COURSE NAME

SOFTWARE QUALITY AND TESTING

CSC 4133

(UNDERGRADUATE)

CHAPTER 3

MATURITY MODELS

SOFTWARE PROCESS

- A process comprises a set of activities that are executed to develop products
  - The activities are in the form of methods, techniques, strategies, procedures, and practices
  - The activities heavily rely on information repositories, such as documents, standards, and policies
- Different processes are driven by different goals and availability of resources
- It is useful to follow a defined process because of the following benefits
  - The process can be repeated in subsequent projects
  - The process can be evaluated by using a variety of metrics, such as cost, quality, and time to deliver
  - Actions can be taken to improve the process to achieve better results

SOFTWARE PROCESS

A software process comprises the following tasks.

1. Requirements Analysis
2. Designing/Modeling
3. Coding /Development
4. Testing
5. Implementation/Integration phase
6. Operation/Maintenance
7. Documentation

PROCESS IMPROVEMENT (VERIFICATION)

- To be able to improve a defined process (from baseline – existing practices), organizations need to evaluate its capabilities and limitations
- Why does a development/test process need to be improved?
  - Quality: a better test process should give more insights into the quality characteristics of a system being tested
  - Lead Time: a better test process saves testing time, and thereby gives more time to other areas of system development
  - Cost: a better test process is expected to be carried out with a lower cost
- Capability Maturity Model (CMM) allows an organization to evaluate its software development processes and supports incremental process improvement
- Test Process Improvement (TPI) model has been developed to improve the testing process
- Testing Maturity Model (TMM) has been developed to evaluate a testing process

CAPABILITY MATURITY MODEL (CMM)

- In the CMM model, the maturity level of an organization tells us to what extent an organization can produce low cost, high quality software
- Having known the current maturity level, an organization can work to reach the next higher level
- There are five maturity levels in the CMM model
    1. Level 1: Initial
    2. Level 2: Repeatable
    3. Level 3: Defined
    4. Level 4: Managed
    5. Level 5: Optimizing

CMM MATURITY LEVELS

1. Initial: processes are disorganized, even chaotic. Success is likely to depend on individual efforts, and is not considered to be repeatable, because processes would not be sufficiently defined and documented to allow them to be replicated.
2. Repeatable: basic project management techniques are established, and successes could be repeated, because the requisite processes would have been made established, defined, and documented.
3. Defined: an organization has developed its own standard software process through greater attention to documentation, standardization, and integration.
4. Managed: an organization monitors and controls its own processes through data collection and analysis.
5. Optimizing: processes are constantly being improved through monitoring feedback from current processes and introducing innovative processes to better serve the organization's particular needs.

TEST PROCESS IMPROVEMENT (TPI)

- A test process is a certain way of performing activities related to defect detection. A few such activities are as follows:
  - Identifying test goals
  - Preparing a test plan
  - Identifying different kinds of tests
  - Hiring test personnel
  - Designing test cases
  - Procuring test tools
  - Assigning test cases to test engineers
  - Prioritizing test cases for execution
  - Organizing the execution of test cases into multiple test cycles
  - Executing test cases
  - Reporting defects

TEST PROCESS IMPROVEMENT (TPI)

- How to improve a test process?
  - Step 1: Determine an area for improvement
  - Step 2: Evaluate the current state of the test process (baseline- quality, time, cost)
  - Step 3: Identify the next desired state and the means to achieve it
  - Step 4: Implement the necessary changes to the process

TESTING MATURITY MODEL (TMM)

- Similar to the concept of evaluating and improving software development processes, there is a need for a framework to assess and improve testing processes
- Continuous improvement of testing processes is an ideal goal of organizations
- Evaluation plays a key role in process improvement
- TMM pioneered by Ilene Burnstein to help org evaluate and improve testing processes
- The TMM framework describes an evolutionary path of test process maturity in five levels.
- Each level is characterized by the concepts of
  - Maturity goals
  - Supporting maturity goals
  - Activities, Tasks, and Responsibilities (ATRs) - views from manager, developer, tester, customer

TMM LEVELS

- LEVEL 1: INITIAL
  - There are no maturity goals to be met at this level, testing begins after code is written
  - An organization performs testing to demonstrate that the system works
  - Test cases are designed and executed in an ad hoc manner
  - No serious effort is made to track the progress of testing, testing is not viewed as a critical activity

TMM LEVELS

- LEVEL 2 : PHASE DEFINITION
  - Develop testing and debugging goals
  - Initiate a test planning process: Identify test objectives, Analyze risks, Devise strategies, Develop test specifications, Allocate resources
  - Institutionalize basic testing techniques and methods
- LEVEL 3 – INTEGRATION
  - Establish a software test group (test by developer vs. testing team–user view)
  - Establish a technical training program
  - Integrate testing into the software lifecycle
  - Control and monitor the testing process

TMM LEVELS

- LEVEL 4 – MANAGEMENT AND MEASUREMENT
  - Establish an organization-wide review program
  - Establish a test management program
  - Evaluate software quality
- LEVEL 5 – OPTIMIZATION/DEFECT PREVENTION AND QUALITY CONTROL
  - Application of process data for defect prevention
  - Statistical quality control
  - Test process optimization

REFERENCES

- Software Testing And Quality Assurance – Theory and Practice - Kshirasagar Naik & Priyadarshi Tripathy
- Software Quality Engineering: Testing, Quality Assurance and Quantifiable Improvement - Jeff Tian