

BCS3263 – SOFTWARE QUALITY ASSURANCE

**Assignment 2**

Lecture’s Name

MISS AZLINA BINTI ZAINUDDIN

Member’s Name

KHAIRUNNISAK BINTI NOORDIN CB12018

NORUL HUSNA BINTI OTHMAN CB13008

NORFATINFAIZAH BINTI ABD. RAHIM CB13012

NUR SYUHAIDAH BINTI ISMAIL CB13006

Based on the Corrective Actions and Preventive Actions, give one case study of mistakes or accident **FOR EACH** of internal sources types below and provide steps for corrective actions and preventive actions referring to the standard and organization process in your study. All the cases must include the original referrence with clearly stated your source of referrence.

1. Software development process
2. Software maintenance
3. SQA infrastructure procedures
4. Software quality management procedures

* Software development process

**CASE STUDY ON SOFTWARE DEVELOPMENT PROCESS**

The MHC-PMS (Mental Health Care-Patient Management System) is an information system that is intended for use in clinics. It makes use of a centralized database of patient information but has also been designed to run on a PC, so that it may be accessed and used from sites that do not have secure network connectivity. When the local systems have secure network access, they use patient information in the database but they can download and use local copies of patient records when they are disconnected. The system is not a complete medical records system so does not maintain information about other medical conditions. However, it may interact and exchange data with other clinical information systems. BloomRose Corp was given the project tender to develop the MHC-PMS system. All of the system requirement specification, budget allocation with other related specifications have been well documented, informed, understand and agreed between the client and business analysts. Therefore, the team of developer has been assigned to build the proposed system and all of the requirement specification has been point out well along with the work breakdown structures. Thus, the developers and software designers begin to discuss and it turn out that they have some contradictions on the best software methodology that should be implemented. Recently, the developers begin to develop appropriate coding without manage to use any methodology as they decide not to put much pressure on the methodology for a meanwhile. At the end, they develop the system without implement any software methodology and executed the system very well. But, at the Testing phase, the Testing team configure out a few defects in the systems which leads the developers to re-code the system for longer time than scheduled that possibly impact the system failed to be release on the exact dateline.

**Corrective Action Process**

* Locate and document the root cause of the defaults.
* Scan the entire system to ensure no other related nonconformity could occur.
* Process redesign
* Implement the best software methodology before began to develop the code
* Establish thorough follow-up to ensure the correction is effective and recurrence has been prevented.

**Preventive Action Process**

* Take proactive steps to ensure a potential nonconformity does not occur.
* Employ process and system analysis to determine how to build in safeguards and process changes to prevent non-conformance. For example, use a failure mode and effects analysis to identify risks and potential deficiencies and to set priorities for improvement.
* Initiate an improvement project, with project plans, justification for planned expenditures, resource controls and evaluation.
* Use a variety of appropriate disciplines at different times during the project.
* Establish a means for communicating what has been done and what has to be done to facilitate communication about changes to project team members.
* Include a clear trail of actions taken and decisions made to substantiate the decision to proceed, document lessons learned and avoid needless reinvention on future similar projects.

**References**

1. <http://asq.org/quality-progress/2005/03/problem-solving/corrective-vs-preventive-action.html>
2. <http://www.realsoftwaredevelopment.com/software-development-top-30-mistakes/>
3. <http://fmdic.org/wp-content/uploads/2012/05/Lewandowski-CAPA.pdf>
4. Software Engineering Edition 9th by Ian Sommerville

* Software maintenance

**CASE STUDY ON SOFTWARE MAINTENANCE**

In an effort to find out more about the tools, procedures, and techniques project personnel use in their work, the Computer-Aided Software Engineering (CASE) Environments Project interviewed personnel in eight software maintenance projects within an agency of the U.S. government .These interviews highlighted problems that we believe are typical of many software maintenance organizations (i.e., the need for more effective software maintenance tools, lack of communication between individuals working on similar projects, low status of maintenance personnel, and lack of a design-for maintenance philosophy during the software development phase). This report highlights the findings of these interviews, provides our analysis of the findings, and makes recommendations directed at the agency for improvement in the areas of tools, people, and process. We believe that what we observed is very typical of the state of the practice in these areas and as such that this report and its recommendations are applicable to other large or small software maintenance projects.

**Problem of Case Study**

1. One major finding is that the maintenance group frequently has to support software which is not developed with maintainability in mind. Schedule pressures often result in software being transferred to the maintenance phase before all deliverables are completed. Consequently, the maintenance group devotes significant time to issues related to supporting software which is poorly designed, coded, tested and documented; that is, code which was not designed for maintenance. Thus we heard a consistent message that tools for such activities as reverse engineering and testing were a high priority and that several groups were not aware of the state-of-the-art in tools. We also heard that the software to be maintained does not have effective documentation to support it, either in written reports or embedded in the code. We do not believe that it is the intent of the original developers to produce incomplete work. Rather, the pressures to meet unrealistic schedules force both in-house developers and contractors to release their software before it is ready. We also saw evidence that schedule pressures within maintenance projects result in the same problem with maintained code. Thus the problem is being perpetuated in the maintenance phase.

**Corrective Action Process**

* To overcome these problems, we recommend a two-phase approach. In the short term, there is a strong need for better tools to support reverse engineering, testing, configuration management, and documentation. These tools will help deal with the problems of existing code. In the longer term though, there is a need to improve the quality of the developed software. This involves such issues as developing schedules that provide more time to develop a quality product, writing effective software documentation, adhering to quality assurance standards, providing effective training on technical and non-technical issues, and coordinating communication between development and maintenance groups.

1. We found that communication was less effective than it could have been between different organizational entities. First, little communication appears to occur between different project personnel regarding software engineering issues. Because many of the problems confronted are common across projects, the sharing of experiences, mistakes made, lessons learned. Therefore communications ought to be encouraged through such means as informal reports, seminars, a software engineering bulletin board, etc. Second, we found that communication on technical issues from the project level to higher management could be improved. Some project personnel we interviewed felt that upper management was not aware of the depth of some of the technical problems they were facing. Third, we felt that communication between projects and contractors requires reassessment. We heard that some (but not all) contractors working for the agency performed at unacceptably low levels. This could partly be due to inadequate contractor evaluation prior to contractor selection. However, it could also be due to insufficient contractor oversight during the term of the project. Clearly contractor problems can result in significant cost impact, and the use of a TQM approach to resolving this issue (e.g., working more closely with the contractor, or forming joint teams with the contractor) could be worthwhile. Finally, we heard that the effectiveness of communications with customers varies significantly. The agency policy needs to encourage customers to be involved more closely with the project throughout its life-cycle rather than just at the time of field installation. We heard that the status and prestige of maintenance projects is less than that of development projects. Maintenance project positions are perceived as less challenging and attract personnel who have less experience. When these personnel gain experience, they move on, so that personnel stability in maintenance projects is less than it should be. However, the maintenance personnel we talked to did find their work rewarding and challenging and even preferred it to development work. Some were frustrated, though, in that they believe development groups are more likely to receive better support.

**Corrective Action Process**

* To overcome these problems we feel that every opportunity should be taken to improve the status of maintenance project personnel. As stated above, they should have a measure of control over when they have to accept code into maintenance. They also need better access to the tools to do their job effectively. Finally, they should be given public recognition (inside the agency) when an effective job has been done. The maintenance group has developed two software quality assurance documents which contain much useful information to support effective software engineering practices. If the standards in these documents were followed, then the number and severity of the product-quality problems associated with existing software development could be lessened.

**Preventive Action Process**

1. Training on quality standards should be mandatory rather than voluntary.
2. The quality standards are enforced through quality audits using an independent

Software quality assurance (SQA) audits function.

1. Improved working environment and improved morale among personnel.
2. Improved quality of products through designed for maintenance.
3. Improved quality products through enforcement of SQA standards.
4. More effective relations with contractors and customers.
5. Improved inter-project and technical/management communication.
6. Long-term reduction in costs due to effective maintenance of quality

**References**

1. A Case Study in Software Maintenance [Susan Dart Alan M. Christie Alan W Brown] <http://www.sei.cmu.edu/reports/93tr008.pdf>

* SQA infrastructure procedures
  + Internal Audit Quality Report

**CASE STUDY ON SQA INFRASTRUCTURE PROCEDURES**

For example case study is the ABC Company implemented the system used to track issues during validation does not contain sufficient information to determine or verify that the issue has been resolved. The problem is when record states 'test script was updated.' But without a document number and revision, this cannot be verified.

|  |
| --- |
| **CORRECTIVE AND PREVENTIVE ACTION FORM** |
| X  Type of Action Required: Corrective Preventive  Reason Initiated:  X  Internal Identifier Internal Audit Report Additional Details:  N/A  Customer Requested Management Review Finding  System Issue Other  What went wrong?  Audit report issue record states 'test script was updated.' But without a document number and revision, this cannot be verified.  What was done to correct it?  Contacted user directly to obtain information details. Provided compensation for the incomplete related to the audit report issue.  What should be done to prevent it from recurring?  Define all mandatory information then makes audit report enhancements to ensure all changes working as required.  Corrective and Preventive Actions by : |
| Action Plan |
| 1. Schedule a review meeting with management team to discuss issue and identify changes required. |
| 2. Complete audit report per discussion. |
| 3. Complete quality audit report and test script testing to ensure all changes working as required. |
| 4. Roll out system enhancements for any unknown issues. |

**References**

1. Software Engineering Edition 9th by Ian Sommerville
2. [http://www.ivtnetwork.com/video/change-control-case-study-1-change-control-audit- finding](http://www.ivtnetwork.com/video/change-control-case-study-1-change-control-audit-%20%20%20%20finding)

* Software quality management procedures

**CASE STUDY ON QUALITY MANAGEMENT PROCEDURES**

Representatives of the U.S. Food and Drug Administration (FDA) have been very active on several ISO 9000 technical committees, including TC 176 and TC 210. The regulatory requirements for medical devices, previously known as the good manufacturing practices, were revised in 1995 to more closely emulate ISO 9001:1994. These revisions to 21 CFR (Code of Federal Regulations) 820 were renamed the Quality System Regulation (QSR).

With the advent of the QSR, the inspection methodology used by the FDA was also changed. The agency adopted the quality system inspection technique (QSIT) for most of its routine regulatory inspections. Again, the philosophy behind this change was based on a review of the system rather than on an investigation of compliance to the letter of the regulation.

The ISO 9001 international quality management standard was not adopted verbatim for a variety of reasons, the most critical being the FDA believed--and rightly so--several of the elements in the standard were not quite rigorous enough for the purpose of ensuring a robust system for the products the agency regulates.

One crucial element that was strengthened was the need for a system (or subsystem) to monitor the effectiveness of actions implemented to resolve past or potential nonconforming conditions. This subsystem, known as corrective and preventive action (CAPA), has become of ever increasing importance and value. The requirements for this subsystem are identified in 21 CFR 820.100.

Within its documented quality (and regulatory compliance) system, a medical device manufacturer should have a high-level procedure that describes its CAPA program.

**Corrective action procedures**

* Documentation of the identified nonconforming product, process or condition should be performed according to the organization's documented procedures.
* After being informed of the identified nonconformance, responsible personnel then assign appropriate personnel to investigate the nonconformance and identify its root causes.
* Representatives from other functions may be used.
* Once the investigation is completed, appropriate meetings, discussions and training sessions are held with department personnel and other groups whose activities may be impacted by the investigation and corrective action implementation.
* The identification of the root causes (established, in part, by failure analysis, when appropriate) is indicated on the company's appropriate documents.
* The documents are completed, returned to the individuals who identified the nonconformance and then reviewed according to the process described in documented procedures.
* Once the corrective action has been verified as implemented, the documents are forwarded to quality department personnel, who enter the corrective action information into the database for future statistical analysis.

**Preventive action procedures**

* As a result of identified nonconformances (for example, identified through trending) and effective corrective actions, an individual may decide to implement a similar action to prevent the occurrence of a nonconforming condition.
* If department management identifies and approves a preventive action plan, it should also initiate documentation.
* The preventive action plan and associated documentation are described on the report form.
* Once the preventive action plan is accepted, the initiator verifies implementation and effectiveness.
* Documentation of preventive action is forwarded to quality department personnel for entry into the database and possible subsequent statistical analysis.
* Finally, records of corrective and preventive actions are maintained in accordance with the company's defined process for control of quality records.

**References**

<http://asq.org/quality-progress/2001/11/standards-outlook/corrective-and-preventive-action-in-medical-device-manufacturing.html>