Human Computer Interaction

Dr. Nivin Atef
Nivin.atef@cis.asu.edu.eg

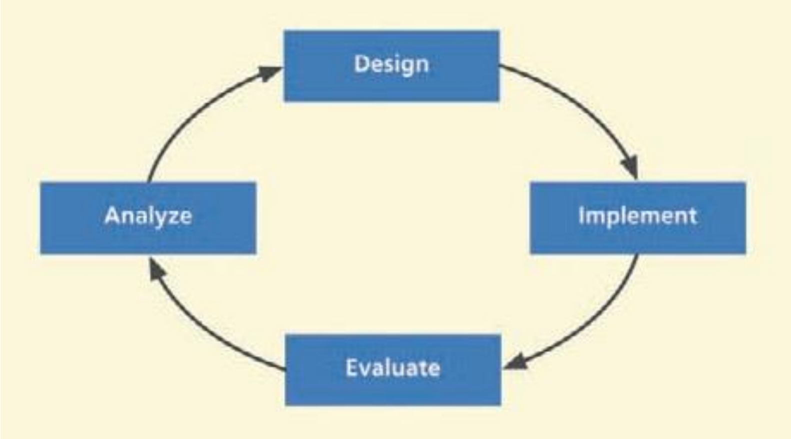
User Centered Design (Recap)

- **Analysis** translates to understanding user **work** and **needs**.

- **Design** translates to creating **conceptual design** and determining **interaction behavior** and **look** and **feel**.

-Implementation translates to **prototyping**

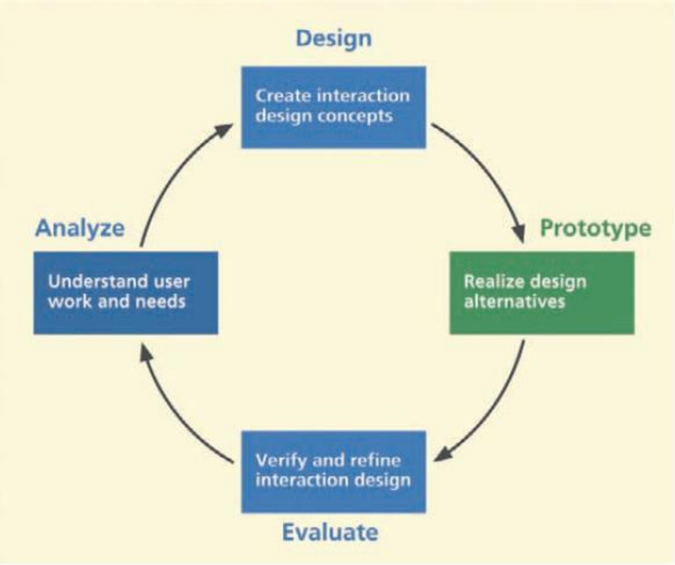
- **Evaluation** translates to ways to see if our design is on **track** to **meet** user **needs** and **requirements**.



Prototyping

User Centered Design - Prototyping

- We build room for **several iterations** into our **design process**, and we do it by making the early iterations as **cheap** as possible.
- Corresponds to the **cost** of the iteration step – or, equivalently, its **fidelity** or **accuracy**.
- An **early implementation** might be a **paper sketch** or **mockup**.
- Only a **pale shadow** of what it would look and behave like as interactive software. But it's **incredibly cheap** to make, and we can **evaluate** it by showing it to **users** and **asking** them **questions** about it.



Design
Create interaction design concepts
Analyze
Understand user work and needs
Prototype
Realize design alternatives
Verify and refine interaction design
Evaluate

Why Prototype?

- Prototypes are much **faster** to **build** than finished implementations
  - Get feedback earlier, cheaper

- If we have a design **decision** that is hard to **resolve**, we can **build** and **experiment** with **alternatives**

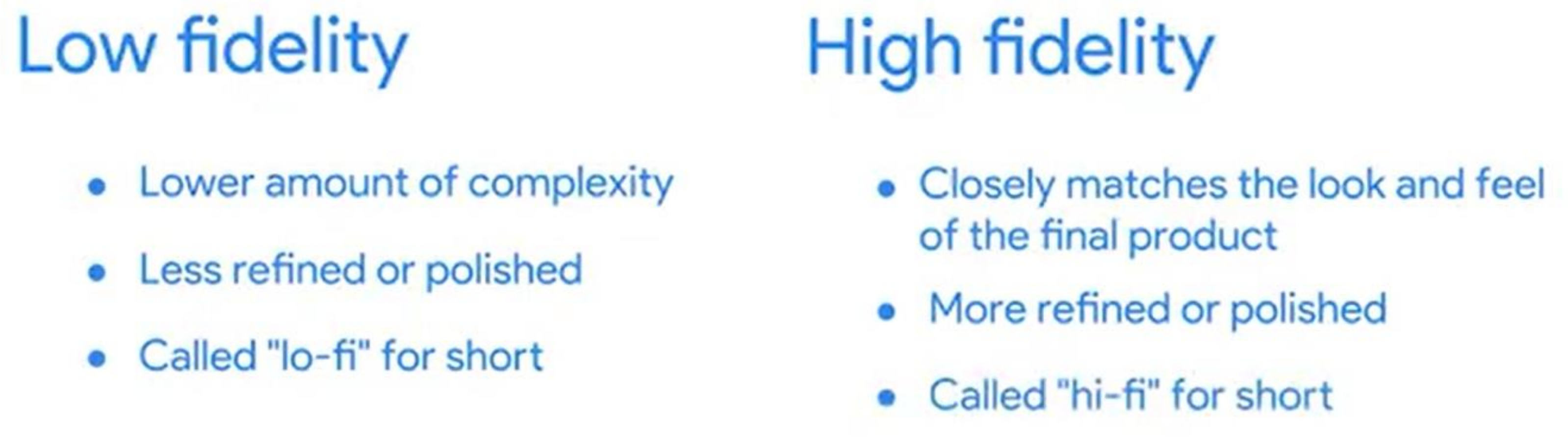
- If we discover **problems** in the design, a prototype is **easier** to **change** or **throw away.**

Prototype Fidelity

- In UX, **fidelity** means **how closely** a design **matches** the **look-and-feel** of the **final product.**

- Low fidelity: omits details, use cheaper materials
- High fidelity: more like finished product

Prototype Fidelity (Cont.)



Low fidelity

- Lower amount of complexity
- Less refined or polished
- Called "lo-fi" for short

High fidelity

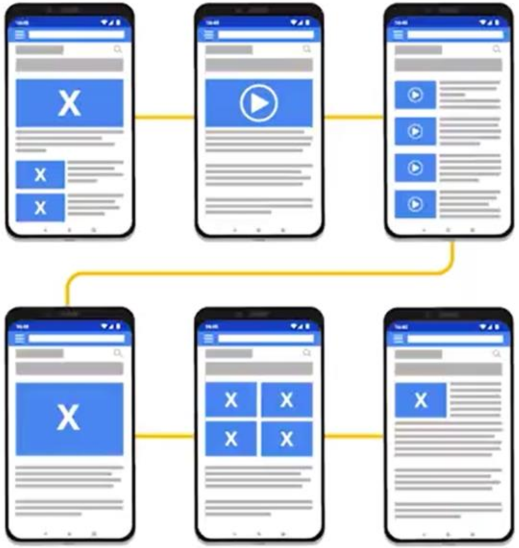
- Closely matches the look and feel of the final product
- More refined or polished
- Called "hi-fi" for short

Low Fidelity Prototypes

- Low fidelity prototypes include the **most basic content** and **visuals** and are usually static (not interactive).
- They are often used to help map out the **shell** of the **interface**, its **screens** and basic **information architecture**.
- Usually serve as a **checkpoint** for the product team and stakeholders at the **beginning** of the design process.
- They help teams **visualize** and **test early concepts**, **requirements** and design assumptions at the beginning of a web design project.

Low Fidelity Prototypes - Wireframes

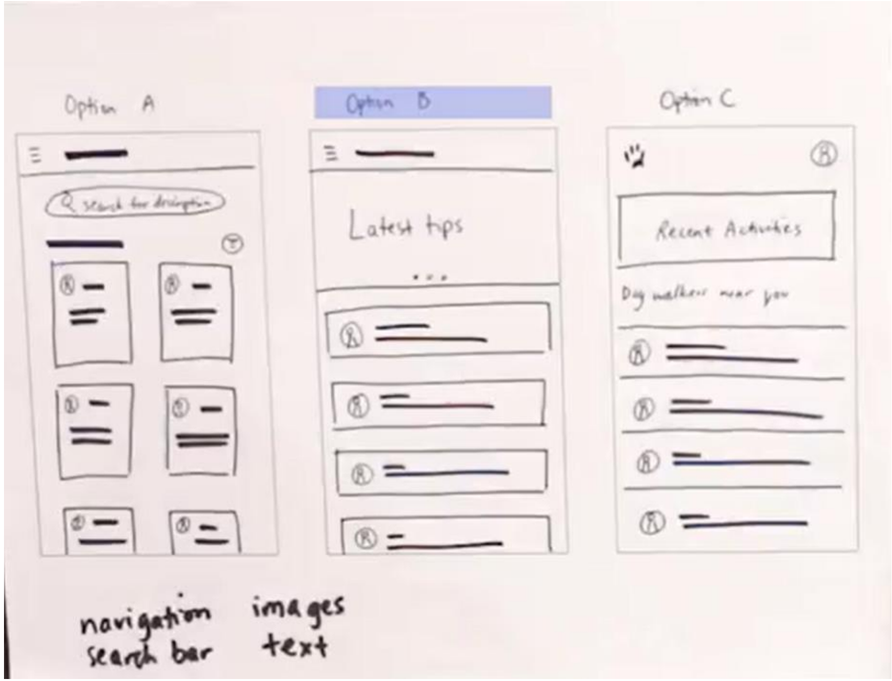
- A wireframe is a **basic outline** of a **digital experience** (an app or a website)
- Mostly **lines** and **shapes** with some **text**.
- You can create wireframes by **hand** or by using **digital tools**.
- We usually start creating wireframes by **drawing on a piece of paper.**

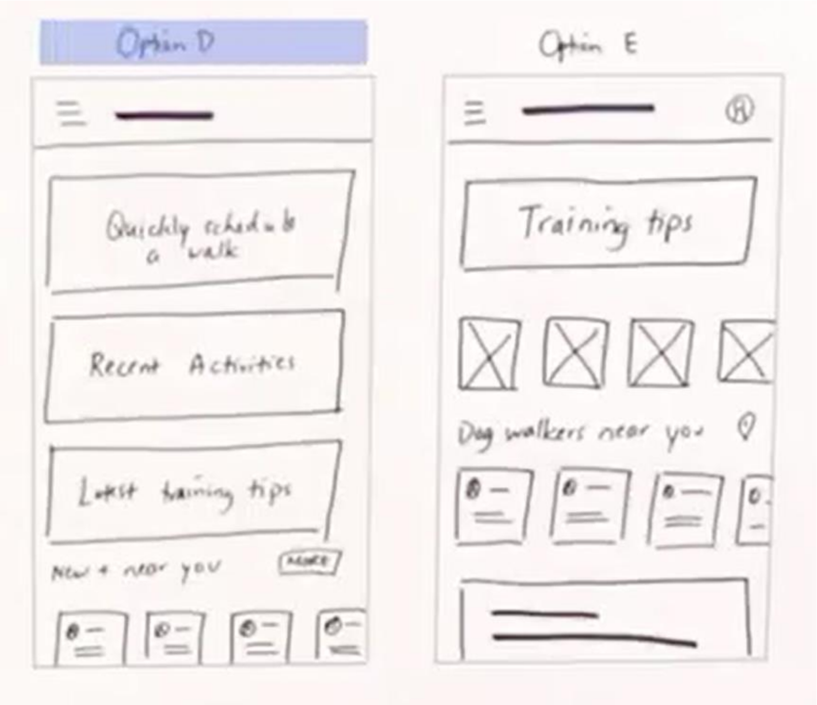


Why Wireframes?

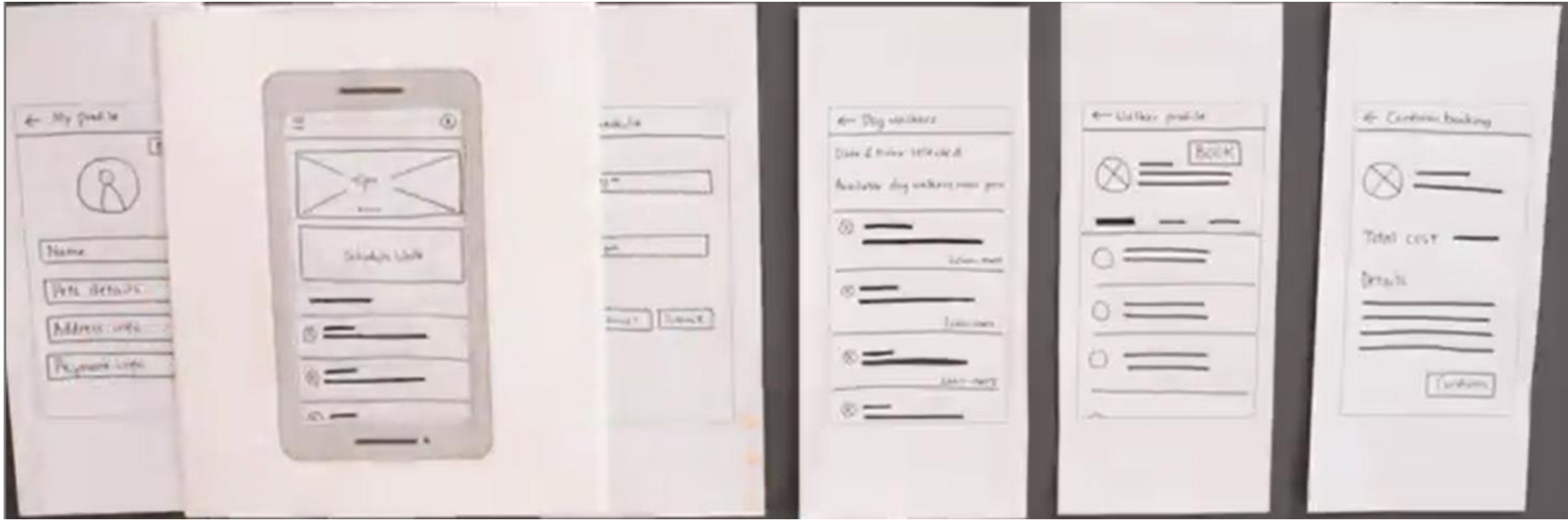
- Wireframes **establish** the **basic structure** of a page **before** any **visual** considerations (like color or images) are added.
- Wireframes serve as an **outline** to get the **team** on the **same page** early in the project.
- Wireframes **highlight** the **intended function** of the product.
- Wireframes help designers **save time** and **resources**. Wireframes allow the team to quickly try out **different design options**.

Example


Option A
Option B
Option C
search for description
Latest tips
...
Recent Activities
Dog walkers near you
navigation
search bar
images
text


Option D
Option E
Quickly schedule a walk
Recent Activities
Latest training tips
New + near you
MORE
Training tips
Dog walkers near you

Example

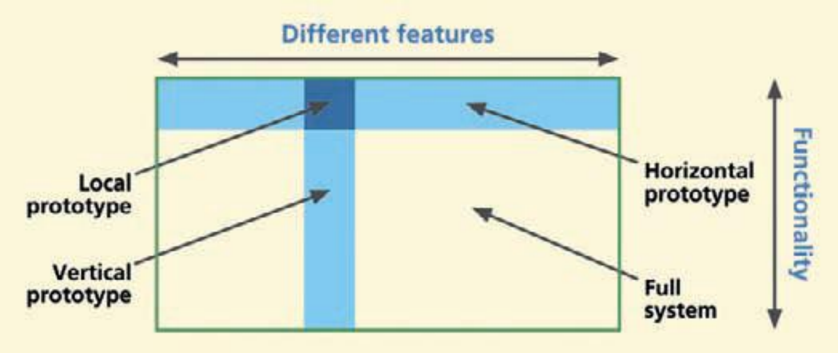

My profile
Name
Pets details
Address info
Payment info
Schedule Walk
Dog walkers
Date & time selected
Available dog walkers near you
Walker profile
BOOK
Confirm booking
Total cost
Details
Confirm

High Fidelity Prototypes

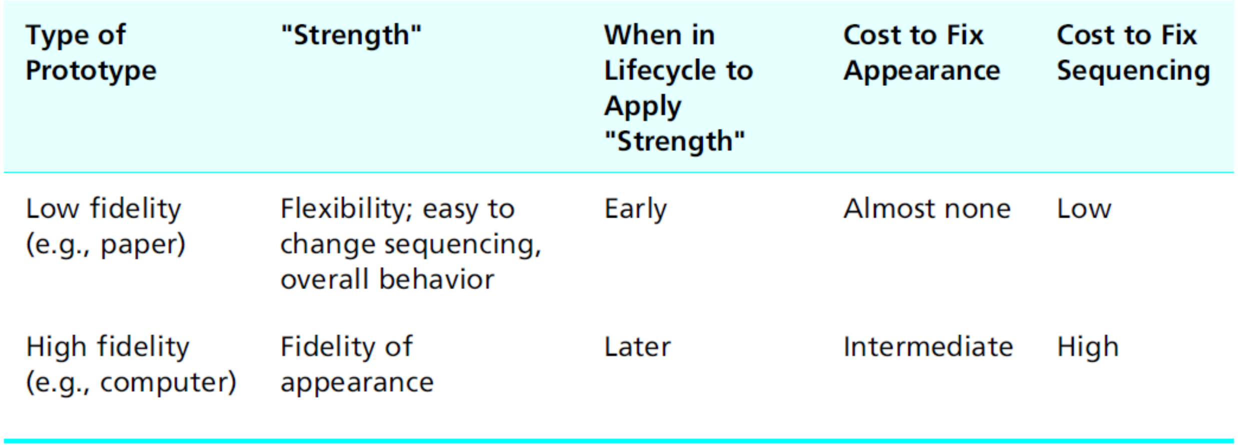
- High fidelity prototypes are **more complete representations** of the **end product** than low fidelity wireframes.
- Many are **clickable** and **respond** to the user’s actions, mimicking authentic interface interaction.
- Their **content** are **more precise** as designers concentrate on refining the **graphics**, **spacing** and **layout**.
- They are often built in the **advanced stages** of the **design process** to **communicate design decisions** to the **development team** prior to coding the final product.

Fidelity is Multidimensional

- **Breadth**: fraction of features covered
  - Only enough features for certain tasks
- **Depth**: degree of functionality
  - refers to **how deeply** each feature is actually **implemented**.
  - Limited choices, canned responses, no error handling



Low fidelity vs High fidelity



Type of Prototype	"Strength"	When in Lifecycle to Apply "Strength"	Cost to Fix Appearance	Cost to Fix Sequencing
Low fidelity (e.g., paper)	Flexibility; easy to change sequencing, overall behavior	Early	Almost none	Low
High fidelity (e.g., computer)	Fidelity of appearance	Later	Intermediate	High

Summary of comparison of low-fidelity and high-fidelity prototypes

Prototyping

- Paper prototypes
- Computer prototypes
- Wizard of Oz prototypes

Paper Prototype

- **Paper prototypes** are an excellent choice for **early design iterations**.
- Interactive paper mockup
  - **Sketches** of **screen appearance**
  - **Paper pieces** show windows, menus, dialog boxes

- Interaction is natural
  - Pointing with a finger = mouse click
  - Writing = typing

- A **person simulates** the computer's operation
  - Putting down & picking up pieces
  - Writing responses on the "screen"
  - Describing effects that are hard to show on paper

- **Low** fidelity in **look & feel**

Why Paper Prototyping?

- **Inexpensive**; all you need is a **pen** and **paper**.
- **Faster to build**
  - Sketching is faster than programming

- **Easier to change, allow for rapid iteration.**
  - Easy to make changes between user tests, or even *during* a user test
  - No code investment– everything will be thrown away (except the design)

- **Focuses** attention on **big picture**
  - Designer doesn't waste time on details
  - Customer makes more creative suggestions

- **No special skills** are required. So graphic designers, usability specialists, and even users can help
- Building paper prototypes is a **collaborative activity**

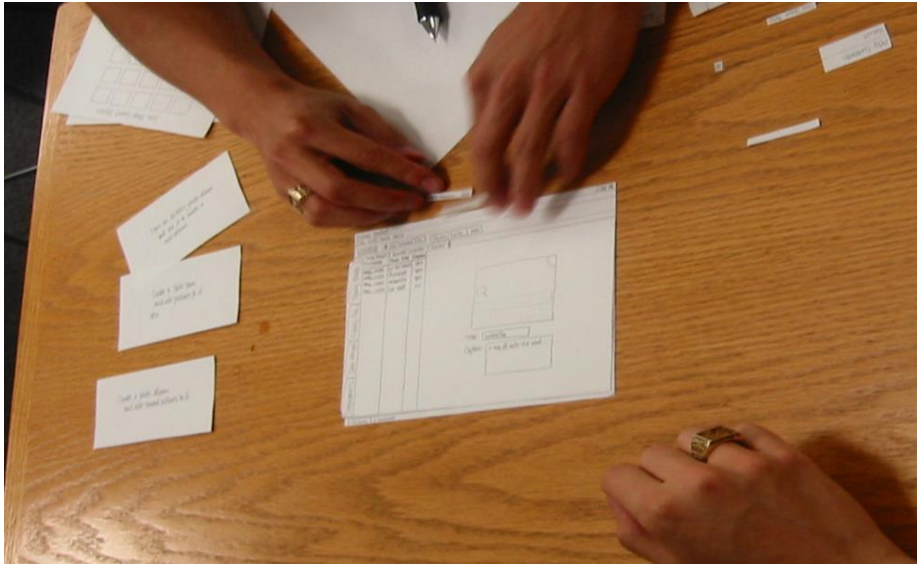
Tools for Paper Prototyping

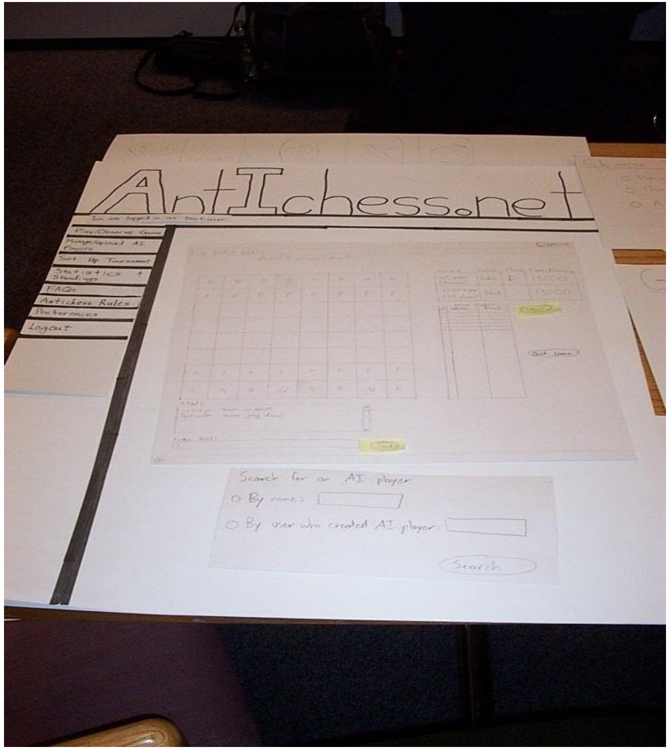
- White poster board (11”x14”)
    - For background, window frame
- Big (unlined) index cards (4”x6”, 5”x8”)
    - For menus, window contents, and dialog boxes
- Restickable glue
    - For keeping pieces fixed
- White correction tape
    - For text fields, checkboxes, short messages
- Overhead transparencies
    - For highlighting, user “typing”
- Photocopier
    - For making multiple blanks
- Pens & markers, scissors, tape

Tips for Good Paper Prototypes

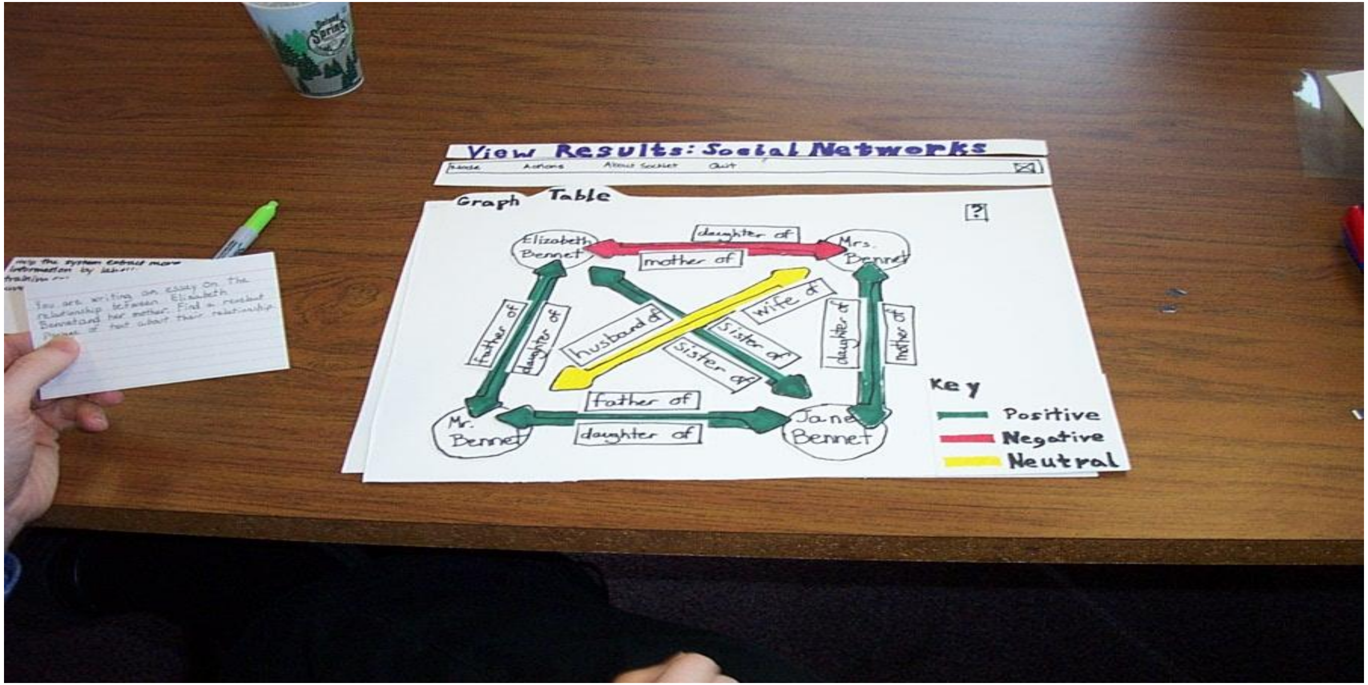
- Make it **larger** than life
- Make it **monochrome**
- Replace tricky visual feedback with audible descriptions
    - Tooltips, drag & drop, animation, progress bar
- Keep pieces organized
    - Use folders & open envelopes

Size Matters





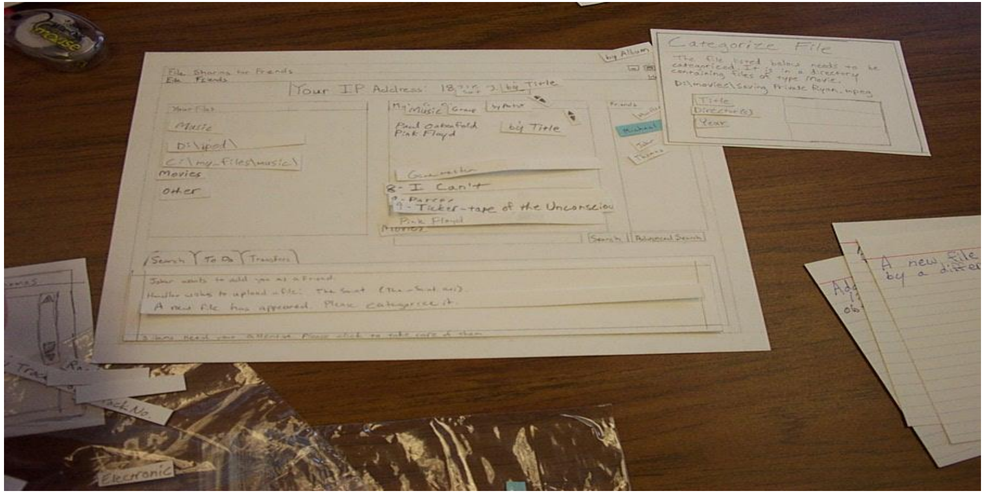
The Importance of Writing Big and Dark

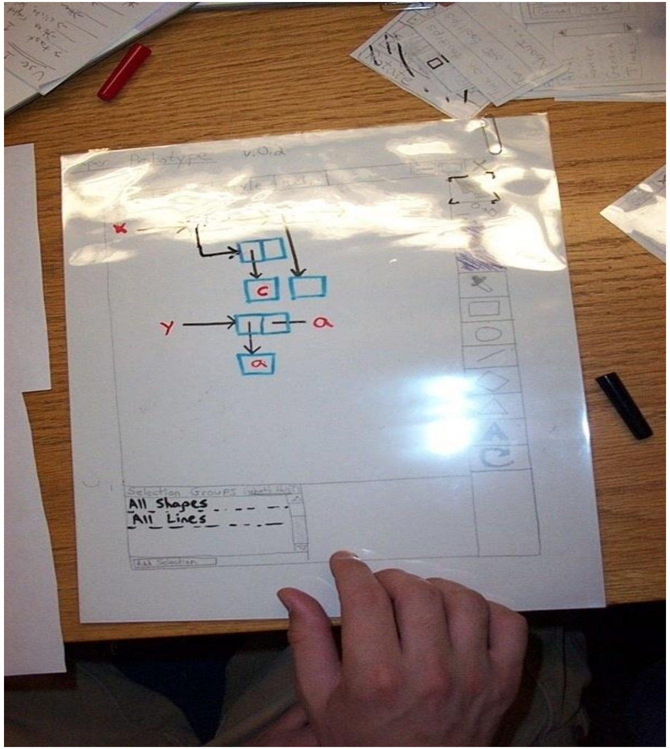


Help the system extract more information by label...
training ...

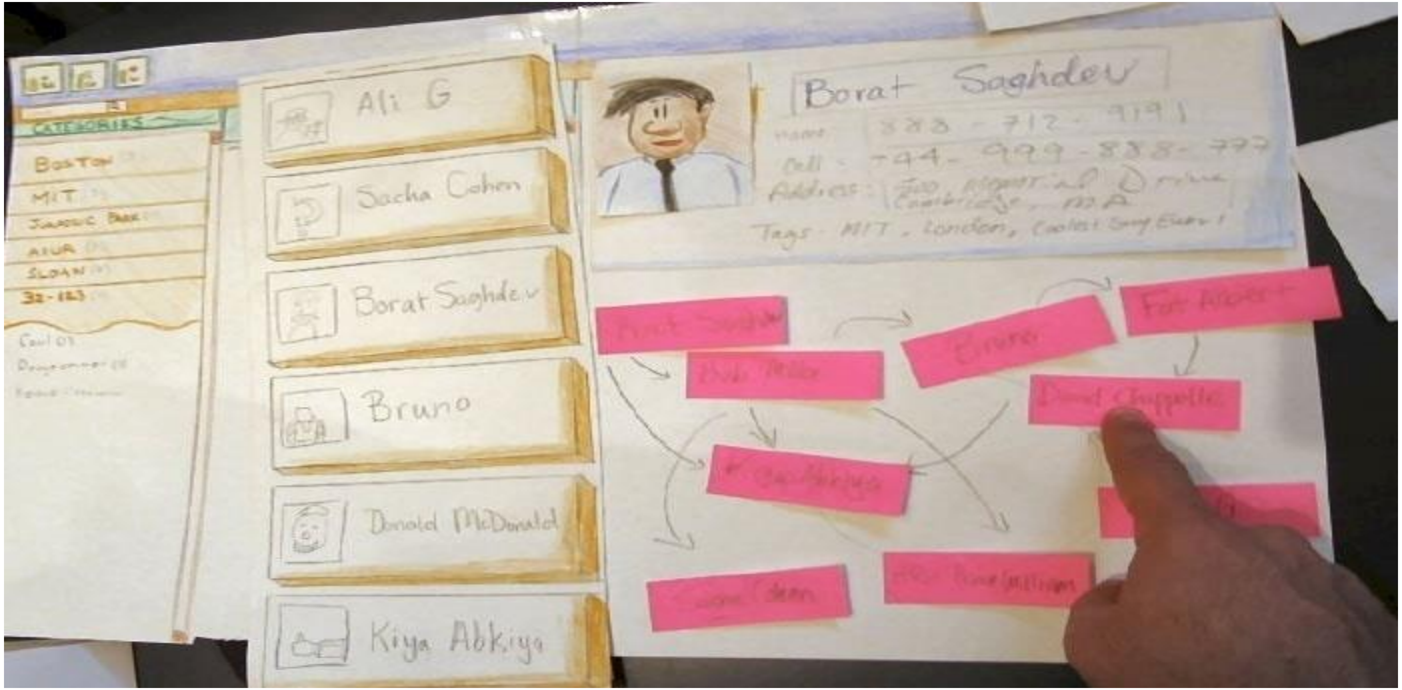
You are writing an essay on the relationship between Elizabeth Bennet and her mother. Find a passage of text about their relationship

Post-it Glue and Transparencies are Good



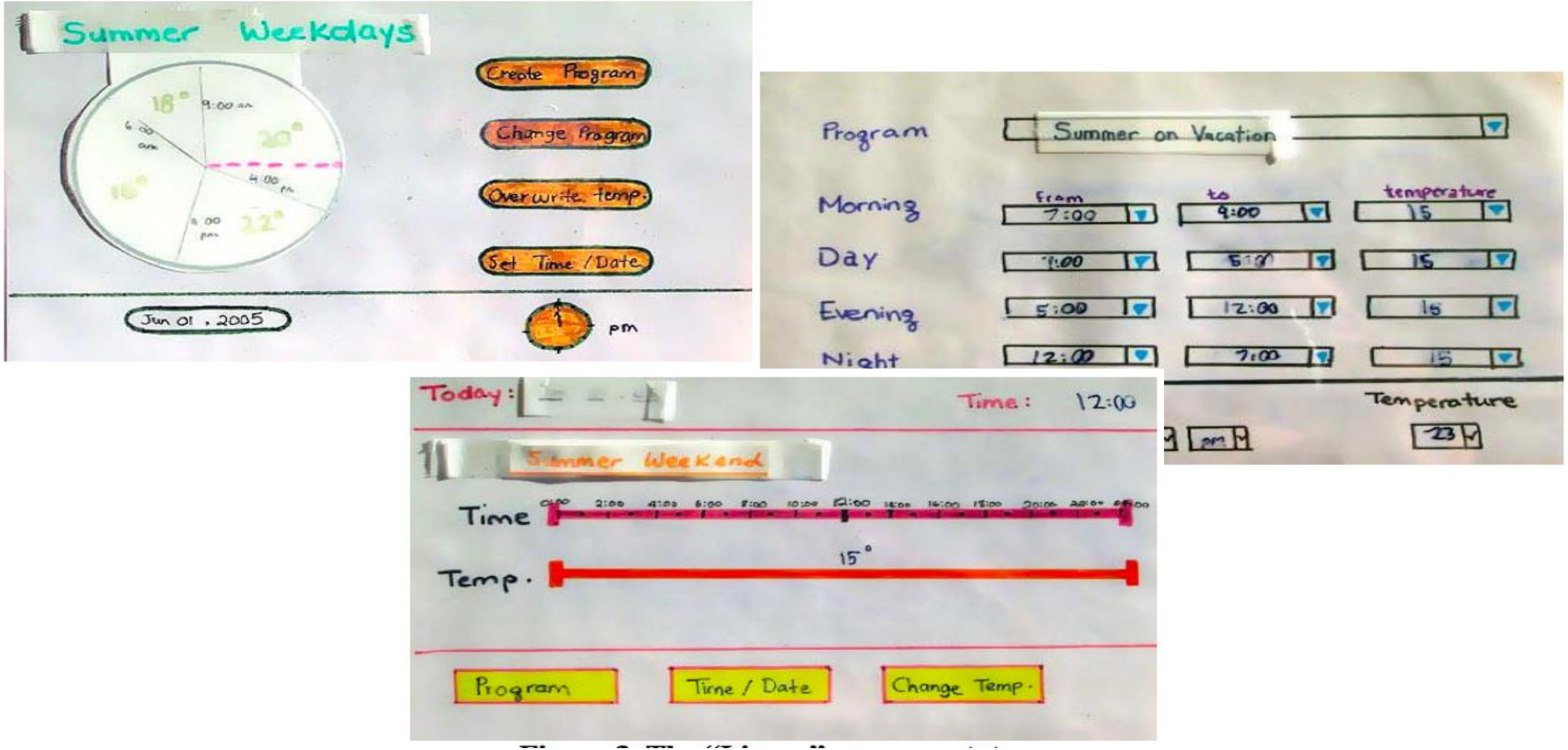


Paper Prototypes



Low-Fidelity Prototypes Aren't Always Paper

Multiple Alternatives Generate Better Feedback



What You Can Learn from a Paper Prototype

- **Metaphor** or **Conceptual** model
  - Do users **understand** it?
- Functionality
  - Does it do what's needed? **Missing features**?
- Navigation & task flow
  - Can users find their way around?
  - Are information preconditions met?
- Terminology
  - Do users understand **labels**?
- Screen contents
  - What needs to go on the screen?

What You Can’t Learn

- Look: color, font, whitespace, etc
- Feel: efficiency issues
- Response time
- Are small changes noticed?
    - Even the tiniest change to a paper prototype is clearly visible to user
    - Ex: cursor change, highlight change, message down in the status bar.
- Exploration vs. deliberation
    - Users are more deliberate with a paper prototype; they don’t explore as much

Disadvantages of Paper Prototypes

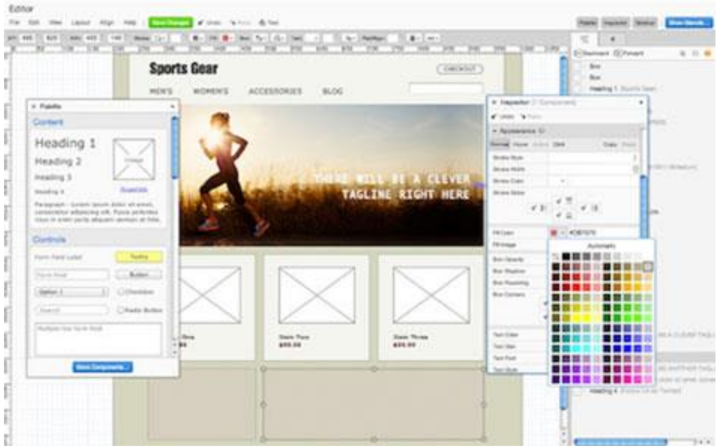
- It can be **hard** to **interpret** a paper prototype.
- It **requires** a lot of **imagination**.
- You need to bring the paper prototype to the **testing location**, and a person has to act as a computer to manually change the design in real time.
- Paper prototypes are **difficult** to create with a **remote team.**

Prototyping

- Paper prototypes
- Computer prototypes
- Wizard of Oz prototypes

Computer Prototype

- Interactive software simulation
- High-fidelity in look & feel
- Low-fidelity in depth
  - Paper prototype had a human simulating the backend; computer prototype doesn’t
  - Computer prototype may be **horizontal**: covers most features, but no backend



What You Can Learn From Computer Prototypes

- Everything you learn from a paper prototype, plus:
- **Screen layout**
  - Is it clear, overwhelming, distracting, complicated?
  - Can users find important elements?

- Colors, fonts, icons, other elements
  - Well-chosen?

- **Interactive feedback**
  - Do users notice & respond to status bar messages, cursor changes, other feedback

- **Efficiency issues**
  - Controls big enough? Too close together? Scrolling list is too long?

Why Use Prototyping Tools?

- Faster than coding
- No debugging
- Easier to change or throw away
- Don’t let your UI toolkit do your graphic design

Prototyping

- Paper prototypes
- Computer prototypes
- Wizard of Oz prototypes

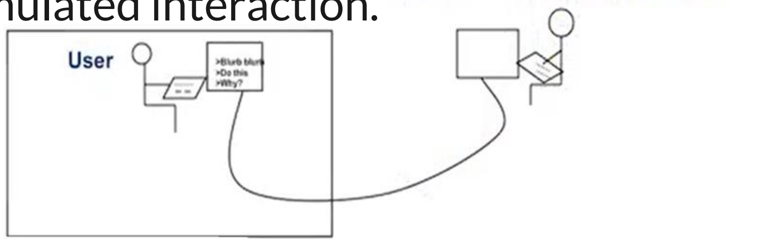
Wizard of Oz Prototype

- “Wizard of Oz” = “man behind the curtain”
- The “Wizard of Oz prototyping” technique is a **deceptively** simple approach to the **appearance** of a **high degree of interactivity** and **highly flexible prototype behavior** in **complex situations** where user inputs are **unpredictable**.

- The setup requires **two connected computers**, each in a different room.

- Useful when your design **ideas** are still **wide open** and you **want** to see **how users behave naturally** in the course of simulated interaction.

- Ex: Social Search Engine



Wizard of Oz Prototype

- Make an interactive application without (much) code
  - Front end interface
  - (remote) wizard controls user interface
  - Makes sense when it’s faster/cheaper/easier than making a real thing

- Often used to simulate future technology
  - Speech recognition
  - Learning

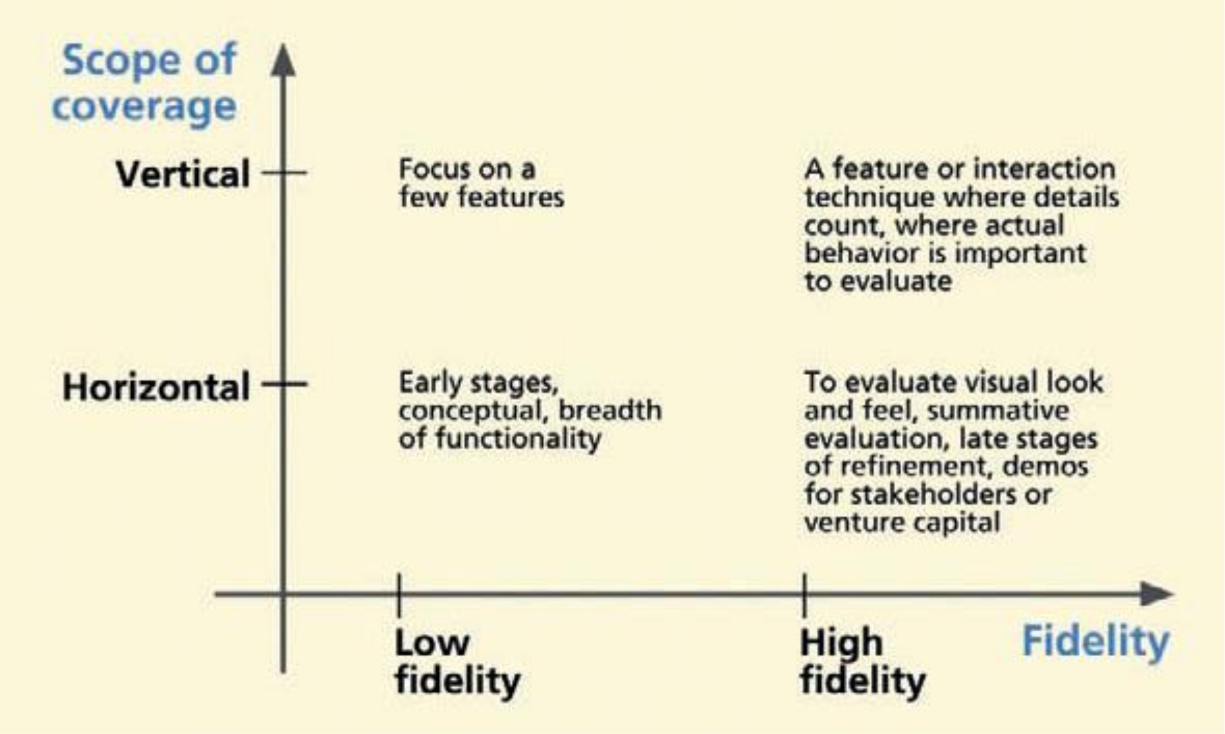
- Get feedback from people

  Hi-fidelity user think its more real

  Low-fidelity: more license to suggest changes

- Issues
  - Two UIs to worry about: user’s and wizard’s

Choosing the Right Breadth, Depth, level of fidelity, and amount of interactivity


Scope of coverage
Vertical
Horizontal
Focus on a few features
A feature or interaction technique where details count, where actual behavior is important to evaluate
Early stages, conceptual, breadth of functionality
To evaluate visual look and feel, summative evaluation, late stages of refinement, demos for stakeholders or venture capital
Low fidelity
High fidelity
Fidelity

Depth, breadth, and fidelity considerations in choosing a type of prototype.

Summary

- Prototype fidelity
  - Depth, breadth, look, feel
- Kinds of prototypes
  - Paper
  - Computer: storyboard, forms
  - Wizard of Oz
- Don’t get attached to a prototype
  - Because it may need to be thrown away

Sources

- https://www.coursera.org/learn/wireframes-low-fidelity-prototypes/home
- User Interface Design and Implementation MIT Open Course Ware ocw.mit.edu
- The UX Book: Process and Guidelines for Ensuring a Quality User Experience, REX HARTSON & PARDHA S. PYLA, Elsevier.

Thank You