CS360
Business Information Systems
Analysis and Modeling

Problem Solving and System Design
Design versus Analysis

Analysis: capturing and documenting...
- stakeholders
- current practice
- functional requirements
- user defined system success factors

Design: devising and specifying...
- user information interfaces
- data structures and access mechanisms
- system formulae and algorithms
- system versus user processes and procedures
- computer system component responsibilities
- user control interfaces
- external system interfaces for data interchange
* Characterizing a problem and devising a solution are often two forms of the same process.
Design Heuristics

Stakeholder intensions
requirement elements
model elements
design elements

Resulting Design Elements

Representational paradigm or Ontology
Modeling Continuum

- Models abstract system descriptions
  - User System Diagram
  - Entity Relationship Diagram
  - Object Models
  - Use Case Diagram
  - Class Diagram
  - Sequence Diagram
  - System Flowchart
  - Menu Tree

- Analysis
- Design
- Development
- Implementation
- Operations & Maintenance
Matrix of Models

Used together these models form a design strategy

- Information System Component
  - People
  - Procedures
  - Software
  - Hardware
  - Networks
  - Data

- User-Driven Design
  - User-Driven Needs Assessment
  - Processing Specifications
  - Platform and peripheral specs
  - Input / Output File specs

- Design Models
  - Use Case
  - Class Diagram
  - Sequence Diagram
  - USD
  - Menu Tree
  - System Flowchart
User-Driven Design

As design matures the focus shifts from external to internal system considerations.

Refinement follows this path over and over.

Diagram:
- Platform
- Peripherals
- Processing
- Output Files
- Inputs
- User
Structured Decomposition

“Divide and conquer”
- solve a large complex problem by devising a system of compact, manageable parts, “modules”

Modularity
- confining the functionality to satisfy user defined needs in easily identified and accessible modules

Managing “Coupling” and “Cohesion”
- Cohesion: the extent to which all the functionality within a module is isolated and independent from functionality in other modules
- Coupling: the extent to which functionality within a module depends on the functionality of other modules

Maximize cohesion / Minimize coupling
- “within reason”
“Object-Think” achieves Structure

* Consider the problem as a special case within a “domain” of related problems
  * How is it the same as every other case?
  * How is it different from every other case?
  * Where is evolution likely to happen?
  * What parts of the problem will remain unchanged?

* Focus on the “business rules” rather than on the “accidents” of input / output
  * paper tape, punched card, keyboard, bar/QR code reader, UPC chip transponder
  * paper report, web page, text message, voice mail, “Vulcan mind meld?”

* Correctly identifying the “anatomy” of the problem is the KEY!!
Where does the road end?

* Designs are rarely “perfect!”
  * We learn with each design decision.
  * Better understanding leads to better decisions.

* But we must DELIVER!
  * Users can’t wait forever.
  * Waiting for perfection is eternal.
  * Focus on the user’s goals!

“The world is an ever-changing environment to which design must be ever attentive.”
Grey Haired Guy!
Creative Thinking

* Designing “Solutions” has many facets
  * framing the problem
  * identifying the constraints
  * recasting the context
  * creating options
  * choosing from the alternatives

* Creative designers...
  * are naturally curious
  * learn from experience but are not blinded by it
  * explore many approaches, only eliminating options once examined
  * involve as many thinkers as possible to unbind the opportunities and their imagination
  * are always looking for new ways to address old “problems” and new ways to apply proven designs to new “problems”
Evaluating Alternative Designs

A Good Design ...

- is clear and easy for the user and developer to understand
- accommodates the existing and limited variations of the current requirements ("REUSE")
- can handle increases in data or transaction load
- consistent with the user’s conception of secure and reliable system operation
- likely to allow an execution efficient implementation
- economical to implement
  - time
  - hardware
  - software
- resistant to operational and functional obsolescence
Obsolescence

* A Good Design retards obsolescence ...
  * the **process** of becoming obsolete or outdated and no longer used or usable
    * operational obsolescence occurs when - -
    * the implemented design is no longer operable to accomplish the desired task!
    * functional obsolescence occurs when - -
    * the task for which the design is implemented is no longer desired!

* Obsolescence is none-the-less inevitable
  * therefore: design is a perpetual process of renewal pursuing effectiveness and efficiency in the ever-changing environment
The Design Report

* New System Overview
  * CDs, ERDs, USD
  * User interface designs
  * Measurable system performance objectives

* Hardware and Software Specifications
  * request for proposal / bids
  * cost estimates

* Cost/Benefit Analysis

* Project Budget

* Project Status Report

* Appendices
  * input / output prototype samples
  * any additional documented project details