Chapter 12: Construction

Objectives

- Be familiar with the system construction process.
- Understand different types of tests and when to use them.
- Understand how to develop documentation.

Introduction

- *Construction* is the development of all parts of the system:
  - The software itself
  - All documentation and new operating procedures
  - Includes implementation, testing and configuration & change management work flows
- Programming is the largest, but least risky part of systems development
  - Project failure is not usually due to poor programming but to poor analysis, design, installation or project management
- Most organizations devote more time to testing & evaluation than to programming.

WILEY

Managing Programming

- Project managers must:
  - Assign programming tasks
    - Group related classes to minimize coupling & maximize cohesion of modules
    - Assign the classes to programmers
  - Coordinate activities
  - Manage the schedule

Coordinating Activities

- Hold weekly project meetings

- Create and enforce standards

- Divide resources into three areas:
  - Development
  - Testing
  - Production

- Implement change control measures

Managing the Schedule

- Time estimates must be revised as construction proceeds
  - Build a 10% error margin into all schedules
- Common cause for schedule problems is scope creep
  - Occurs when new requirements are added to the project after the system design was finalized
  - Changes become more expensive when added later in the project schedule
- Small slippages in the schedule can add up to large schedule problems
- Risk assessments can help predict problems
  - Evaluate their likelihood
  - Evaluate their impact

WILEY

Cultural Issues

- Offshore outsourcing introduces potential cultural conflicts
- Context may influence a person's ability to see potential solutions
- Individualism vs. collectivism may determine how people work together and how they view intellectual property
- Monochronic vs. polychronic determines how people view deadlines
- Other issues:
  - Power distance
  - Uncertainty avoidance
  - Masculinity vs. femininity
  - Long term vs. short term orientation

Designing Tests

- The purpose of testing is to uncover as many errors as feasible
  - It is impossible to prove that the system is error free
  - It is too expensive to look for all possible bugs
- Four stages of testing
  - Unit tests
  - Integration tests
  - System tests
  - Acceptance tests

WILEY

Testing and Object Orientation

- Encapsulation and Information-Hiding
- Polymorphism and Dynamic-Binding
- Inheritance
- Reuse
- Object-Oriented Development Process and Products

Test Planning

- A test plan define a series of tests to be conducted
- Testing takes place throughout the development of an object-oriented system
  - Develop the test plan at the beginning and modify it as the system evolves
- Each test has a specific objective and describes a set of specific test cases
  - Test specifications are created for each type of constraint that must be met by a class
  - Stubs are hard-coded placeholders that allow testing using unfinished classes

Unit Tests

- Unit tests focus on a single class
- Black box testing
    - Examines externally visible behaviors of a class
    - Driven by CRC cards, behavior state machines and method contracts, not by tester's interpretation
    - Each item in the spec becomes a test
- White box testing
    - Examines the internals of a class
    - Driven by method specifications for the class
    - Small method sizes limits the usefulness of this type of testing
- Behavioral state machines can identify tests for a class

Integration Tests

- Assess whether a set of classes that must work together do so without error
- Four common approaches
  - User interface testing
  - Use case testing
  - Interaction testing
  - System interface testing
- Most projects use all four approaches

System Tests

- Conducted to ensure all classes work together without error

- Similar to integration testing but broader in scope

  - How well does the system meet the business requirements

  - Also tests the system's documentation

Acceptance Tests

- Performed primarily by users with support of the project team
- Goal is to confirm that the system meets the business needs and is acceptable to the users
- Alpha testing—data is artificial
- Beta testing—data is real but carefully monitored for errors

Developing Documentation

- Documentation of the system must be done throughout system development

- Two fundamentally different types
  - System documentation
    - Assists programmers and analysts build or maintain the system
    - Created as the project unfolds
  - User documentation
    - Assists users to operate the system
    - Most users will not read the manuals before starting to use the system

- Online documentation makes searching simpler

- Developing & testing documentation takes time

Types of Documentation

- Reference Documents
  - Tell users how to perform specific tasks
- Procedure Manuals
  - Describe how to perform business tasks
  - Each procedure normally entails multiple tasks
- Tutorials
  - Teach people how to use specific components of a system

Designing Documentation Structure

- Online documentation will likely become the standard
- Develop a set of documentation navigation controls that lead the user to documentation topics
- Topics generally come from 3 sources
  - Commands and menus in the user interface
  - How to perform certain tasks, which can be found in:
    - Use scenarios
    - WNDs
    - Real use-cases
  - Definitions of important terms

WILEY

Writing Documentation Topics

- Start with clear titles

- Include introductory text

- Finish with detailed, step-by-step instructions

- Consider using screen images

- Video tutorials are very helpful (e.g., record the desktop while performing a task)

- Follow established guidelines (fig. 12-10)

Identifying Navigation Terms

- Table of Contents is developed from the logical structure of the documentation topics

- Sources for items for the index and search engine

  - Set of commands in the user interface (e.g., File ▸ Open)

  - Major concepts of the system (often use-cases and classes)

  - The set of business tasks to be performed (e.g., order placement)

  - Synonyms of the preceding items (users' vocabularies may not be precise)

Summary

- Managing Programming
- Designing Tests
- Developing Documentation

WILEY