionality difficult.
UBC Student Aid – past 344 project (html)

html:
final platform
didn’t need
to be glitzy
easy to copy
existing text, look and feel
then alter everything

Apply Online
To apply for a BC Student online, please click "start"

Start

Browser requirements
Ongoing site maintenance

Browser requirements
This website is best viewed using Microsoft Internet Explorer 3.3 or higher or Netscape 4.x or higher. Earlier versions of these browsers may cause difficulty in navigating StudentAid BC Online or completing the application forms and are not recommended.

If you could not access StudentAid BC Online, check your browser version by following these links:
- Click 'Help' on your menu bar, then click 'About' to find your browser version.
- To upgrade your version of Microsoft Internet Explorer, click here.

To upgrade your version of Netscape, click here.

Ongoing site maintenance

home alarm system

flash:
product for the home
needed to gauge reactions to having it in ones house
imagery + graphic resolution critical
e-reader & note-taking tool

Flex:

needed to test how well the concept worked for actually taking notes in lecture

highly functional
detailed vertical

sonic stage music synchronization tool

flash w/ imported photoshop

observe scanned, hand-drawn sketches
gorgeous
didn’t need to be

didn’t do well.
how do you know when you have – or need – a high-fi prototype?

- scope is complete (horizontal and vertical)
- prototype can be tested in just about every way performance as well as subjective and cognitive analysis; more realistic scenarios; in field
- feels like time to switch to final development platform
- design is becoming rigid and finalized.

case studies + activity

iterative prototyping in industry

for each of the following case studies, complete a worksheet page.

some high-level questions:
- what were the main goals of the prototype?
- what fidelity was the prototype?
- did the design team choose the correct tools for the job? what were the tradeoffs?
activity recap:

1. What investigative challenges do you need to answer?
2. What kind of evaluation should you do to answer that question?
3. What should a prototype to support that evaluation emphasize?
4. What prototyping tool might be a good choice?

most important lessons today:

1) it is COMPLICATED (slow, expensive) to prototype multiple dimensions at once.
   ⇒ so don’t. Instead: modularity of prototyping.

2) each prototyping tool has strengths and weaknesses
   - may make it better (more efficient and capable) for some of these prototyping dimensions than others.
   ⇒ you may need multiple tools throughout your design’s life cycle.
usability testing

by now (W09 pre-reading – RSP)
• usability testing

upcoming:
• project part II – usability evaluation of your medium fidelity prototype
learning goals

• explain when usability studies are typically conducted, why they’re conducted, and what you might try to learn
  – give examples of locations, tasks, metrics, evaluation methods that might be involved

• experience in analyzing a reported usability study

where do usability studies fit in?

HCI starts with understanding the problems that users are having

then designing a system that solves these problems

→ requirements, task examples specify what it should do
→ decide on conceptual/interface design for how system will do it

usability studies: see if we succeeded!
what is a usability study?

• we spent 1\textsuperscript{st} half of term on \textit{evaluation for understanding} the problems of the users.
• we’re now conceptualizing solutions; \textit{creating interface(s), refining with walkthroughs}.
• next we’ll refine, create medium-fidelity prototypes
• then: evaluate usability of that system
  \(\rightarrow\) \textit{how easy} it is for the user to \textit{get} the system to do what s/he needs it to do.

\begin{itemize}
  \item Understand USERS:
    \begin{itemize}
      \item who they are
      \item their key tasks
    \end{itemize}
  \item Understand DESIGN:
    \begin{itemize}
      \item design space and risks
      \item choose design approach
    \end{itemize}
  \item EXAMINE existing:
    \begin{itemize}
      \item user tasks & objectives
      \item contexts
      \item interfaces
    \end{itemize}
  \item EVALUATE w/:
    \begin{itemize}
      \item observation
      \item questionnaire
      \item task analysis
    \end{itemize}
  \item MAKE use of:
    \begin{itemize}
      \item requirements
      \item task analysis
      \item real & virtualized users
      \item technology options
      \item company IP
    \end{itemize}
  \item REFINE Design:
    \begin{itemize}
      \item by element
      \item considering task
      \item varied contexts
    \end{itemize}
  \item EVALUATE w/:
    \begin{itemize}
      \item usability testing – controlled, uncontrolled
      \item heuristic evaluation
    \end{itemize}
  \item CONFIRM & debug:
    \begin{itemize}
      \item performance in real use
    \end{itemize}
\end{itemize}

\begin{itemize}
  \item PRE DESIGN
    \begin{itemize}
      \item user and task descriptions
      \item design requirements
    \end{itemize}
  \item EARLY DESIGN
    \begin{itemize}
      \item low fidelity prototyping methods
      \item throw-away prototypes
      \item design direction
      \item risk analysis
    \end{itemize}
  \item MID DESIGN
    \begin{itemize}
      \item testable medium-fidelity prototypes
    \end{itemize}
  \item LATE DESIGN
    \begin{itemize}
      \item testable medium-fidelity prototypes
      \item alpha/beta systems or complete specification
    \end{itemize}
\end{itemize}

K MacLean - derived from version by Saul Greenberg (U Calgary)
how is usability testing different from other evaluations we’ve done?

purpose: test a relatively refined prototype and its support of task execution

evaluation goals: specific to your interface and needs, but generally need to capture things like breakdowns, performance, learning challenges, acceptance

task definitions and participant choice are crucial to ensure you’re covering the right ground

tools used: Usability testing makes use of observation, interviews, questionnaires – a large subset of our basic evaluation instruments.

metrics: count it.

evaluating to understand: often looking for qualitative insights stories, workflows, obstacles, dependencies, missing links.

testing for usability: performance dominated by how long, and how many mistakes.

where and how:

evaluating to understand: good to do in natural environment
usability testing: to obtain experiment control, often done in lab
biggest difference: experimental material

Usability testing requires a refined interface.

- This could be… the medium-fidelity prototype you’ll create
- Or it could be the bad old interface, which you plan to revise or replace
  i.e. might be one part of "evaluate for understanding"
- Contrast this with the primary focus of ‘evaluation for understanding’: task modeling, needs requirements, etc.

when designing a usability test:

choice of methods: triangulate
  - typically: one instrument counts something, while another interprets what was counted

choice of metrics: driven by your requirements
  - as well as basic usability principles

how many users:
  - should be representative of your user groups
    - e.g. if you want to support both expert and novice users, should have good numbers of both!
  - Within a demographic, <4-5 is dubious; >10-12 is of marginal additional value.
  - Sometimes constraints dictate low numbers.
    - If you have to generalize, consider who your test users are, and how representative they are?
how many users?

context

generally: experimenter specifies the task
can be:

- at quite low level; e.g. the subtask that will take you from one screen to the next.
- or, at entire task level: see if someone can figure it out, start to finish, and watch /count / measure the challenges s/he has

not done with those task examples yet!

- can use them as a basis for a stripped-down task description much as you did for cognitive walkthroughs
  (but don’t usually want to include the story)
example usability study

https://youtu.be/QckIzHC99Xc?t=56s

example usability study script
metrics

examples of common ones

time:
- to complete a task (entire, or a portion)
- learn a task
- resume a task after interruption
- find something on a screen
- attain specified degree of proficiency
- etc.

errors:
- number per task or unit of time, in navigation, in selection, in interpretation;
- number of users making the error;
- alternately: number of successes
- etc.

events of interest:
- page views or clicks,
- access of particular tools
- timeouts
- questions asked or help tools consulted
- # users willing to recommend
- etc.

subjective factors:
- task level satisfaction
- perception of aesthetics
- perceived ease of use
- perceived preference
- etc.

(can be measured on a Likert or semantic rating scale)
ICICS usability lab
located in X7

activity
analyze a documented usability study
usability testing
in your project

evaluation goals
- you will likely want to draw from your requirements and task examples; may need to prioritize;
- test *how well* your system supports what you intended it to
- metrics, evaluation methods, etc. should follow

medium fidelity prototype scope
- prototype won’t be a complete working system
- it should do just enough to test if your design will meet your goals (and be achievable in the time available)

alternatives to usability testing

Usability testing generally requires users.

- “discount” methods can also target refined prototypes and be done without users:
  - heuristic evaluation
  - cognitive walkthrough

- because you don’t need users . . .
  - can do it first! (before a usability study)
  - possible to apply these methods yourself while iterating on a design (before it’s totally finished)
on your own: plan a usability study

Imagine you’re running a usability study of Kobo.com (or other site or application)
Kobo is a website for buying e-books, and managing purchases

1. what design stage are you in?
2. what would be good evaluation goals?
   - think about: why would you be doing this study (“finished” site)
3. what would be a good study task? (for this evaluation)
4. what are good metrics for that task?
5. what are evaluation type and data recording methods to collect those metrics?

usability study – one example

Design Stage: mid or late (refined design); or PRE design (check existing before initiate a re-design)
Evaluation goal: how easy is it easy for users to complete purchases?
Task: buying an e-book on Kobo.
Metrics: time to complete a transaction; # of errors, backtracks, etc

for an in-lab, in person usability test, you might do:
observation (video recorded), analyzed by counting events of interest (errors, dead ends, can’t find something); AND
a structured post-interview.

to administer remotely:
replace observation with web clickthrough logging; AND
a online post-questionnaire.