CMPT 363: User Interface Design

Spring 2021

Week 4: User-Centered Design, Sketching, & Prototyping

Instructor: Victor Cheung, PhD

School of Computing Science, Simon Fraser University

© Victor Cheung, 2021

Recap from Last Lecture

- Usability Testing (cont’d)
  - Ranking severity, what to do with the results
- User-centered Design
  - Useful in every steps of the design process
- Sketching & Prototyping
  - Levels of fidelity
    - LFPs (we’ll continue with the rest in the next lecture)

- How would you create an LFP of the LED light designs without the use of any electronics?

Today

- Different levels of prototyping
  - LFPs (previous lecture part)
  - Medium Fidelity Prototypes (MFPs)
  - High Fidelity Prototypes (HFPs)

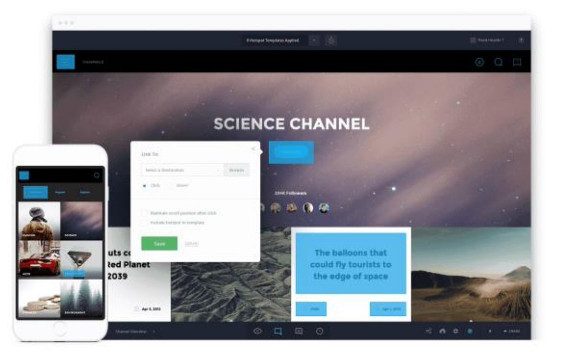
- Reminder: contact your teammates if you are recently put to a team!

More on Prototyping

Continue from last lecture (on LFPs)

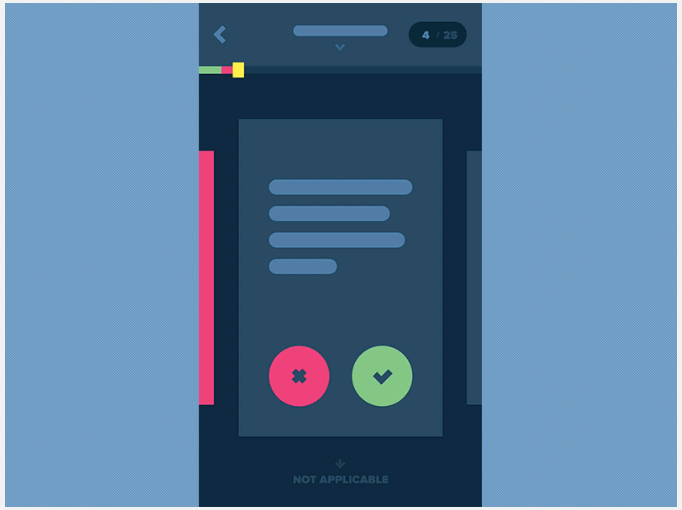
Medium-Fidelity Prototypes (MFPs)

- Put ideas into similar medium (pun-not-intended) as the final design
  - For computer interfaces, usually the device or simulator
  - For products, usually materials with similar texture, weight, and quality
- Implement some but not all features
  - For example, a walkthrough of a specific scenario (e.g., buying a ticket for a specific movie in a cinema website)
  - Incorporate automation instead of doing things manually
    - E.g., scripted simulations, interface builders, wizard of oz
- Typically narrowing down to a few versions



Scripted Simulations

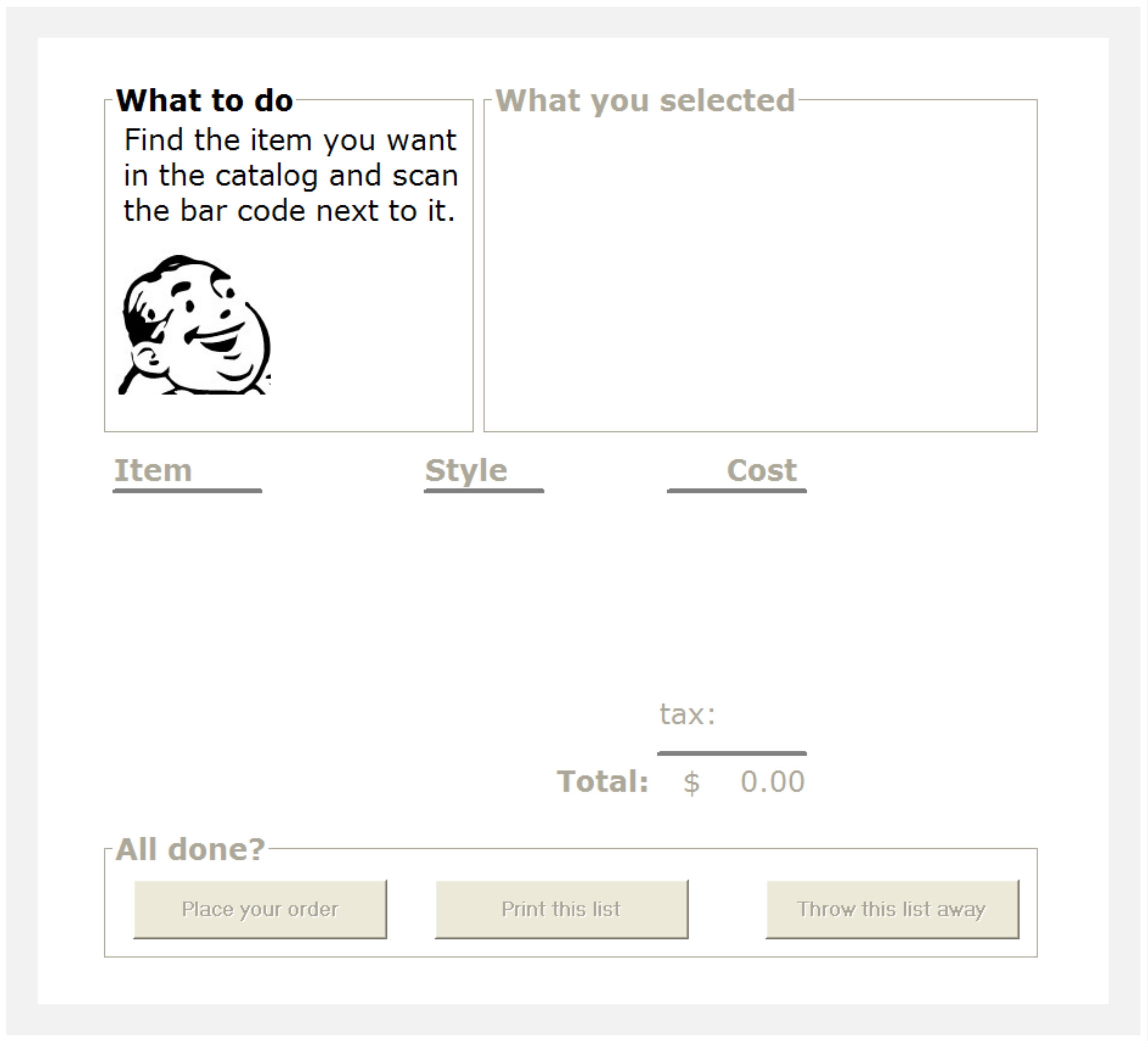
- Create a storyboard with media tools
  - Interface controls & content better resemble the real thing (e.g., actual photos or images in the descriptions)
- Provide interactivity so interface responds to user input
  - Scene/screen transitions activated by simple user inputs (e.g., a mouse click, typing a few letters)
- User given a tight script/task to follow
  - When followed the prototype would behave like a working system
  - Might not see anything (or unresponsive) if deviated



Source: https://dribbble.com/shots/2926913--4-Compound-Questionnaire

Scripted Simulations Example

- A visual shopping interface
  - Imagine a shopping app with which you can go to a store or read through its catalog, scan the bar code, and collect all when you leave
  - (like Amazon Go & IKEA app)



What to do

Find the item you want in the catalog and scan the bar code next to it.

What you selected

Item Style Cost

tax:

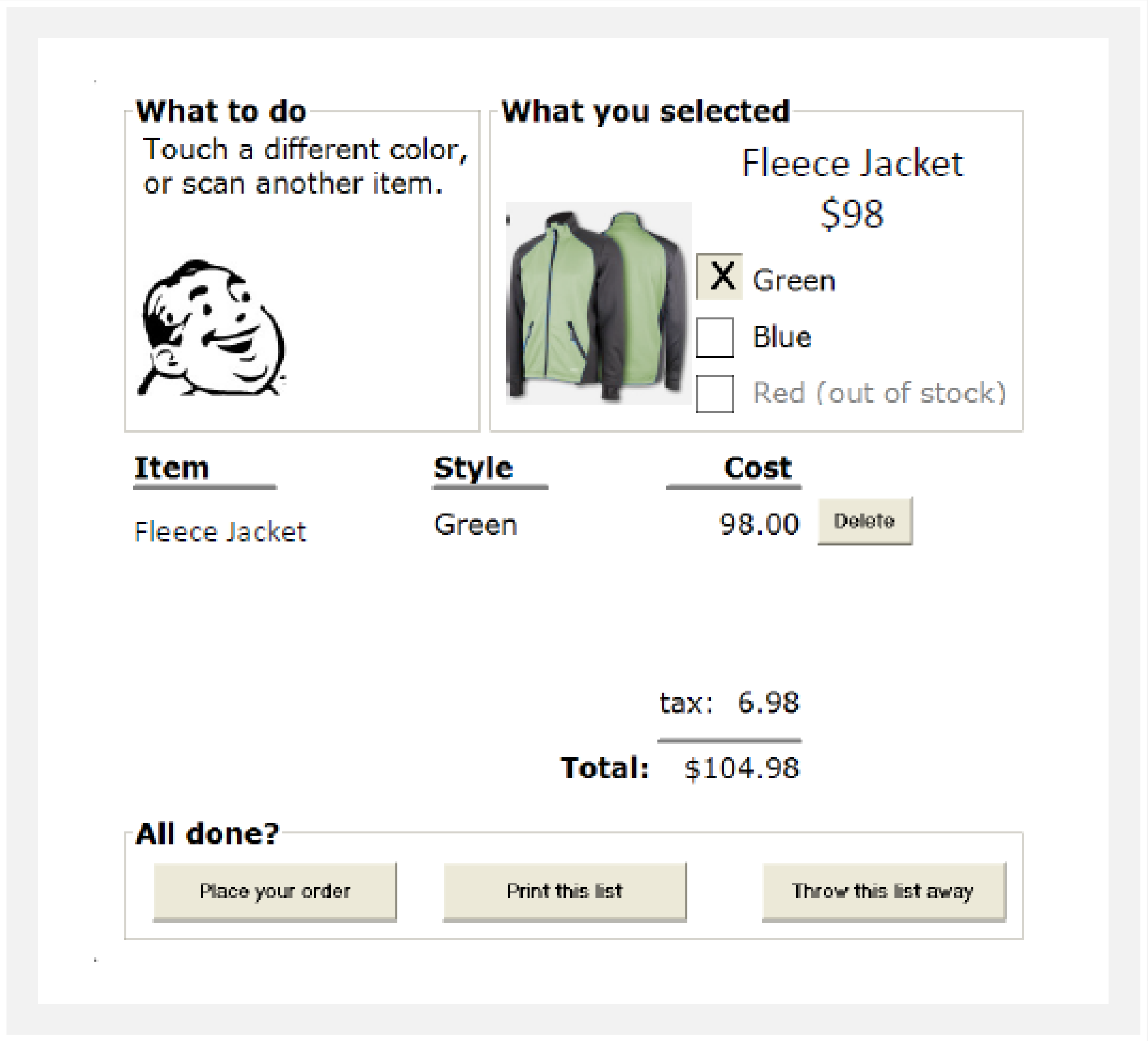
Total: $ 0.00

All done?

Slides by Mark Hancock

Scripted Simulations Example

- Create a script where the user has to scan a fleece item
  - Scanning other items might not work!
- The interface shows that the scanned fleece item has been added to the cart
- Next step of the script could be to select a different colour



What to do

Touch a different color, or scan another item.

What you selected

Fleece Jacket

$98

☒ Green

□ Blue

□ Red (out of stock)

Item	Style	Cost
Fleece Jacket	Green	98.00

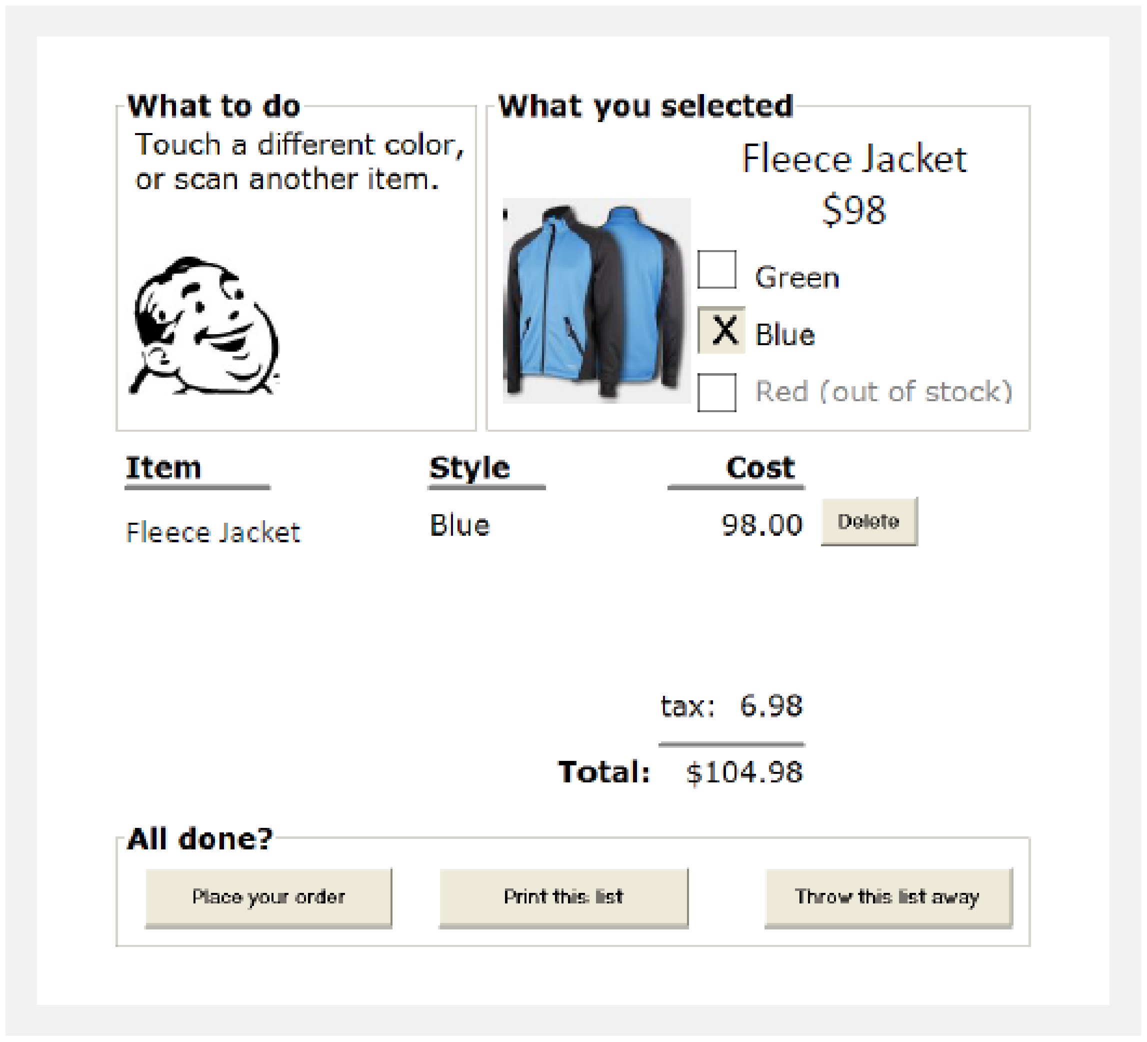
tax: 6.98

Total: $104.98

All done?

Scripted Simulations Example

- As the user selects the colour Blue (scripted), the interface reflects the action
- Next step of the script could be to place their order
- Other buttons might not function!



What to do

Touch a different color, or scan another item.

What you selected

Fleece Jacket

$98

☐ Green

☒ Blue

☐ Red (out of stock)

Item	Style	Cost	
Fleece Jacket	Blue	98.00	Delete

tax: 6.98

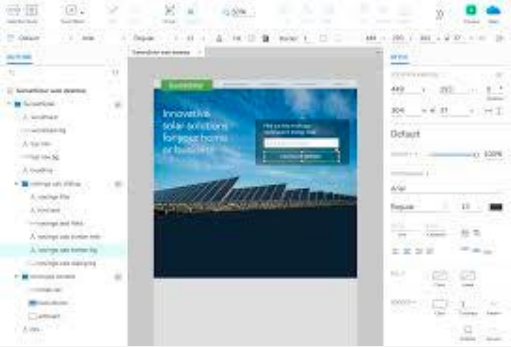
Total: $104.98

All done?

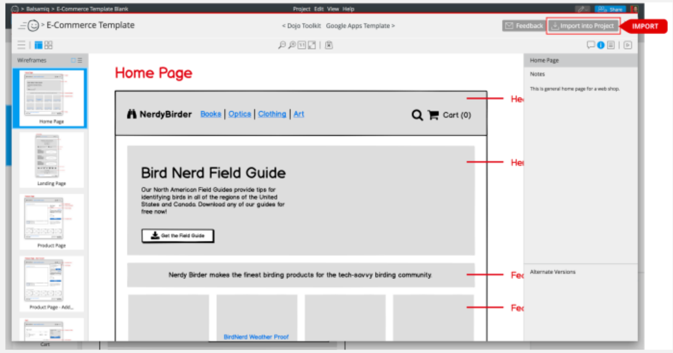
Slides by Mark Hancock

Interface Builders

- For computer interfaces (e.g., desktop, mobile, wearable) there are many online modern design tools for laying out common interfaces elements and widgets
  - E.g., Adobe XD, Balsamiq, InVision, Axure, Sketch, Figma…etc
- Excellent for showing look & feel in a polished manner, but usually constrained to the widget library



Axure

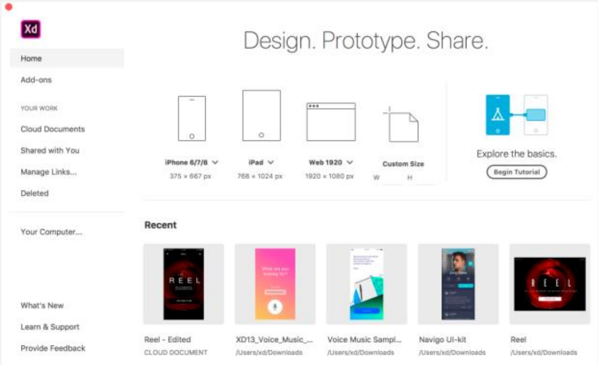


Bird Nerd Field Guide

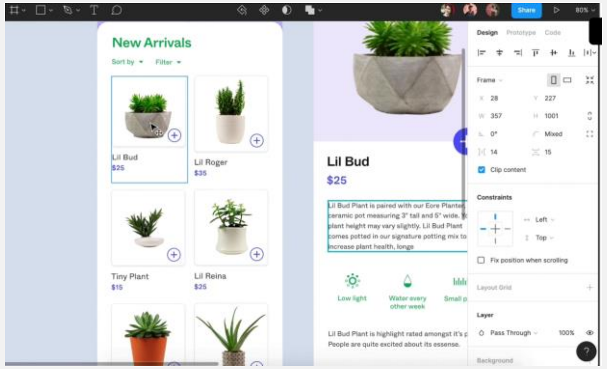
Our North American Field Guides provide tips for identifying birds in all of the regions of the United States and Canada. Download any of our guides for free now!

Nerdy Birder makes the finest birding products for the tech-savvy birding community.

Balsamiq



Adobe XD



Lil Bud

$25

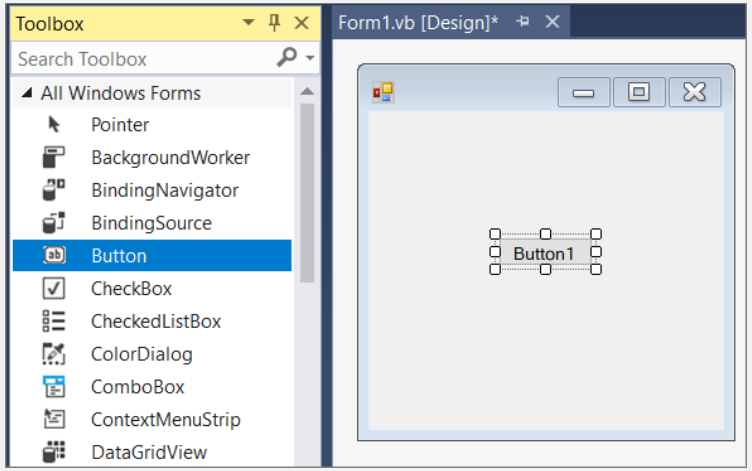
Lil Bud Plant is paired with our Eora Planter, ceramic pot measuring 3" tall and 5" wide. The plant height may vary slightly. Lil Bud Plant comes potted in our signature potting mix to increase plant health, longe

Lil Bud Plant is highlight rated amongst it's p... People are quite excited about its essense.

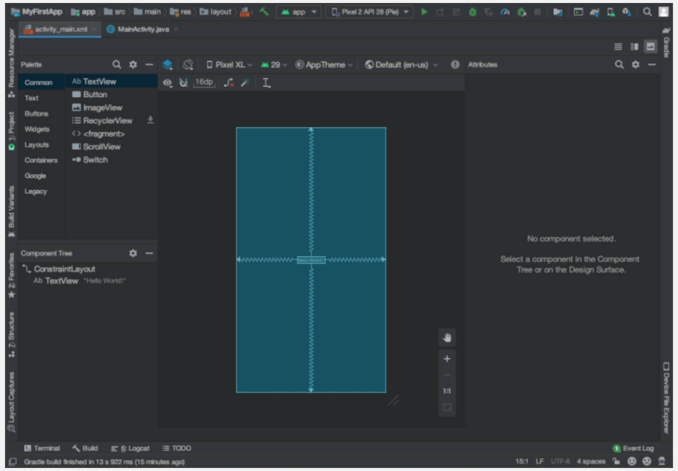
Figma

Development Tools for Specific Platforms

- Some companies also provide interface builders within their software development tools for code generation



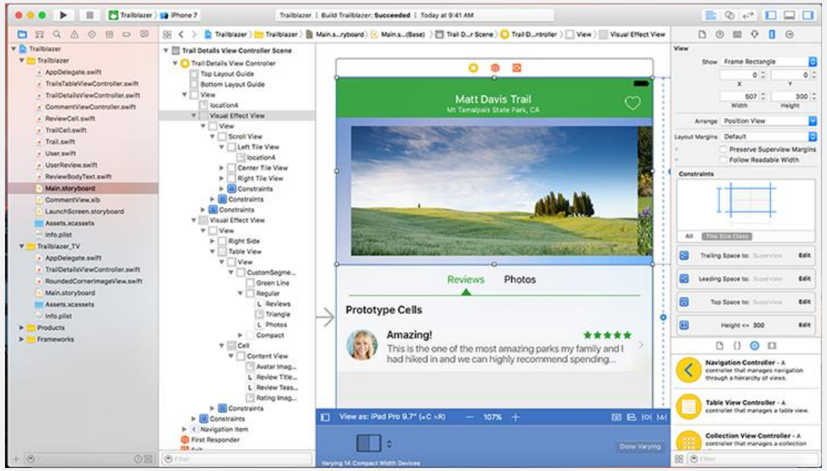
Microsoft Visual Studio



No component selected.

Select a component in the Component Tree or on the Design Surface.

Android Studio



This is the one of the most amazing parks my family and I had hiked in and we can highly recommend spending...

Apple Xcode

Wizard of Oz

- A method of testing a system that does not exist – the listening typewriter, IBM 1984
  - “the state of the art in automatic speech recognition today, however, is not advanced enough […]” p295
  - “an increasingly important […] human factors tool is simulation of user interfaces before the interfaces are ever built” p295

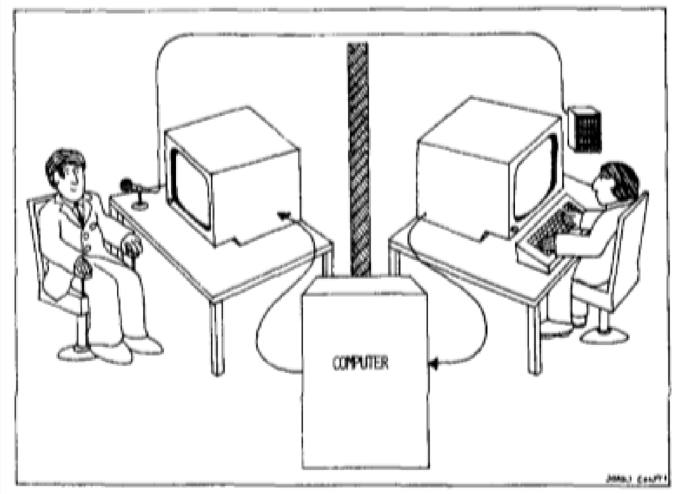
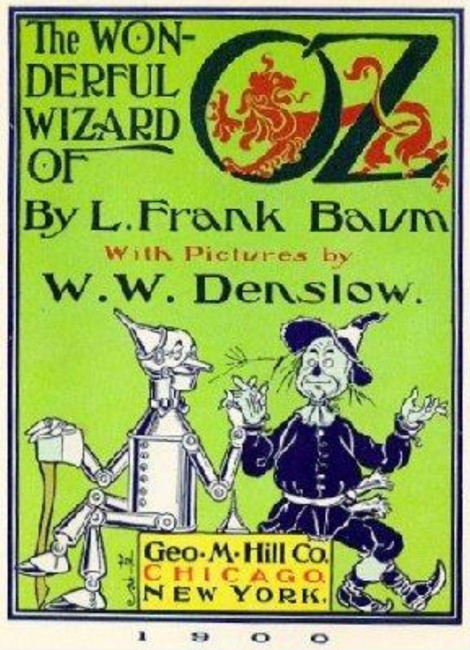


Figure 1 from the listening typewriter simulation

John D. Gould, John Conti, and Todd Hovanyecz. 1983. Composing letters with a simulated listening typewriter. Commun. ACM 26, 4 (April 1983), 295–308. DOI:https://doi.org/10.1145/2163.358100

Wizard of Oz

- Based on an American children's novel where an ordinary man pretends to be a powerful wizard by using "amplifying technologies" while hidden behind a curtain
- First used by John Kelly in his dissertation to simulate a natural language processer
- To the participant, it would appear that they are interacting with a working system



By User User: on en.wikipedia - Originally from en.wikipedia; description page is (was) here, Public Domain, https://commons.wikimedia.org/w/index.php?curid=854090

Typical Uses of the Wizard of Oz Method

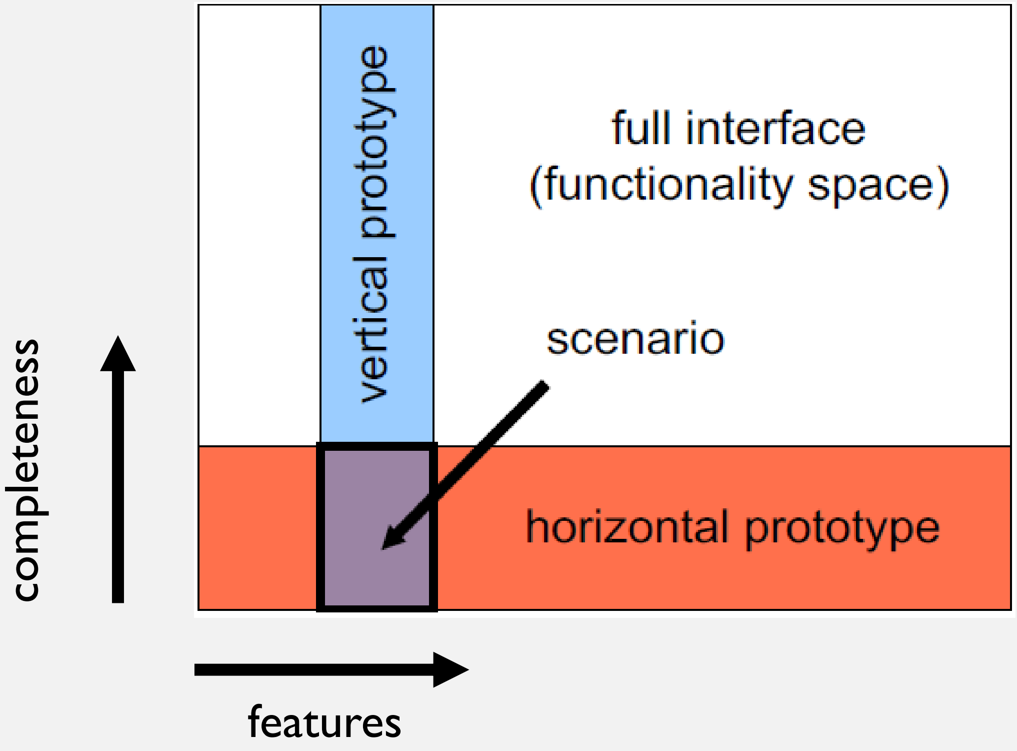
- Idea is to “trick” the participant to think that they are interacting with a working system

- Good for
  - Interfaces with high cost of development
  - Testing best-case design (e.g., voice recognition)
  - Limited time and recourses and at earlier stages of development
  - Envisioning future systems where the technologies aren’t perfect yet (common in research)

Depth & Breadth of a Prototype

- Prototype is a "discount" view of the real system, meaning it will only be able to show part of the final product
  - Need to choose what to show

- Breadth: lots of features but limited in functionalities
  ➔ horizontal prototype
- Depth: detailed functionalities but limited features
  ➔ vertical prototype



Horizontal VS. Vertical Prototypes

HORIZONTAL

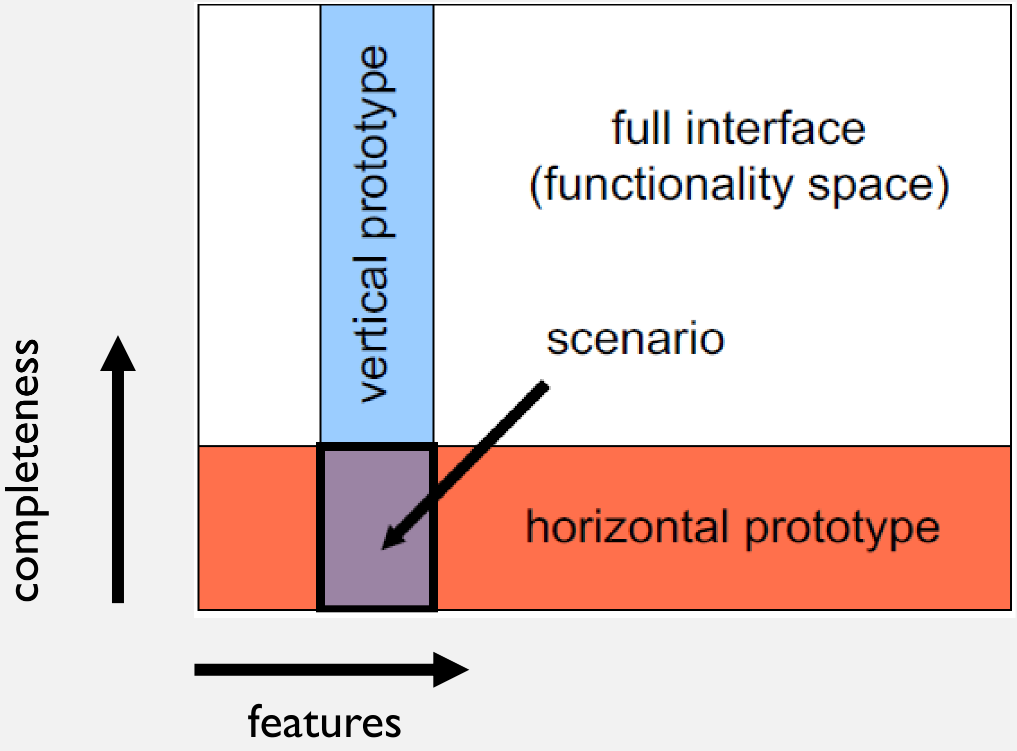
- Broad in features, but shallow in functionality
- Best used for
  - Demonstrating the product concept and conveying an early overview to stakeholders
  - Getting a sense of how much functionality will really be used to prioritize resource allocation
- Shopping app example: Most interactive elements work but do not take the user anywhere further

VERTICAL

- Deep in functionality, but narrow in features
- Best used for
  - Supporting realistic user experience evaluation, often achieved by using some stock/fake data
  - Isolating part of an individual interaction workflow to understand actual usage
- Shopping app example: user can go through the whole ordering process of only a few products

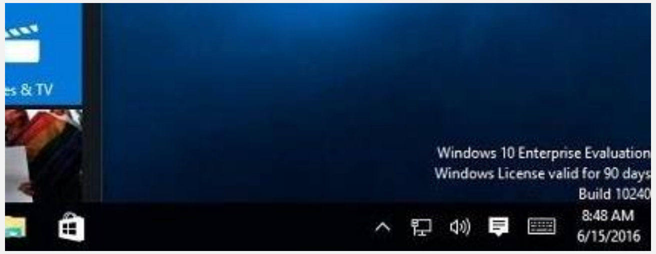
Variations of the Horizontal & Vertical Prototypes

- “T” prototypes
  - Much of the interface is realized in a shallow level, with one (or a few) part done in depth with complete functionality
  - Grow by adding more “stems” to the features
- Local prototypes (scenario)
  - Intersection between a feature and a step in the task flow (e.g., showing items in the shopping cart)
  - Useful for design alternatives (e.g., different layouts of the shopping cart page)
  - Grow in either direction



High-Fidelity Prototypes (HFPs)

- Nearly fully implemented
- Looks and behaves like the final design
- Using materials one would expect to be in the final design
    - For interfaces it will be the content that it is showing
    - For objects it will be the physical matter/medium it is built with



Windows 10 Enterprise Evaluation
Windows License valid for 90 days
Build 10240
8:48 AM
6/15/2016

Top-By u/Kruzat but modified by User:Smnt - https://imgur.com/a/tp7JG6P but modified by User:Smnt, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=84813507
Bottom-https://www.pcmag.com/how-to/how-to-try-windows-10-for-free-for-90-days

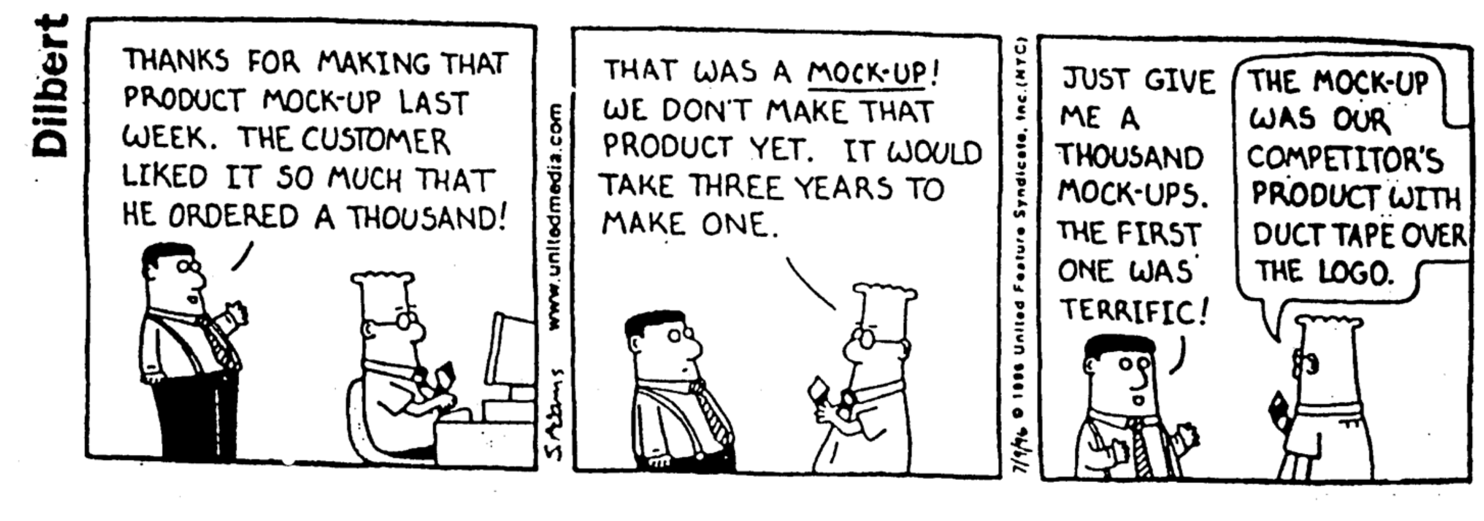
Using HFPs

BENEFITS

- Realistic – fully-featured
- Allows detailed evaluation & get detailed feedback
- Can be used as a pitching tool

DRAWBACKS

- Needs a lot of time and effort to create
- Changes can be time-consuming
- Users/managers might think it’s done


Dilbert
THANKS FOR MAKING THAT PRODUCT MOCK-UP LAST WEEK. THE CUSTOMER LIKED IT SO MUCH THAT HE ORDERED A THOUSAND!
www.unitedmedia.com
S.Adams
THAT WAS A MOCK-UP! WE DON'T MAKE THAT PRODUCT YET. IT WOULD TAKE THREE YEARS TO MAKE ONE.
7/9/96 © 1996 United Feature Syndicate, Inc.(NYC)
JUST GIVE ME A THOUSAND MOCK-UPS. THE FIRST ONE WAS TERRIFIC!
THE MOCK-UP WAS OUR COMPETITOR'S PRODUCT WITH DUCT TAPE OVER THE LOGO.

LFPs VS. HFPs (based on ID-Book p431)

Type	Advantages	Disadvantages
LFPs	• Making revision is easy and cheap • Can spend more time on the design before investing resources into development • Possible to have multiple designs to compare • Regarded as a proof-of-concept and people are more willing to be critical	• Limited error checking • Details can be missing • Limited usefulness for usability testing due to lack of polished look & feel, and functionalities • Heavy reliance on facilitator leads to limited navigation & flow
HFPs	• Close to the finished design to elicit useful and convincing feedback • Computer-controlled mechanisms lead to full range of navigation & flow	• Expensive to develop and change • Potential of setting unrealistic expectations, harder to get critical feedback

Summary

- Sketching & Prototyping
  - Levels of fidelity – uses, benefits/drawbacks
  - Wizard of Oz – a way to let people experience the design while it is still being developed
  - Horizontal, vertical, and variations of prototypes – properties, uses

Post-Lecture Activity

- Read/watch these (and those in the slides)
  - Chapters 12 of ID-Book: Design, Prototyping, and Construction, Chapter 11 of the UX book
  - Paper Prototyping 101, UX Prototypes: Low Fidelity vs. High Fidelity https://www.nngroup.com/videos/paper-prototyping-101/, https://www.nngroup.com/articles/ux-prototype-hi-lo-fidelity/
  - Prototyping 101: The Difference between LFP and HFP and When to Use Each https://theblog.adobe.com/prototyping-difference-low-fidelity-high-fidelity-prototypes-use/
  - How to Create a LFP like an App Expert https://www.thinklions.com/blog/how-to-create-a-low-fidelity-prototype-like-an-app-expert/

- With your LFP design for the LED behaviour
  - Go to http://jsbin.com/gosake/1/edit?js,output to implement the MFP version of your LED design
  - Show this to your friends & family, give them the list of the 6 informational states, see if they can correctly map them to your design