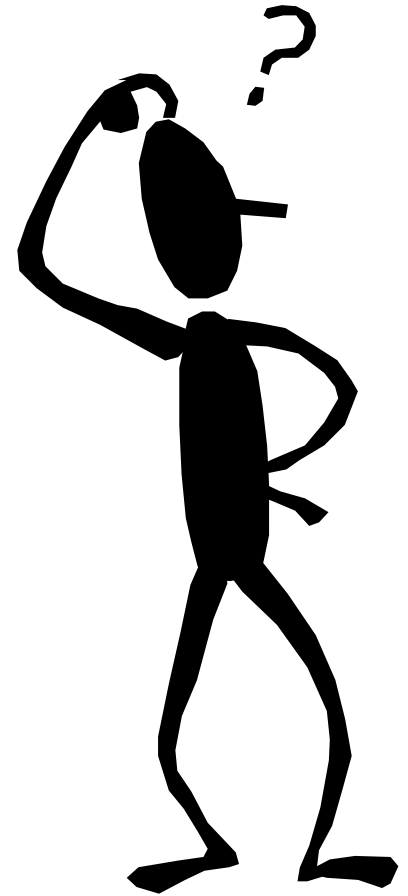


## Chapter 12

# DESIGN, PROTOTYPING and CONSTRUCTION

# Overview

- Prototyping
- Conceptual design
- Concrete design
- Using scenarios
- Generating prototypes
- Construction



# Prototyping

- What is a prototype?
- Why prototype?
- Different kinds of prototyping
  - Low fidelity
  - High fidelity
- Compromises in prototyping
  - Vertical
  - Horizontal
- Final product needs to be engineered

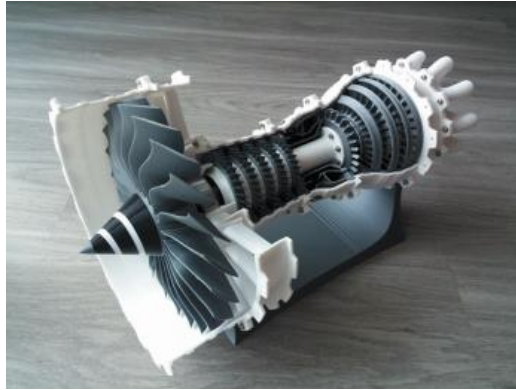
# What is a prototype?

- One manifestation of a design that allows stakeholders to interact with it
- In other design fields, a prototype is a small-scale model:
  - A miniature car
  - A miniature building or town

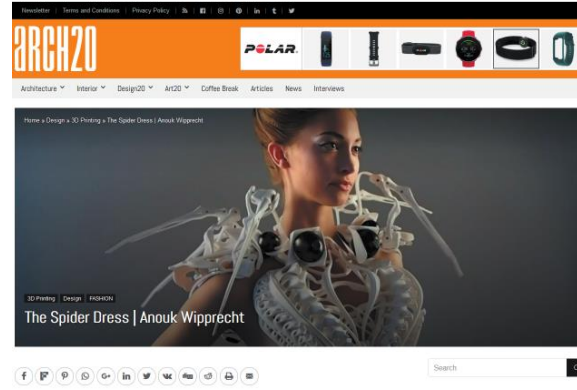
Source: [PalmPilot wooden model](#)  
© Mark Richards



# 3D Printing Examples



(a)



(b)



(c)

Examples of 3D printing: (a) model jet engine, (b) Spider Dress 2.0 by Anouk Wipprecht, and (c) teddy bear “printed” form a wireframe design

Sources: (a) [Build Your Own Jet Engine](#). Licensed under CC-BY-3.0, (b) [arch20](#), and (c) used courtesy of Scott Hudson

# What is a prototype in interaction design?

In interaction design, a prototype can be (among other things):

- A series of screen sketches
- A storyboard, for example, a cartoon-like series of scenes
- A PowerPoint slide show
- A video simulating the use of a system
- A lump of wood (for instance, the PalmPilot)
- A cardboard mock-up
- A piece of software with limited functionality written in the target language or in another language

# Why prototype?

- Evaluation and feedback are central to interaction design
- Stakeholders can see, hold, and interact with a prototype more easily than a document or a drawing
- Team members can communicate effectively
- Ideas can be tested out
- Prototyping encourages reflection: an important aspect of design
- Prototypes answer questions and support designers in choosing between alternatives

# Low-fidelity Prototyping

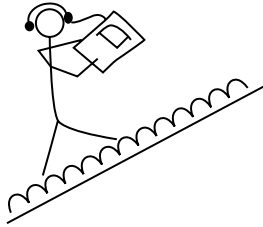
- Uses a medium which is unlike the final medium, for example, paper or cardboard
- Is quick, cheap, and easily changed
- Examples:
  - Sketches of screens, task sequences, and so on
  - 'Post-it' notes
  - Storyboards
  - 'Wizard-of-Oz'



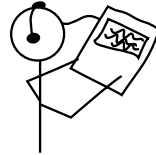
# Storyboards

- It is a series of sketches showing how a user might progress through a task using the product
- Often used with scenarios, bringing in more detail and a chance to role play

# Example storyboard



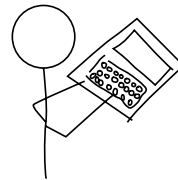
Christina walks up hill; the product gives her information about the site



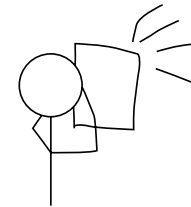
Christina adjusts the preferences to find information about the pottery trade in Ancient Greece



Christina scrambles to the highest point



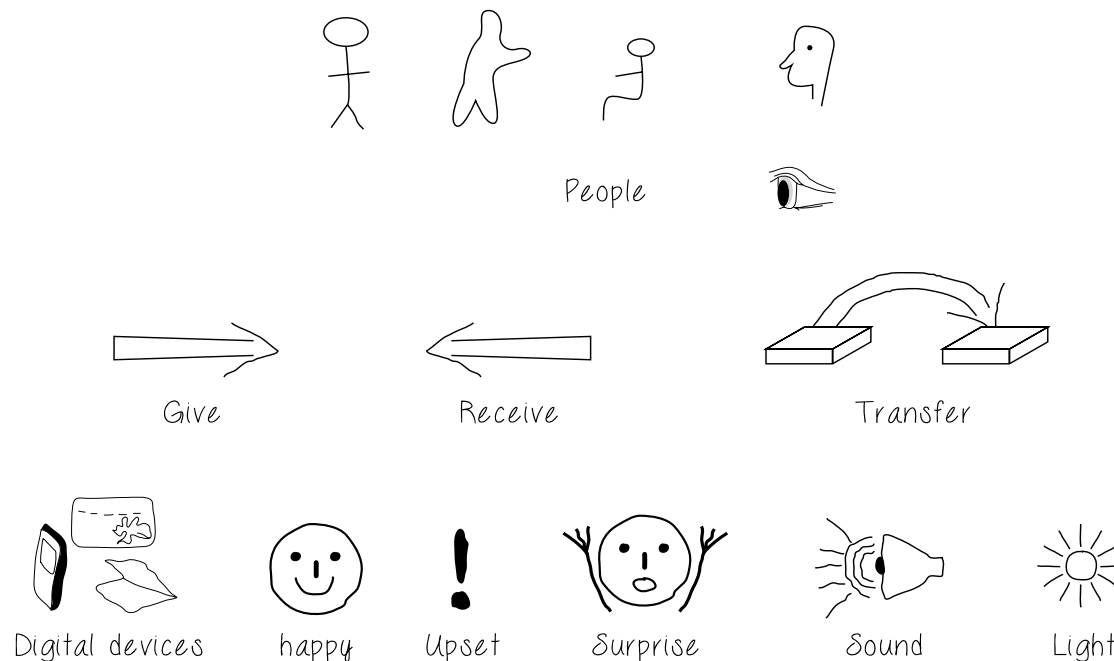
Christina stores information about the pottery trader's way of life in Ancient Greece



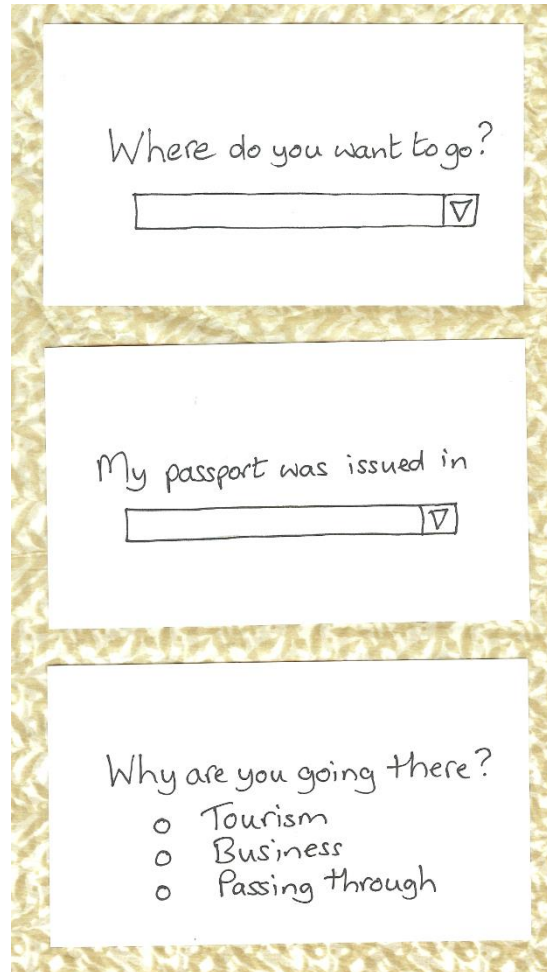
Christina takes a photograph of the location of the pottery market

# Sketching

- Low-fidelity prototyping often relies on sketching
- Don't be inhibited about drawing ability — Practice simple symbols



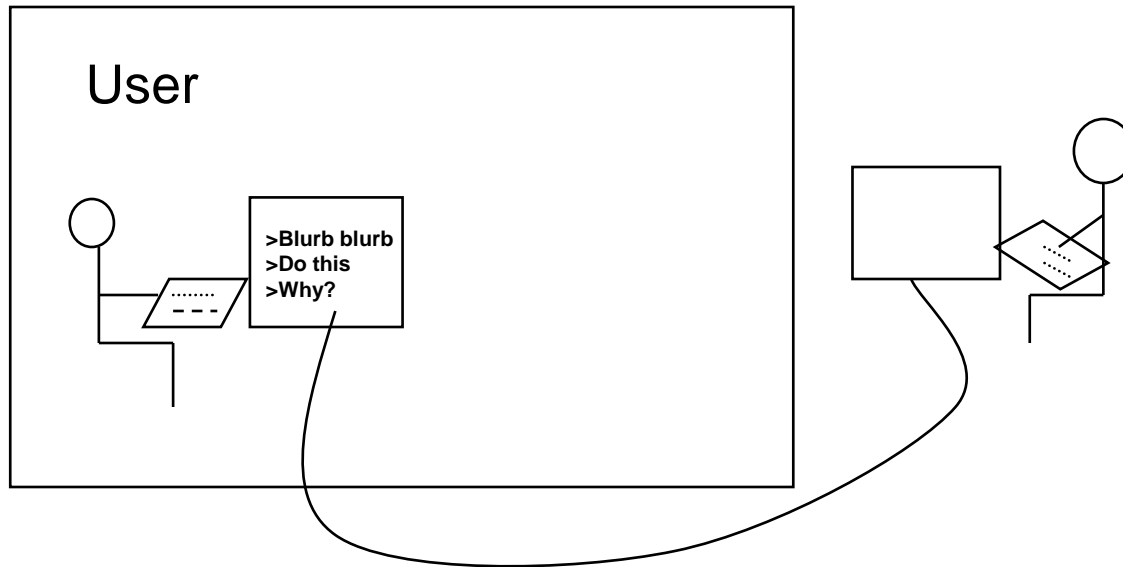
# Prototyping with index cards



- Index cards (3 x 5 inches)
- Each card represents one element of interaction
- In evaluation, can step through the cards

# 'Wizard-of-Oz' prototyping

- The user thinks they are interacting with a computer, but a human is responding to output rather than the system
- Usually done early in design to understand users' expectations
- What is 'wrong' with this approach?



# High-fidelity prototyping

- Uses materials that you would expect to be in the final product
- Prototype looks more like the final system than a low-fidelity version
- High-fidelity prototypes can be developed by integrating existing hardware and software components
- Danger that users think they have a complete system...see compromises

# Compromises in prototyping

- Prototyping involve compromises
- For software-based prototyping, maybe there is a slow response? sketchy icons? limited functionality?
- “In the wild” prototypes operational but not necessarily robust
- Two common types of compromise:
  - Horizontal:** Provides a wide range of functions, but with little detail
  - Vertical:** Provides a lot of detail for only a few functions
- Compromises in prototypes must not be ignored. Product needs engineering

# Conceptual design

- A conceptual model is an outline of what people can do with a product and what concepts are needed to understand and interact with it
- Understand problem space and current requirements; empathize with users
- Creativity and brainstorming techniques
- Mood board may capture desired feel
- Consider alternatives: scenarios and prototyping helps



# Building scenarios



# Empathizing with users



The Third Age Suit

Source: [Ford Motor Company](#)

# Choosing an interface metaphor

- Interface metaphors combine familiar knowledge with new knowledge in a way that will help the user understand the product.
- Three steps: understand functionality, identify potential problem areas, and generate metaphors
- Evaluate metaphors:
  - How much structure does it provide?
  - How much is relevant to the problem?
  - Is it easy to represent?
  - Will the audience understand it?
  - How extensible is it?

# Considering interaction and interface types

- Which interaction type?
  - How the user invokes actions
  - Instructing, conversing, manipulating, exploring, or responding
- Do different interface types provide insight?
  - Shareable, tangible, augmented reality, and so forth

# Expanding the initial conceptual model

- What functions will the product perform?
  - What will the product do and what will the human do?
- How are the functions related to each other?
  - Sequential or parallel?
  - Categorizations, for example, all actions related to privacy on a smartphone
- What information is needed?
  - What data is needed to perform the task?
  - How is this data to be transformed by the system?

# Concrete design

- Difference between conceptual and concrete is emphasis
- Many aspects to concrete design
  - Color, icons, buttons, interaction devices, and so on
- User characteristics and context
  - Inclusiveness, input, and output modes
- Accessibility
  - Web Content Accessibility Guidelines
- Cross-cultural design
  - Language, colors, icons, and information architecture
  - Indigenous knowledge and perspectives

# Generating prototypes

- Generate a storyboard from a scenario
  - Break down scenario into steps
  - Create a scene for each step
- Sketching out a storyboard prompts designers to think about design issues
- Generate a card-based prototype from a storyboard or from a use case
  - Consider each step in use case – what interaction element is needed
  - Draw a card that captures it

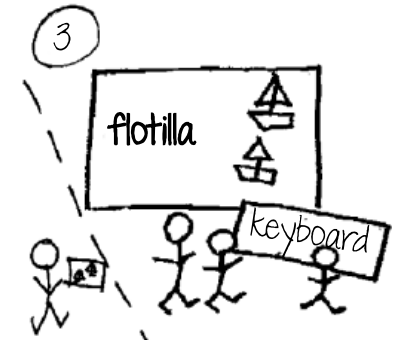
# Generating storyboard



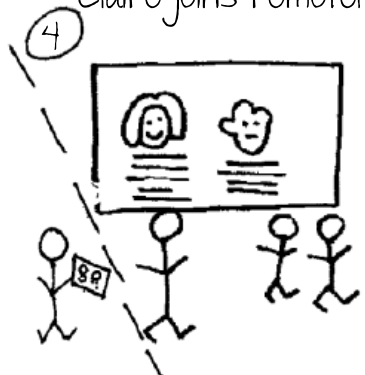
Thomson family gather around  
Claire joins remotely



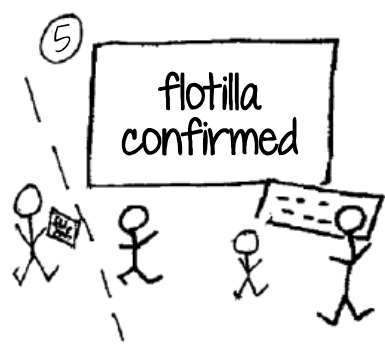
Will tells their  
initial idea



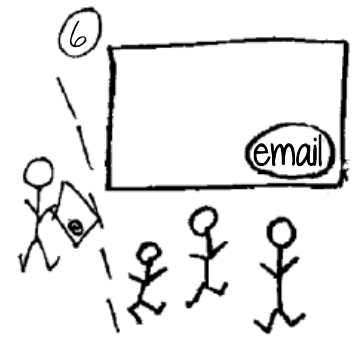
System suggests  
flotilla



System shows  
descriptions



Will asks  
for details



Details emailed



# Generating card-based prototype

Where do you want to go?

My passport was issued in

Why are you going there?

- Tourism
- Business
- Passing through

Destination

Nationality

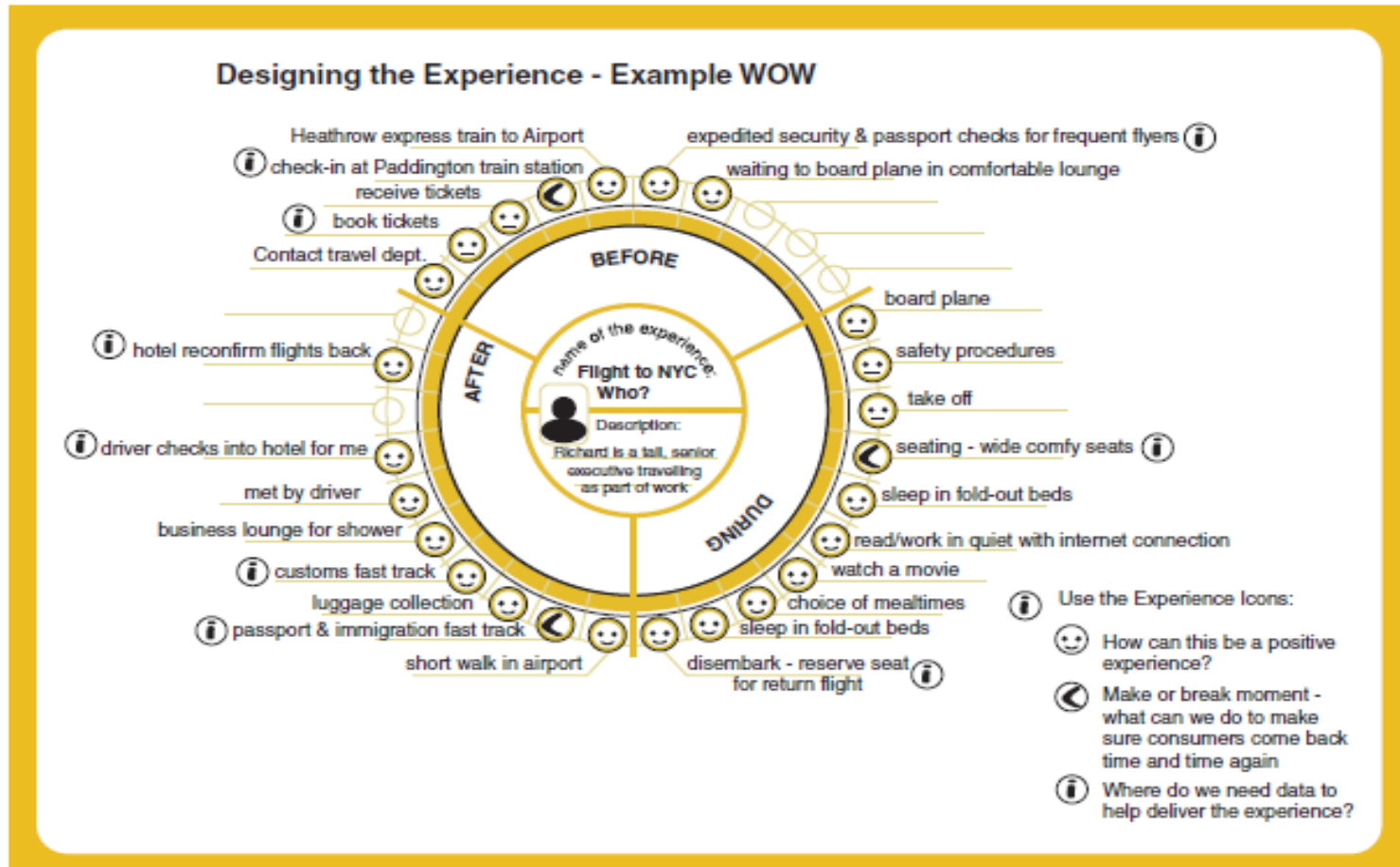
The purpose of my trip is

- Tourism
- Business
- Transit

# Explore the user's experience

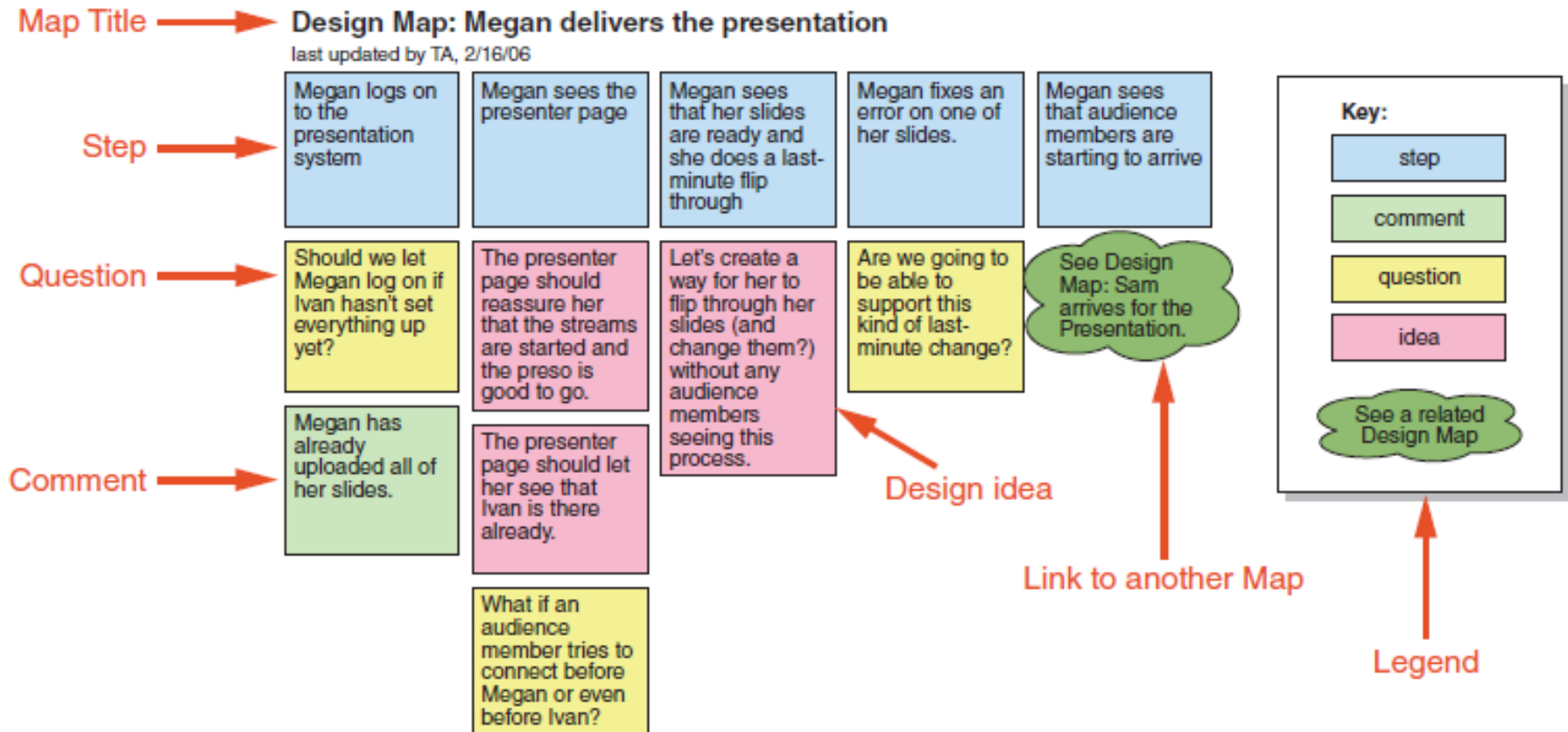
- Use personas, card-based prototypes, or stickies to model the user experience
- Visual representation called:
  - Design map
  - Customer or user journey map
  - Experience map
- Two common representations
  - Wheel
  - Timeline

# An experience map drawn as a wheel



Source: [LEGO](#)

# An experience map drawn as a timeline

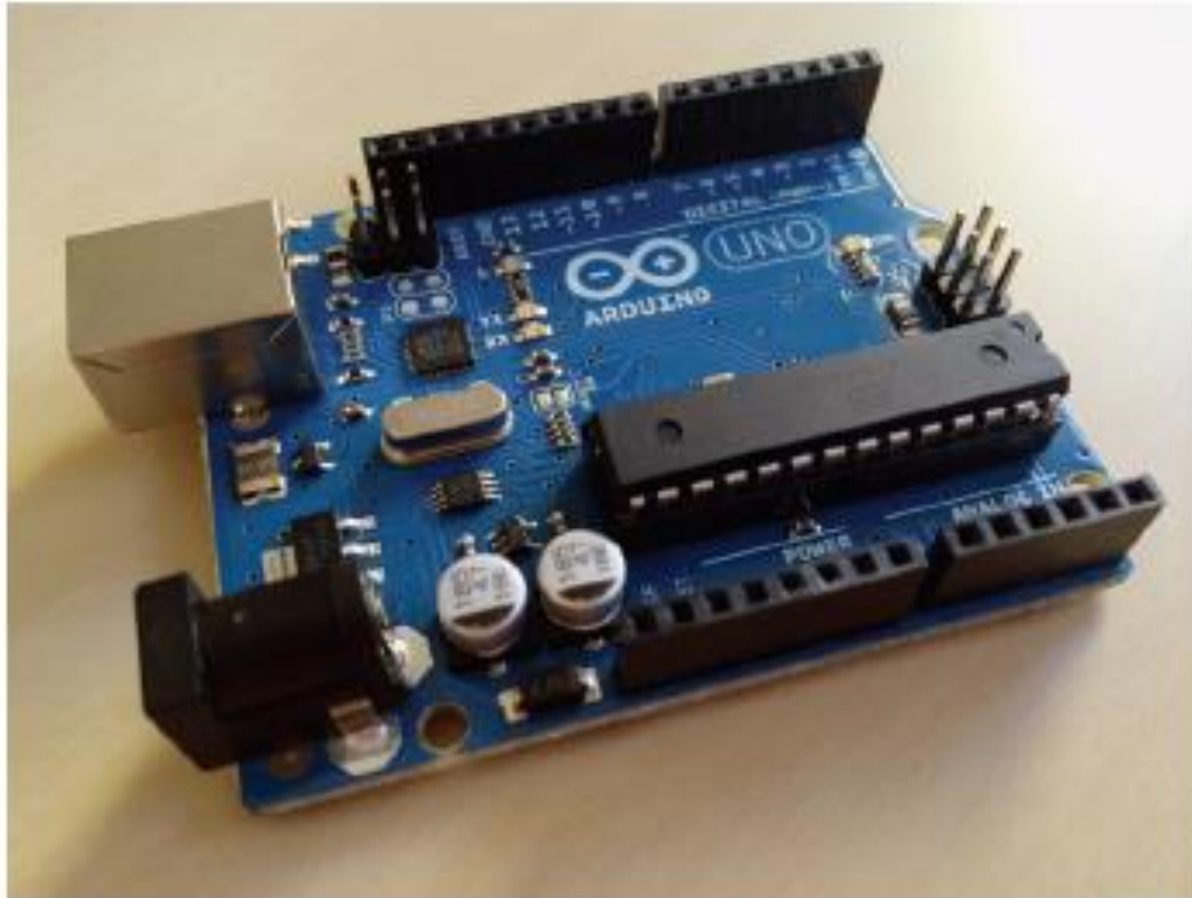


Source: Adlin and Pruitt (2010), p134. Used courtesy of [Morgan Kaufmann](#).

# Construction: Physical computing

- Build and code prototypes using electronics
- Toolkits available include
  - Arduino
  - LilyPad (for fabrics)
  - Senseboard
  - BBC micro:bit
  - MaKey MaKey
- Designed for use by wide range of people

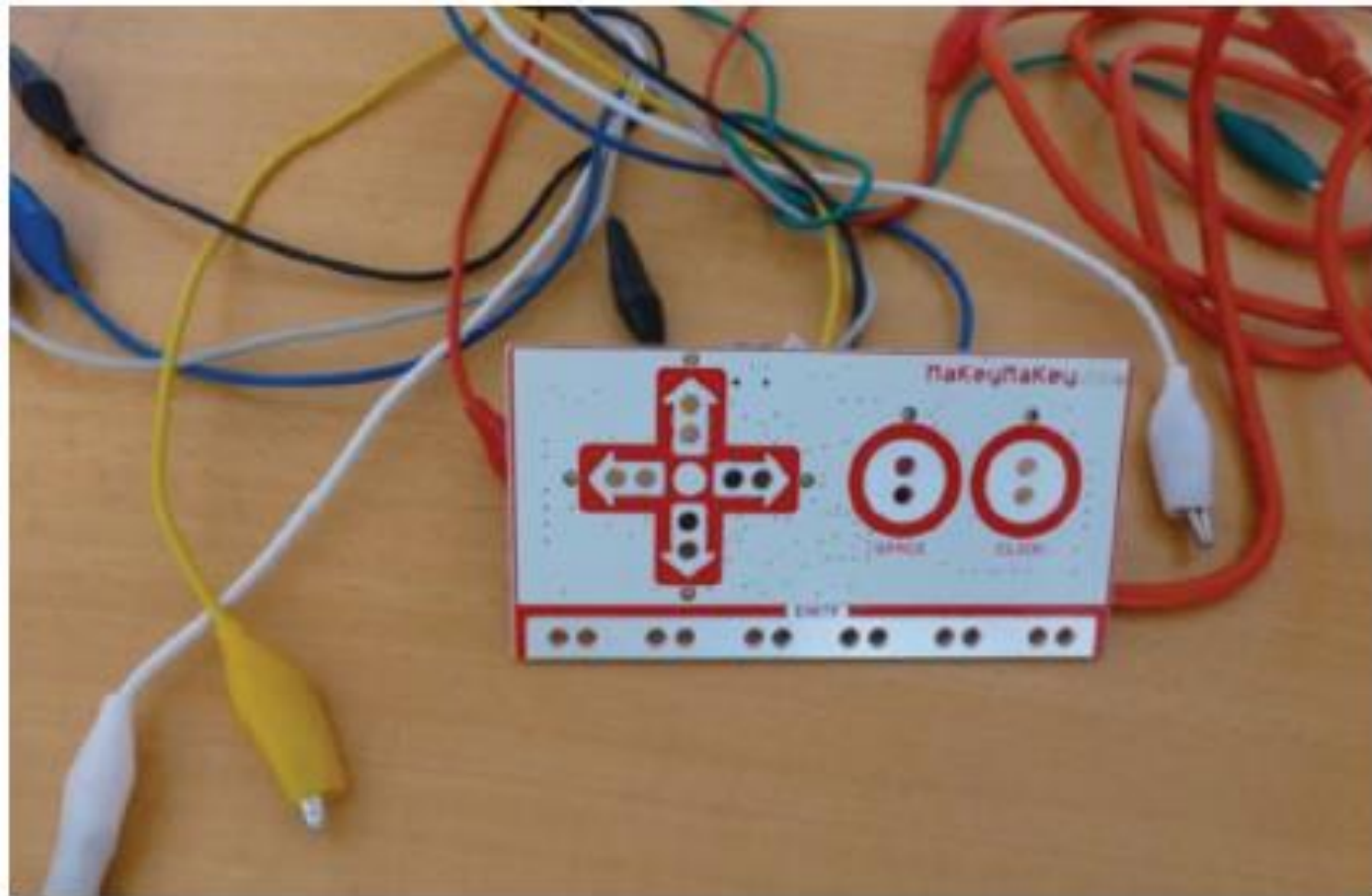
# Physical computing kits



The Arduino board

*Source:* Used courtesy of Dr. Nicolai Marquardt

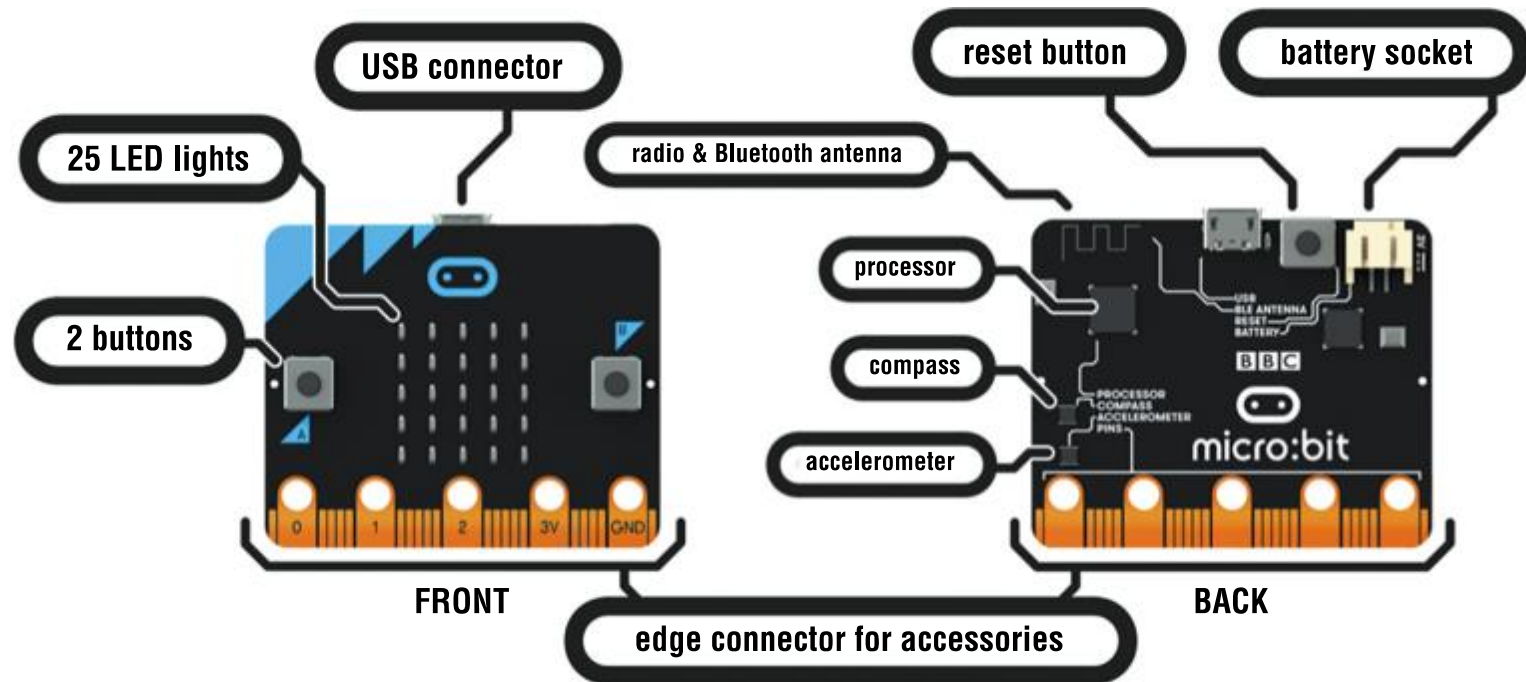
# Physical computing kits



The Makey Makey toolkit

Source: [Makey Makey](http://makeymakey.com)

# Physical computing kits



The BBC micro:bit

Source: [micro:bit](https://microbit.org/). Used courtesy of Micro:bit Foundation



# Construction: SDKs

- Software Development Kits
  - Programming tools and components to develop for a specific platform, for example, iOS
- Includes: IDE, documentation, drivers, sample code, and application programming interfaces (APIs)
- Makes development much easier
- Examples:
  - Amazon's Alexa Skills Kit for voice-based services
  - Apple's ARKit for augmented reality
  - Microsoft's Kinect SDK for motion tracking

# Summary

- Prototyping may be low fidelity (such as paper-based) or high fidelity (such as software-based)
- Existing software and hardware helps create prototypes
- Two aspects to design: conceptual and concrete
- Conceptual design develops an outline of what users can do and what concepts are needed to understand the product.
- Concrete design specifies design details, for example, layout or navigation

# Summary (*continued*)

- Three approaches to develop an initial conceptual model: interface metaphors, interaction styles, and interface styles.
- Expand an initial conceptual model by considering whether product or user performs each function, how those functions are related, and what information is required to support them
- Generate prototypes from scenarios and use cases
- Physical computing kits and software development kits facilitate the transition from design to construction