LECTURE 4
ROLE OF THE SYSTEMS ANALYST & DESIGN

Thanks to

Kendall and Kendall

Fifth Edition

Major Topics

- Information systems
- Phases of analysis and design (my notes)
  - Requirement Analysis, System Design, Implementation & Unit Testing
  - Integration & System Testing, Evaluation & Feedback
- System maintenance
- CASE tools (Nayyab's slides)
- Alternate methodologies

Information

- Information is an organizational resource which must be managed as carefully as other resources
  - Should keep in mind information systems.
- Costs are associated with information processing
  - Higher cost = more complex systems

Categories

- Information systems fall into one of the following eight categories:
  - Transaction processing systems (TPS)
  - Office automation systems (OAS)
  - Knowledge work systems (KWS)
  - Management information systems (MIS)
  - Decision support systems (DSS)
  - Expert systems (ES)
  - Group decision support systems (GDSS)
  - Executive support systems (EES)

New Technologies

- New technologies are being integrated into traditional systems
  - Ecommerce uses the Web to perform business activities
    - SEO-Crawler to sue all companies
  - Enterprise Resource Planning (ERP) has the goal of integrating many different information systems within the corporation
    - Production planning, marketing, sale, inventory management, shipping, payment
  - Wireless and handheld devices, including mobile commerce (m-com) Open source software

Nature of Analysis and Design

- System analysis and design is a systematic approach to identifying problems, opportunities, and objectives; analyzing the information flows in organizations; and designing computerized information systems to solve a problem

- Systems analysts act as
  - Consultants outside the businesses
  - Supporting experts within a business
  - As change agents
  - Problem solvers, and require communication skills
  - Ethical with users and customers

Systems Development Life Cycle

- The systems development life cycle is a systematic approach to solving business problems
- It is divided into seven phases
  - SDLC Phases or Software Processes
- Each phase has unique activities

Phase 1

- Identifying
  - Problems
  - Opportunities
  - Objectives
- Personnel involved
  - Analyst
  - User management
  - Systems management

Phase 2

- Determining information requirements
  - Interview management, operations personnel
  - Gather systems/operating documents
  - Use questionnaires
  - Observe the system and personnel involved
- Learn the who, what, where, when, and how, and the why for each of these

Phase 3

- Analyzing system needs
  - Create data flow diagrams
  - Document procedural logic for data flow diagram processes
  - Complete the data dictionary
  - Make semistructured decisions
  - Prepare and present the system proposal
  - Recommend the optimal solution to management

Phase 4

- **Designing** the recommended system
  - Design the user interface
    - Design Input/output
  - Design system controls
    - DFD, Algorithms etc
  - Design files and/or database
  - Produce program specifications
  - Produce decision trees or tables
    - Structure diagrams, sequence diagrams etc

Phase 5

- Developing and documenting software
  - Design computer programs using structure charts, Nassi-Schneiderman charts, and pseudocode
  - Walkthrough program design
  - Write computer programs
  - Document software with help files, procedure manuals, and Web sites with Frequently Asked Questions

Phase 7

- Testing and maintaining the system
  - Test and debug computer programs
  - Test the computer system
  - Enhance system

Phase 6

- Implementing and evaluating the system
  - Plan conversion
  - Train users
  - Purchase and install new equipment
  - Convert files
  - Install system
  - Review and evaluate system

System Maintenance

- System maintenance is
  - Removing undetected errors, and
  - Enhancing existing software
- Time spent on maintenance typically ranges from **48-60** percent of total time

System Enhancements

- Systems are enhanced for the following reasons:
  - Adding additional features to the system
  - Business and governmental requirements change over time
  - Technology, hardware, and software are rapidly changing

CASE Tools

- CASE tools are automated, microcomputer-based software packages for systems analysis and design
- Four reasons for using CASE tools are:
  - To increase analyst productivity
  - Facilitate communication among analysts and users
  - Providing continuity between life cycle phases
  - To assess the impact of maintenance

CASE Tool Categories

- CASE tools may be divided into several categories
  - Upper CASE (also called front-end CASE) tools, used to perform analysis and design
  - Lower CASE (also called back-end CASE). These tools generate computer language source code from CASE design
  - Integrated CASE, performing both upper and lower CASE functions

Upper CASE

- Upper CASE tools
  - Create and modify the system design
  - Store data in a project repository
  - The repository is a collection of records, elements, diagrams, screens, reports, and other project information
  - These CASE tools model organizational requirements and define system boundaries

Lower CASE

- Lower CASE tools generate computer source code from the CASE design
- Source code may usually be generated in several languages

Reverse Engineering

- Reverse engineering is generating the CASE design from computer program code
- Source code is examined, analyzed, and converted into repository entities
  - Definition: **Reverse engineering**, also called **back engineering**, is the process of extracting knowledge or design information from anything man-made and re-producing it or reproducing anything based on the extracted information. (wikipedia)

Advantages of Reverse Engineering

- It has the following advantages:
  - Reduced system maintenance time
  - Program documentation is produced for loosely documented programs
  - Structured programs may be generated from unstructured, older programs
  - Future system maintenance is easier to implement
  - Unused portions of programs may be eliminated

Object-Oriented Analysis and Design

- Object-oriented (O-O) analysis and design is used to build object-oriented programs
- This includes not only data but the instructions about operations that manipulate the data

Types of Object-Oriented Analysis and Design

- There are three types of object-oriented analysis and design:
  - Object-oriented analysis (OOA)
  - Object-oriented design (OOD)
  - The Unified Modeling Language (UML), a standardized object-oriented modeling language

Nature of Analysis and Design

- Structured analysis and design provides a systematic approach to developing systems and is cyclic in nature
- Analysis and design errors detected in the later phases of the systems development life cycle cost more to fix than if detected in earlier phases

Alternate Methodologies

- Alternate methodologies are available for analyzing systems
- These include
  - Prototyping
  - ETHICS
  - Project champions
  - Soft Systems Methodology
  - Multi-view