What is a Prototype?

**Student answers**
- Early rendition of your project
- Not completely functional
- Some level of functionality
- Proof of concept
- UI – includes mock-ups

**Why?**
- Decide whether to build the real system
- Lessens risk
- Get better client specifications

---

Prototyping Guidelines

- Address high risk issues
  - Requirements uncertain
  - User interface unknown
  - Implementation strategy unknown
  - Platform unknown
- Focus only on the issue
  - Build a system that addresses the problem and ignores all other concerns
  - Ok to suspend normal development standards and QA

---

Benefits of Prototyping

- Help to elicit and validate requirements
- Explore a UI design
- Explore potential designs and solutions
- Oracle for later testing

---

UI Design

- UIs are hard to design/specify
  - Interactive nature
  - Graphical layout
  - Difficult for users to think abstractly
- Mock-ups
  - Provide direct experience with interface
  - Examples make it easier to identify good, bad characteristics
  - Observe look and feel
  - Work through usage scenarios
  - See if all information, options are accounted for
Paper UI Prototyping

- Good first step for a UI design
- Storyboard
  - Draw versions of system screens
  - Good for group presentation
- Walk through scenario
  - Draw information displayed, options available
  - Good for detailed individual feedback

Live UI Prototyping

- Wizard of Oz
- Interact with UI program
- Inputs given to behind-the-scenes programmer who generates responses
- Scripts
  - Create screens
  - Associate script with buttons
  - Script produces next screen
  - Logic may be hard-coded

System Design

- Demonstrate feasibility of system
- Learn about necessary technologies
- Explore design technique
  - Ideally, see limitations of design and do it a different way “for real”

Throw it away

- “Plan to throw one away...you will anyway” – Fred Brooks
- No investment in quality
  - No documentation, testing
    - Example: no unit tests in XP “spike solution”
  - Design may be poor
    - Unmaintainable, poor performance & reliability, insecure...

Risks of Prototyping

Student answers

- Manager doesn’t understand – requires you to keep working on it
- Team that comes afterwards may not learn what you learned

Risks of Prototyping

- Client may believe the system is real
- Unrealistic expectations of progress
- Delivering or building on the prototype is almost always a mistake
- Implementors make poor choices
  - Justified in prototype, but not in real system
- Tempting to build the real system the same way
- Prototype is not identical to final system
  - Users may interact differently due to different response characteristics
  - Must interpret prototype experience with care
## Prototyping Assignment

- Prototype plan
- Goals of prototyping (pick one or two)
- Requirements elicitation/refinement
- UI mockup
- Explore a design
- Prove feasibility
- Learn a technology
- What you will build
- How you will evaluate it
- Prototype results
  - What you did
  - What you learned
  - Updates to requirements and risks
  - Changes to schedule or design
  - Other lessons learned
- Presentation of all of the above in class

## Problem Vs. Solution

- Requirements: what is the problem?
- What will the system do
  - Addresses key risk: building the wrong system
  - Important to focus on independently of solution
  - Avoid prejudicing the design
- Design: what is the solution?
- How the system will do it

## Patient Monitoring

A patient monitoring program is required for the ICU (intensive care unit) in a hospital. Each patient is monitored by an analog device which measures factors such as pulse, temperature, blood pressure, and skin resistance. The program reads the factors from the analog device (specified for each patient) and stores the factors in a database. For each patient, safe ranges for each factor are also specified by medical staff. If a factor falls outside a patient’s safe range, or if an analog device fails, the nurses’ station is notified.

## Understanding Software Problems

15-413: Introduction to Software Engineering

Jonathan Aldrich

Problems and quotations taken from Problem Frames by Michael Jackson

## The Machine and the World

- Focusing on the Problem
  - Are all intensive care patients to be monitored, or only some of them?
  - Are different vital factors to be monitored for each patient, or is it the same factors for all of them?
  - Do the medical staff specify the periods as well as the values, or does someone else specify the periods?
  - In what ways does the analog device fail? How can these failures be detected and diagnosed?
  - How does monitoring need ever change while the patient is being monitored?

The solution is here
The computer and its software
Connections between the world and the computer
The problem is here
The world outside the computer
The Machine and the World

Could the context (the world) be a computer also?

Call Forwarding

- Call forwarding is a common feature in telephone systems. It allows a subscriber at one number to arrange to have incoming calls forwarded to another number: the subscriber at n1 can set up either no call forwarding at all, or forwarding to any specified number n2. The problem is to develop and describe the detailed requirements for the feature.

- Question: Should call forwarding be transitive?
  - For example, if n1 forwards to n2, which forwards to n3, should n1 then forward to n3?

The problem is not at the interface

- Supporting both kinds of delegation require additions to the interface
- However, understanding the problem is done entirely in terms of the world
  - People
  - Phone numbers
  - Moving offices
  - Delegating responsibility
- These do not appear in the interface

Moral: You have to understand the world to get the software requirements right