Product Life-Cycle: Requirements

Dewayne E Perry
ACE 5.124
Office Hours: M/W 11:00-12:00
perry @ ece.utexas.edu
www.ece.utexas.edu/~perry/education/
Dilbert on Requirements

I'll need to know your requirements before I start to design the software.

First of all, what are you trying to accomplish?

I'm trying to make you design my software.

I mean what are you trying to accomplish with the software?

I won't know what I can accomplish until you tell me what the software can do.

Try to get this concept through your thick skull: the software can do whatever I design it to do!

Can you design it to tell you my requirements?
Introduction

- We (SEs) took responsibility for both
  - The problem and solution spaces
  - The world and the machine

- Requirements Engineering:
  - Elicitation
    - Record desired behavior of the system from the user
    - Help users (if needed) in understanding the problem
  - Description
    - Document critical aspects
    - Provide user friendly feedback on the elicitation process
      - Mock-ups
      - Prototypes
      - Examples
  - Analysis
    - Consistency
    - Adequacy
    - Completeness
    - Solving the right problem
Introduction

- **External drivers**
  - **Customer needs**
    - What customer is willing to pay for
    - What problems the customer wants to solve
  - **Business needs**
    - Strategic/marketing vision
    - To minimize cost and interval
  - **Project needs**
    - Resources
  - **Domain requirements**

- **Internal drivers**
  - **Cost**
  - **Interval**
  - **System characteristics** -
    - Functional properties
    - Non-functional properties
Introduction

Basic problem: how to elicit needs of the customer
- May be computer illiterate
- May not know what they want
- May not know what they need
- May not be able to tell us what they know

Various bases for requirements
- Customer-based
  - Customer written descriptions of what they want
- Market-based
  - Set of opportunities and goals
- Entrepreneurial-based
  - A great idea that willing to bet on

Basic problems:
- Vague, ambiguous, incomplete, inconsistent, . . .
- Everything subject to change
Introduction

Documenting Requirements

Record intent
- Record of desired behavior as described by the user
  - Can be reviewed
- User decisions about the desired behaviors
  - Not programmers/developers

Characteristics
- Reference document
- Implementation independent
- Complete, except where incompleteness explicitly noted
- Avoid duplication and inconsistency

Uses
- Basis for work estimation
- Insurance against loss of knowledge
  - Resulting from churn, turnover
- Basis for test plan development
- Constraints on future changes
- Arbiter among conflicts