Introduction To Prototyping

What is Prototyping?

- Prototyping is:
  - "The rapid creation of an approximation to a design idea for the purpose of retrieving feedback and knowledge."

What is Prototyping?

- We prototype to:
  - 1. Gain insights into user behavior.
  - 2. Communicate ideas to other teammates and stakeholders.
  - 3. Collect data for arguing the best design choice.

Example: IDEO Digital Camera

No Computation on device!
No Pictures (No lens, etc.)

Lessons from IDEO

- Prototypes are nearly ALWAYS incomplete.
- Goal is to SIMULATE specific aspects of the design and acquire knowledge regarding these targeted aspects.
- IDEO wanted to know more about the digital aspects of the camera.

More About Prototyping

- ***'Known Unknowns'***: Aspects of a design that you know you don't understand and wish to learn.
  - i.e., "Which color scheme is most user friendly?"
  - i.e., "Should this be a mobile or desktop application?"

- ***'Unknown Unknowns'***: Aspects of a design that you don't know are open issues.
  - i.e., "Why can't old people use this tiny screen!?"

Prototyping Process

- Prototyping is NOT a process in which you haphazardly create designs that you believe are various levels of “awesome”.

- Prototyping is about defining questions regarding your designs, and building something that answers those questions.

Prototyping Process

- 1. What are your goals for the prototype? What do you wish to learn?
- 2. How can you measure whether or not that goal has been achieved? How can you measure which of multiple prototypes is superior?
- 3. What is the MINIMUM amount of work necessary to produce, measure, and learn from your prototype?

1. Prototyping Goals

- If you perform your user and task analysis well, then these are represented by your **usability goals and requirements**.

- Thus we can select the usability requirements (probably a subset) that we wish to test with a given prototype.

1. Prototyping Goals

- Example:
- **UR**: *User must be able to type text messages at a minimum of 50 wpm while making less than 3 errors on average.*

- What do you wish to learn?
  - What physical interface is best for supporting users that wish to text and type on their phone?
  - Prototype choices:
    - Physical keypad
    - Touchscreen
    - Others?

2. Measuring Goals

- We talked about the falsifiability of usability requirements.

- If requirements are truly falsifiable, then we should be able to design user studies that prove (based on data) that our prototype is sufficient.

2. Measuring Goals

- Example:
- **UR**: *User must be able to type text messages at a maximum of 50 wpm while making less than 3 errors on average.*

- How to measure this:
  - Pretty simple, give users different phone prototypes and see how well they type.
  - In reality, a bit more complicated than this, but a good start.

3. Minimum Work Necessary

- What is the absolute minimal working prototype with which we can perform our user study?

- Strip out every feature that is not absolutely necessary.

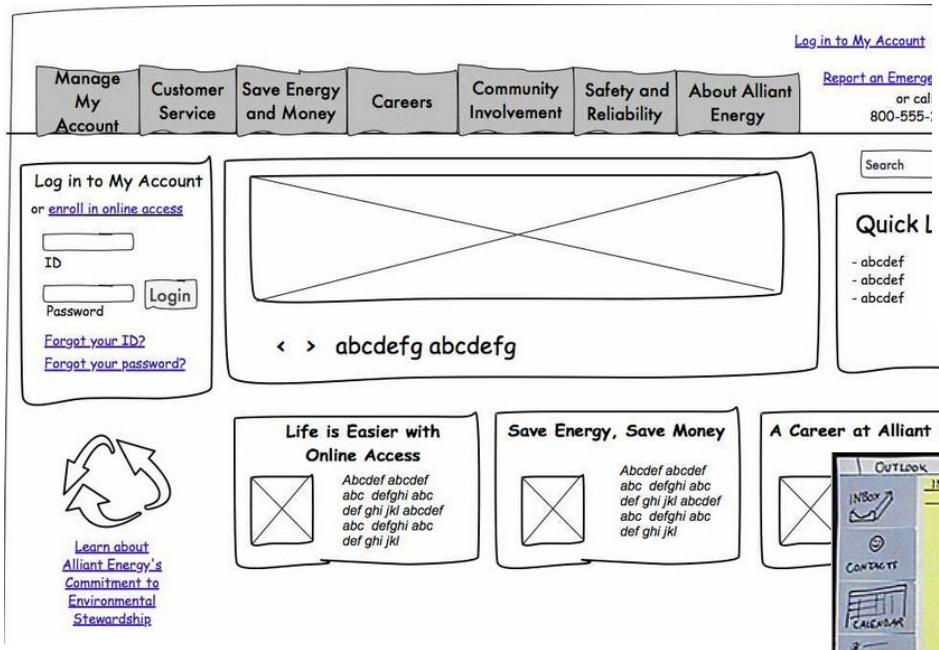
3. Minimum Work Necessary

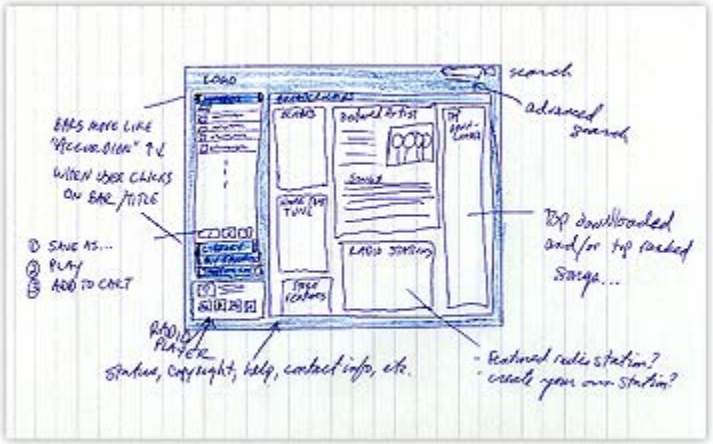
- Example:
- **UR**: *User must be able to type text messages at a maximum of 50 wpm while making less than 3 errors on average.*

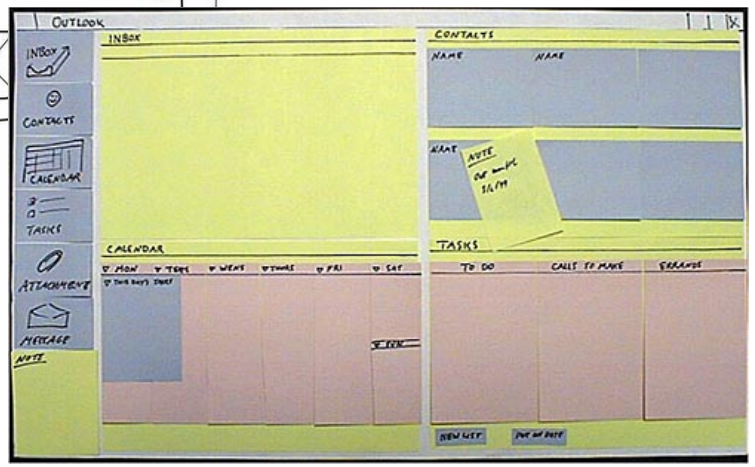
- Minimal:
  - Physical devices that can be typed on, need to be able to record speed and errors while typing.
- Unnecessary features:
  - Sending texts to other phones
  - Selecting contacts to send to
  - Etc.

Types of Prototypes

- Low-Fidelity:

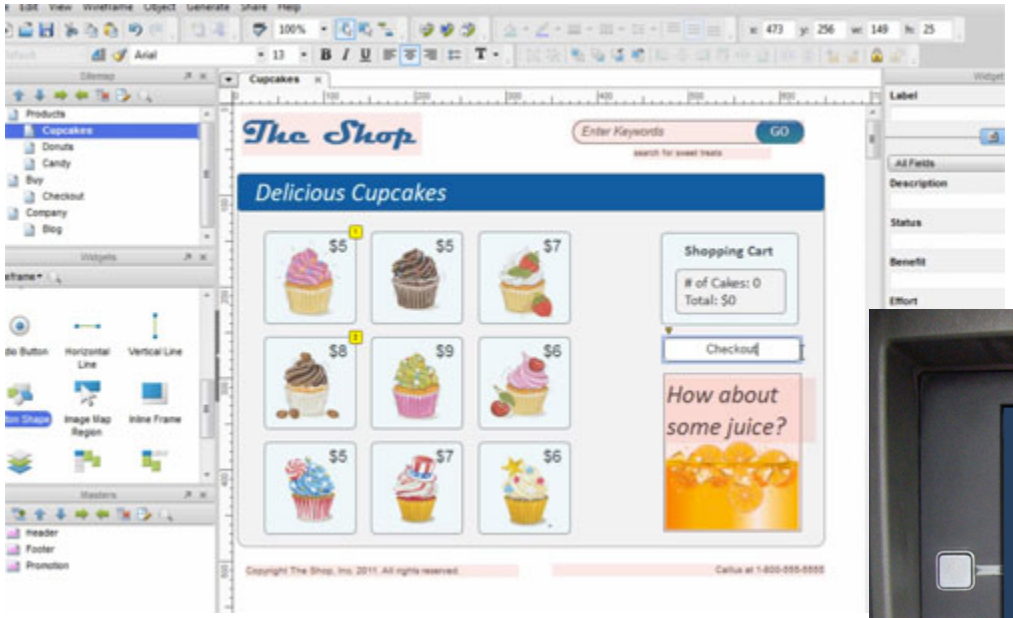






Types of Prototypes

- High-Fidelity:



Types of Prototypes:

- ***Feel***
  - What does it look and feel like?
- ***Implementation***
  - How does it work?
- ***Role***
  - What is the experience like?

Prototyping Rules

- Prototypes should NOT be required to be complete.
- Prototypes should be easy to change.
- Prototypes should be disposable.
  - Or, at least, you should mentally be prepared to dispose them.

Prototypes Can be BIG!

- Walter Teague

Can be small

- Jeff Hawkins, inventor of the Palm Pilot, launched his ground-breaking, mobile computing product in the mid-1990s with just a small block of wood as his guide.

Jeff Hawkins' Block Of Wood

- Did NOT learn about:
  - Battery Life
  - Interface and other digital interactions
  - Sound effects, feedback, etc.

- DID learn about:
  - Form Factor
  - Living with the device 24/7

Cost of Change Over Time

- Prototypes are MORE expensive to change the more high-fidelity they become.
  - SO BE CAREFUL!

- The most common estimate is that it's **100 times cheaper** to make a change before any code has been written than it is to wait until after the implementation is complete [Jakob Nielsen].

- Twenty years of usability engineering experience uniformly indicates that the **biggest improvements in user experience come from gathering usability data as early as possible** in a design project [Jakob Nielsen].

A Lot of Prototypes is GOOD!

Prototyping Strategies

- A few we will look at:
  - Paper Prototyping
  - Digital Mockups
  - Video Prototypes
  - Wizard of Oz Prototypes

Exam 1

- Interaction/interface design (ch 1)
- Conceptual models (ch2)
  - Problem space
  - metaphor
- Understanding users (ch 1 and 10)
  - Roles
  - Personae
  - relationships
- Task analysis (ch 10)
- Usability “laws”
- Interface types (ch 6)
- Visual design
- Emotional interaction (ch 5)
- Prototyping