Group Discussion Report 4
Process Assurance Activities for Conformance

Group Member's Names:
1. Akshay Kalsotra
2. Malika Hafiza Pasha
3. Mustafa Quraishi
4. Sajeel Mohammed Abdul
5. Shoaibuddin Mohammed

Part I: DISCUSSION

Topic 1: Improvement Process – Corrective and Preventive Actions

"Statement Software Ltd." is a software house that specializes in development of custom-made billing systems for the manufacturing industry. A common "Statement Software" contract offers the customer 12 months of guarantee services. The company's help desk (HD) supplies solution to customer calls by phone or at the customer's site. The last quarter's performance report indicates a decline in service quality, a trend that also characterizes the previous two quarters. This trend was identified by the following four HD quality metrics:

- Percentage of recurrent calls: the percentage of customer site calls that required a recurrent call to deal with a defect supposedly solve by prior call.
- Average reaction time to customer site calls (working days).
- Average hours invested in customer site calls, including travel time.
- Customer satisfaction computed from a quarterly customer satisfaction questionnaire, using a 0-10 scale.

The SQA metrics	Quarter I	Quarter II	Quarter III	Quarter IV
Percentage of recurrent calls	12	13	19	21
Average reaction time to customer site calls (days)	0.7	0.8	1.7	1.8
Average hours for customer site calls	4.7	4.9	3.3	3.1
Customer satisfaction	8.3	8.4	6.7	6.5

The *ad hoc* CAPA (Corrective and Preventive Action) team that was appointed to deal with the subject decided that each member should prepare a separate list causes for the decline in the quality of HD services, prior to analysis of the collected information and complementary observations.

1. **Each group member separately provides a list of 5 possible causes for the recorded phenomenon. Justify your list.**

Answer: Reasons and Potential Causes

1. Malika Hafiza Pasha

a. Insufficient Training for Help Desk Employees:

Justification: Because staff could not be efficiently addressing issues on the initial call, this could result in an increase in the number of recurrent calls.

b. Antiquated or including bugs in software:

Justification: If there are technical problems with the billing systems, this may result in longer average response times and a higher frequency of calls because of recurring problems that are not resolved entirely.

c. **Insufficient numbers of employees**

Justification: Because employees can be pressured to move on to the next customer before properly resolving concerns, this could lead to slower reaction times and less hours available per customer call.

d. **Inadequate Channels of Communication:**

Justification: Poor communication between consumers and the help desk may cause miscommunications and a failure to fully grasp the issues at hand, which would reduce customer satisfaction.

e. **Absence of Successful Processes for Solving Problems:**

Justification: Inconsistencies in problem handling could result in a higher call recurrence rate and worse customer satisfaction if there is no formalized method in place to direct the settlement of complicated issues.

2. Akshay Kalsotra

a. **Outdated Hardware and Infrastructure**

Justification: Older hardware or insufficient infrastructure can slow down the help desk's ability to respond quickly and effectively, impacting reaction times and the quality of support provided.

b. **Lack of Customer Feedback Integration**

Justification: Without a structured mechanism to integrate customer feedback into service improvements, recurring problems may not be effectively addressed, impacting customer satisfaction negatively.

c. **Poor Staff Morale and Motivation**

Justification: Low morale and lack of motivation among help desk employees can lead to less diligent work and a decrease in service quality, reflected in lower customer satisfaction scores.

d. **Inadequate IT Support**

Justification: If the help desk's own IT support is inefficient, this can delay the resolution of internal technical problems, indirectly affecting the service provided to customers.

e. **Lack of Regular Software Updates**

Justification: Without regular updates, software issues may not be resolved in a timely manner, leading to increased recurrence of calls due to unresolved or newly emerged software bugs.

3. Mustafa Quraishi

a. **Insufficient Use of Automation Tools**

Justification: Lack of automation in handling routine queries and issues can overburden the staff, leading to longer handling times and decreased efficiency.

b. **Inadequate Data Security Practices**

Justification: If customers feel their data is not secure, it can lead to dissatisfaction and reluctance to engage fully with support services, thereby complicating issue resolution.

c. **Ineffective Onboarding Process for New Employees**

Justification: New employees without adequate onboarding might take longer to become effective, impacting the overall efficiency of the help desk.

d. **Communication Barriers Between Departments**

Justification: Poor interdepartmental communication can lead to delays in resolving issues that require input from different parts of the company.

e. **High Employee Turnover**

Justification: Frequent changes in staff can disrupt service continuity and knowledge retention, impacting service quality.

4. Sajeel Mohammed Abdul

a. **Complex Interface of Billing Software**

Justification: If the billing software is too complex for customers to understand easily, it can increase the number of support calls and reduce customer satisfaction.

b. **Limited Training in Advanced Problem-Solving**

Justification: Help desk employees might be well-versed in routine issues but lack training in solving more complex or less common problems.

c. **Inflexible Service Hours**

Justification: If the help desk's operating hours are not aligned with customers' needs, especially in different time zones, it can lead to delayed responses and reduced satisfaction.

d. **Poorly Defined Service Level Agreements (SLAs)**

Justification: Ambiguous SLAs can lead to mismatched expectations between the company and its clients regarding response times and issue resolution.

e. **Lack of Escalation Protocols**

Justification: Without clear escalation protocols, complex issues may not be forwarded to the appropriate expertise level, delaying resolution.

5. Shoaibuddin Mohammed

a. **Inadequate Remote Support Capabilities**

Justification: Insufficient tools or policies for remote support can hinder the help desk's ability to address issues efficiently, particularly when onsite visits are not possible.

b. **Cultural Misalignment with Clients**

Justification: A lack of understanding or alignment with the cultural or business practices of clients can lead to misunderstandings and dissatisfaction.

c. **Reliance on Manual Processes**

Justification: Excessive reliance on manual processes can lead to errors and inefficiencies in handling support tickets.

d. **Lack of Proactive Issue Detection**

Justification: Without systems to proactively detect and address potential issues, the help desk may only react to problems after they have significantly impacted the customer.

e. **Inconsistent Feedback Collection and Analysis**

Justification: If feedback is not consistently collected and analyzed, improvements may not be effectively targeted, and recurring problems may persist.

2. **Summarize the causes listed by the group members, select top 5 causes, and propose your solutions for each of the causes in the top list.**

Answer:

Summary and Top 5 Causes:

Based on the collective insights from each group member, we've identified a range of issues affecting the help desk service quality at Statement Software Ltd. Here are the top 5 causes for the decline in service quality based on frequency and impact:

1. **Insufficient Training for Help Desk Employees**
    - This includes both initial training for new employees and ongoing training for advanced problem-solving.
2. **Inadequate IT Support and Infrastructure**
    - Issues related to outdated hardware, software, and insufficient IT support which delay the resolution of technical problems.
3. **Poor Communication and Collaboration**
    - This includes internal communication barriers between departments and inadequate channels of communication with customers.
4. **Lack of Effective Process Management**
    - Absence of structured problem-solving processes, unclear service level agreements, and inadequate escalation protocols.
5. **Inefficient Utilization of Technology**
    - This ranges from insufficient use of automation tools to inadequate remote support capabilities and outdated billing software interfaces.

Proposed Solutions for Each Cause:

1. **Enhanced Training Programs**
    - **Solution:** Develop a comprehensive training program focusing on both the technical aspects of the billing software and soft skills for effective communication. Implement regular training sessions that include simulations and role-playing to handle complex customer issues.
    - **Impact:** This will ensure employees are better equipped to handle initial inquiries effectively, reducing the need for recurrent calls.
2. **Upgrade IT Infrastructure and Support**
    - **Solution:** Invest in modernizing hardware and software to ensure the help desk operates efficiently. Establish a dedicated internal IT support team to promptly address any technical issues the help desk encounters.
    - **Impact:** Reduces downtime and speeds up response times, thereby improving overall service quality.
3. **Improve Communication and Collaboration**
    - **Solution:** Implement a unified communication platform that facilitates better coordination between various departments. Regular inter-departmental meetings should be held to discuss ongoing issues and solutions.
    - **Impact**: Enhances understanding and quicker resolution of complex cases that may require input from multiple departments.
4. **Streamline Processes and Define Clear SLAs**

- **Solution**: Create clear, detailed process guidelines and escalation protocols for handling different types of customer issues. Clearly define and communicate service level agreements to both staff and customers.
- **Impact:** Ensures consistency in handling cases, reduces recurrence of problems, and aligns expectations between the company and its clients.

5. **Leverage Technology for Efficiency**
   - **Solution**: Implement advanced CRM and help desk software that includes features for ticket tracking, automation of routine tasks, and proactive issue detection. Also, consider simplifying the user interface of the billing software to make it more user-friendly.
   - **Impact**: Decreases manual work, reduces error rates, and improves the speed and quality of service delivery.

Topic 2: Software Process Quality Metrics

The two software development teams have recently completed their projects. Both applied the same development tool and similar programming style.

Metrics code	Metrics name	Calculation Formula
ADES	Average Development Errors Severity (for the entire development process)	ADES = WNDE/NDE
ACES	Average Coding Errors Severity	ACES = WNCE/NCE

- NDE = Number of development errors.
- NCE = Number of coding errors.
- WNDE = Weighted number of development errors.
- WNCE = Weighted number of coding errors.

With above metrics defined, the following measures were supplied.

Data	Team A	Team B
NCE	154	91
NDE	223	206

1. **What additional data would you require to determine which of the teams achieved results with the better result? Justify your answer.**

Answer:

To ascertain which team produced superior outcomes, the following more information would be needed:

a. WNDE (Weighted number of development mistakes):

This would give an indication of how serious each team's development errors were. We may compute the Average Development Errors Severity (ADES) by knowing the weighted impact of the development errors (NDE) since we have their number.

b. WNCE (Weighted number of coding errors):

It is required to evaluate the seriousness of coding errors. The severity of these problems cannot be detected solely by looking at the number of coding errors (NCE).

The need for this extra information stems from the fact that the sheer quantity of errors (NDE and NCE) obscures the full picture of how they affect the development process. For example, fewer high-severity errors may have more negative effects than many low-severity errors. The weightings (WNDE and WNCE) would probably include how each error affects the system, how difficult it is to rectify, how long it takes to remedy, and how it affects the functionality or project timetable.

2. **After examining the metrics, what differences in software quality conception held by the team leaders may be concluded from the results? Explain why.**

Answer:

According to the metrics presented, it appears that one can deduce some information about the software quality between the two teams by examining the quantity of errors. The inference, therefore, might be merely surface level without the weighted severity of those errors. Still, the following are a few plausible interpretations:

a. **Method of Coding:**

Should Team B's NCE be noticeably lower than Team A's, this could suggest that they are more committed to developing error-free code or that their initial coding phase was more successful.

b. **Development Approach:**

If Team A has more development errors (NDE) than Team B, it may indicate that they follow less strict development procedures or that they take more chances, which may result in greater innovation but also more errors.

c. **Risk management:**

Differing approaches to risk management may also be reflected in the variation in the number of errors. A group that makes fewer mistakes may take a more cautious approach to progress, putting stability ahead of innovation.

d. **Quality Assurance:**

Variations may also exist in the methods used for quality assurance. A greater error count could indicate ineffective quality assurance procedures or indicate improved error detection and recording skills within the team.

Part II: READING

We introduced the software process quality assurance activities for conformance to the requirement specification and user needs, including evaluating software process and development environment for conformance, improving processes – corrective and preventive actions, software process assurance activities for external participants, software process quality metrics, software change control processes, and staff skills and knowledge – training and certification.

In this part, each group is required to complete the following.

1. **Read through the textbook chapters 18 through 23.**

Answer:

Chapter 18: Evaluation of Processes and Development Environment for Conformance

Evaluates the alignment of software processes and development environments with quality standards.

Chapter 19: Improvement Processes – Corrective and Preventive Actions

Covers strategies for correcting and preventing defects in software processes.

Chapter 20: Software Process Assurance Activities for External Participants

Assesses quality assurance activities involving external participants in software projects.

Chapter 21: Software Process Quality Metrics

Uses metrics to monitor and improve the quality of software development processes.

Chapter 22: Software Change Control Processes

Explores management and control of changes to software to maintain quality.

Chapter 23: Staff Skills and Knowledge – Training and Certification

Enhances software quality through staff training and professional certification.

2. **Select one software process quality assurance activity.**

Answer:

One notable software process quality assurance activity detailed in Chapter 19 is the "**Improvement Processes – Corrective and Preventive Actions**".

This activity is crucial for enhancing software process quality by identifying, addressing, and preventing the recurrence of defects and nonconformities within software development and maintenance processes.

3. **Search for at least two recent publications on the software product quality assurance activity you selected.**

Answer:

Paper 1: **Improvement Processes – Corrective and Preventive Actions**

Link: https://ieeexplore.ieee.org/abstract/document/8343603

Paper 2: Improvement of Key Problems of Software Testing in Quality Assurance

Link: https://arxiv.org/abs/1202.2506

4. **Read and discuss the articles.**

Answer:

Paper 1: Improvement Processes – Corrective and Preventive Actions

The steps of the corrective and preventative actions (CAPA) process are outlined in this article along with how it is put into practice. Facilitating ongoing enhancements in efficacy and productivity is a crucial goal of software quality assurance (SQA). One of the primary instruments now employed to accomplish this goal is the CAPA procedure. The following tasks are necessary for a CAPA process to run successfully: gathering and analyzing information; developing solutions and better techniques; putting improved techniques into practice; and monitoring CAPA operations. Both the presence of a large number of ad hoc team members and a permanent core organizational unit are necessary for the proper execution of CAPA activities. The Corrective Action Board (CAB) committee, which is the permanent CAPA body, initiates the CAPA process by screening material, designating members of targeted ad hoc CAPA teams, encouraging implementation, and monitoring the procedure.

Paper 2: Improvement of Key Problems of Software Testing in Quality Assurance

In order to guarantee that software projects adhere to accepted specifications, standards, and functioning without errors, the article addresses the function of quality assurance, or QA. From the beginning to the completion of the project, quality assurance (QA) is essential and covers procedures including software design, development, and multiple types of management (source code, change, configuration, and release).

The prevalent user attitudes, company culture, and insufficient testing procedures are among the major problems with current software methods. These problems give rise to serious difficulties such as testing brevity, shorter testing times, and inadequate documentation.

The goal of the study is to improve the overall quality and dependability of software development processes by suggesting solutions to these important issues in software testing inside QA. By encouraging comprehensive testing procedures, enhancing documentation, and modifying the cultural and attitudinal approaches inside project teams and organizations, these techniques aim to address the inadequacies that have been found.

5. **Summarize your readings as a reading report.**

Answer:

Key takeaway from Paper 1 [Improvement Processes – Corrective and Preventive Actions]:

- Continuous Improvement: To increase software quality assurance's efficacy and efficiency over time, CAPA procedures are crucial.
- Stages of CAPA: Information gathering, analysis, solution creation, implementation, and follow-up are some of the stages that are usually included in the CAPA process.
- Permanent Organizational Unit: The Corrective Action Board (CAB) committee, also known as a permanent core organizational unit, is essential to managing the CAPA process. The roles of the CAB committee are to screen information, designate ad hoc CAPA teams, encourage solution implementation, and monitor CAPA activities in order to initiate the CAPA process.
- Ad Hoc Team Members: Ad hoc team members participate in CAPA operations in addition to the permanent CAB committee. They offer their expertise in the phases of analysis, solution formulation, and implementation.
- Effective CAPA Operation: The CAPA process must be carried out correctly in order for it to be successful. This includes carrying out all of its required actions, which include careful information analysis, creating workable solutions, and rigorous follow-up.

Key takeaway from Paper 2 [Improvement of Key Problems of Software Testing in Quality Assurance]:

- The paper's major results on enhancing software testing in quality assurance draw attention to a number of important problems and offer appropriate remedies.
- The main issues that have been identified are user attitudes that hinder comprehensive testing, organizational cultures that prioritize speed above quality, and poor testing procedures that fall short of covering required scenarios and simulating real-world usage.
- These problems show up as harmful practices including testing hurriedly done to meet deadlines, short-cutting testing durations that compromise thoroughness, and inadequate documentation that makes it harder to replicate and comprehend problems.
- The paper suggests enforcing strict documentation standards to improve the efficacy of quality assurance efforts, fostering a corporate culture that prioritizes quality throughout the software development lifecycle, and implementing strong testing frameworks that guarantee extensive coverage as ways to address these issues.
- These tactics seek to strengthen software products' overall performance and dependability by fixing current flaws.