Lecture 17. Implementation and Maintenance

System Implementation
• Seven major activities
  – Coding
  – Testing
  – Installation
  – Documentation
  – Training
  – Support

The Process of Coding, Testing and Installation
• Coding
  – Physical design specifications are turned into working computer code
• Testing
  – Tests are performed using various strategies
  – Testing can be performed in parallel with coding
• Installation
  – Process during which the current system is replaced by the new system

1. MANAGING PROGRAMMING

Project Manager’s Tasks during Programming
1. The Programmer Paradox
   • More is not always better than less!
   • After the “right” number of people are assigned to a programming task, adding more people slows down rather than speeds up completion of the project.
   • Projects requiring a large team should be broken into a series of independent, smaller parts.

2. Coordinating Activities
   • Weekly (hopefully brief) meetings
   • Create and follow standards
   • Organize programmer’s work areas
     – Development area
     – Testing area
     – Production area
   • Implement change control mechanisms
   • Use program log to monitor program changes

3. Managing the Schedule
   • Use initial time estimates as a baseline
   • Revise time estimates as construction proceeds
   • Fight against scope creep
   • Monitor “minor” slippage
   • Create risk assessment and track changing risks
   • Fight the temptation to lower quality to meet unreasonable schedule demands

2. DESIGNING TESTS

Categories of Testing
• Stub testing
  – Tests control structures before all modules are written
• Unit testing
  – Tests each module to assure that it performs its function
• Integration testing
  – Tests the interaction of modules to assure that they work together
• System testing  
  – Tests to assure that the software works well as part of the overall system  
• Acceptance testing  
  – Tests to assure that the system serves organizational needs  

**Unit Testing**  
• Black Box Testing  
  – Focuses on whether the unit meets requirements stated in specification  
• White-Box Testing  
  – Looks inside the module at actual code  

**Integration Testing**  
• User interface testing  
  – Tests each interface function  
• Use-scenario testing  
  – Ensures that each use scenario works correctly  
• Data flow testing  
  – Tests each process in a step-by-step fashion  
• System interface testing  
  – Ensures data transfer between systems  

**System Testing**  
• Requirements Testing  
  – Ensures that integration did not cause new errors  
• Usability Testing  
  – Tests how easy and error-free the system is in use  
• Security Testing  
  – Assures that security functions are handled properly  
• Performance Testing  
  – Assures that the system works under high volumes of activity  
• Documentation Testing  
  – Analysts check the accuracy of documentation  

**Acceptance Testing**  
• Alpha Testing  
  – Performed by users to assure they accept the system; frequently repeats earlier tests  
• Beta Testing  
  – Uses real data, not test data. Actual users monitor for errors or needed improvements.  

### 3. DEVELOPING DOCUMENTATION  

**Types of Documentation**  
• System Documentation  
  – Intended to help programmers and analysts understand and maintain the system after it is installed  
• User Documentation  
  – Intended to help users operate the system  

**Types of User Documentation**  
• Reference documents  
• Procedures manuals  
• Tutorials  

**Producing Documentation**  
• High quality documentation takes about 3 hours per page or 2 hours per screen  
• The task should not be left to the end of the project  
• Time required to develop and test user documentation should be built into project plan  
• On-line documentation is growing in importance
Value of Online Documentation
• Searching is simplified
• Information can be presented in multiple formats
• New methods of interacting with documentation are possible (e.g., tool tips)
• Less costly than paper documentation

Sources of Documentation Topics
• The commands and menus in the user interface
• Users’ business tasks (what they need to do)
• Definitions of terms

Sources of Navigation Terms
• The commands and menus in the user interface
• Major system concepts (e.g., data entities)
• Set of tasks performed by users
• Synonyms for the items above (users don’t always use our terminology).

Guidelines for Crafting Documentation Topics
• Use the active voice
• Minimize use of “to be” verbs
• Use consistent terms
• Use simple language
• Use friendly language
• Use parallel grammatical structure
• Use steps correctly
• Use short paragraphs

4. MAINTAINING INFORMATION SYSTEMS
The Process of Maintaining Information Systems
• Process of returning to the beginning of the SDLC and repeating development steps focusing on system change until the change is implemented
• Four major activities
  – Obtaining maintenance requests
  – Transforming requests into changes
  – Designing changes
  – Implementing changes
• Deliverables and Outcomes
  – Development of a new version of the software and new versions of all design documents created or modified during the maintenance effort

Conducting System Maintenance
• Corrective maintenance
  – Changes made to a system to repair flaws in its design, coding, or implementation
• Adaptive maintenance
  – Changes made to a system to evolve its functionality to changing business needs or technologies
• Perfective maintenance
  – Changes made to a system to add new features or to improve performance
• Preventive maintenance
  – Changes made to a system to avoid possible future problems

The Cost of Maintenance
• Many organizations allocate eighty percent of information systems budget to maintenance
  – $70 billion annually (IEEE estimate)
  – 75% of the Fortune 1000’s IS budget
• Factors that influence system maintainability
– Latent defects
– Number of customers for a given system
– Quality of system documentation
– Maintenance personnel
– Tools
– Well-structured programs

Managing Maintenance

• Number of people working in maintenance has surpassed number working in development
• Three possible organizational structures
  – Separate
    • Maintenance group consists of different personnel than development group
  – Combined
    • Developers also maintain systems
  – Functional
    • Maintenance personnel work within the functional business unit

Managing Maintenance

• Assignment of personnel
  – Maintenance work is often viewed negatively by IS personnel
  – “Maintaining a computer program is one of life’s dreariest jobs … for most American
    programmers, it is a fate worse than death” -- Edward Yourdon
  – Organizations have historically have rewarded people involved in new development better
    than maintenance personnel
  – Organizations often rotate personnel in and out of maintenance roles in order to lessen
    negative feelings about maintenance

Measures of Effectiveness

• Number of failures
• Time between each failure
• Type of failure
• Mean time between failures (MTBF)
  – A measurement of error occurrences that can be tracked over time to indicate the quality of
    a system

Website Maintenance

• Special considerations
  – 24 X 7 X 365
    • Nature of continuous availability makes maintenance challenging
    • Pages under maintenance can be locked
    • Date and time stamps
  – Check for broken links
  – HTML Validation
    • Pages should be processed by a code validation routine before publication

Website Maintenance

• Special considerations (continued)
  – Re-registration
    • When content significantly changes, site may need to be re-registered with search engines
  – Future Editions
    • Consistency is important to users
    • Post indications of future changes to the site
    • Batch changes