CSC585: Assignment-4

Submitted by: **Malika Hafiza Pasha**

Question 1: Chapter 18 – Evaluation of Processes

The organization operates a user support service for its major software systems.

a. **Suggest three process metrics for the user support services.**
b. **Suggest desired levels for the results of the suggested metrics that will conform to the requirements.**

Answer:

a. Suggested Process Metrics for User Support Services:

1. First Contact Resolution Rate (FCR):

Definition: This indicator calculates the proportion of support issues that are handled without requiring a follow-up during the initial user encounter. It shows how effective and capable the support staff is at solving issues quickly.

2. Average Response Time:

Definition: This statistic measures how long it usually takes the support staff to answer a user's initial question. Setting the standard for communication and problem-solving is crucial for ensuring consumer satisfaction.

3. Customer Satisfaction Score (CSAT):

Definition: This measure assesses how satisfied customers are with how their questions are answered. Usually, customer happiness is gauged by asking them to score their level of satisfaction on a scale via surveys.

b. Desired Levels for the Results of the Suggested Metrics:

1. First Contact Resolution Rate (FCR):

Desired Level: A high FCR should ideally be a sign of a knowledgeable and effective support staff. In most businesses, aiming for an FCR of 70–90% is seen as outstanding since it indicates that most issues are handled efficiently and without the need for escalation or follow-up communication.

2. Average Response Time:

Desired Level: In order to increase customer satisfaction, the objective should be to reduce this time as much as feasible. Aiming for an initial response time of no more than one or two hours could be a reasonable goal, given the average complexity of support issues. Less than thirty minutes should be the ideal response time for sectors that provide crucial services.

3. Customer Satisfaction Score (CSAT):

Desired Level: A CSAT of more than 80% is considered satisfactory in most cases, while a score of 90% or more indicates excellent support that meets or above the expectations of the user.

In order to make sure that the user support service runs smoothly and successfully satisfies user needs, it is imperative that certain metrics and their intended levels be reached. To promote continual improvement in the level of service quality and consumer satisfaction, they must to be routinely observed and evaluated. Depending on the nature of the software systems supported, user expectations, and industry norms, adjustments to the targets may be necessary.

Question 2: Chapter 19 – Improvement Processes – Corrective and Preventive Actions (CPAP)

Section 19.3.6 lists three main tasks of CPAP follow-up.

a. **What are the three tasks?**
b. **Explain in your own words the importance of the follow-up tasks to the success of the process.**

Answer:

a. Three Main Tasks of CPAP Follow-up:

1. Verification of Implementation: This work entails verifying that the scheduled corrective and preventative actions have been carried out in accordance with the action plans.

2. Effectiveness Checks: This task verifies that the actions are successfully addressing and resolving the stated inadequacies for which they were designed after they have been put into action. This could entail going over the problems again, keeping an eye on the relevant KPIs, and carrying out further audits or evaluations.

3. Documentation: accurate recording of every action performed, including the adjustments made, the outcomes of efficacy assessments, and any extra steps or adjustments that were necessary. This guarantees process transparency and traceability and acts as a record for future use.

b. Importance of the Follow-up Tasks to the Success of the Process:

1. Verification of Implementation:

Purpose: Verifies that all preventative and corrective actions are implemented in accordance with the plan. It serves as a compliance check to make sure that the actions are carried out correctly.

Importance: guarantees the implementation of all preventative and corrective actions in accordance with the plan. To avoid any errors in carrying out the actions, it serves as a compliance check.

2. Effectiveness Checks:

Purpose: to confirm that the measures that have been put into place are accomplishing the goals for which they were intended. To do this, expected and actual results must be compared.

Importance: To make sure that the remedial activities are not only put into place but also work and are successful in getting rid of or lessening the flaws or issues, effectiveness checks are essential. This stage guarantees ongoing process improvement and helps prevent problems from happening again.

3. Documentation:

Purpose: should keep a thorough and accurate record of all the activities performed, the reasoning behind them, how they were carried out, and the outcomes of the effectiveness assessments.

Importance: The historical record that documentation offers is essential for accountability and future reference. It offers documentation of compliance and due diligence in the corrective and preventative process and aids in the establishment of a knowledge base for resolving similar problems in the future.

Together, these follow-up duties guarantee that the corrective and preventative actions not only successfully address the immediate problems but also support the organization's systemic improvements and long-term quality. Over time, a strong quality assurance system that increases dependability, lowers errors, and boosts customer satisfaction can be built with the aid of this methodical follow-up procedure.

Question 3: Chapter 20 – Software Process Assurance Activities for External Participants

Employing readymade components and packages provides the contractor with major benefits, but involves risks with respect to carrying out a project.

a. List the main benefits to a contractor and explain in your own words the implications of each one.
b. List the main risks to a contractor and explain in your words the implication of each one.

Answer:

a. **<u>Benefits of Using Readymade Components and Packages for Contractors:</u>**

1. **Cost Reduction:**

Implication: The contractor might save development expenses greatly by incorporating ready-made components. This is due to the fact that it is typically more expensive to buy or license pre-existing, tested, and proven components than to construct comparable functions from scratch.

2. **Time Savings:**

Implication: Because off-the-shelf software components are pre-built and ready for integration, using them helps speed up the development process. This facilitates fulfilling project deadlines and shortening time-to-market, both of which are important in fields where technology is advancing quickly.

3. **Proven Reliability and Quality Assurance:**

Implication: Pre-tests and quality certificates of their own are frequently included with ready-made components. By depending on goods that have already had their performance and stability confirmed, using these components may increase the project's overall reliability.

4. **Access to Cutting-Edge Technology:**

Implication: Contractors do not have to possess in-depth knowledge of every field to utilize the newest technologies. Small and medium-sized businesses, who might lack the capacity to build complicated components internally, can especially benefit from this.

5. **Maintenance and Support:**

Implication: The vendor support and maintenance plans that are often included with readymade components might lessen the load of long-term maintenance that falls on the contractor's internal team. This lets the vendor manage upgrades and support issues, freeing the contractor to concentrate on project's essential tasks.

b. Risks of Using Readymade Components and Packages for Contractors:

1. **Dependency on Third-Party Vendors:**

Implication: When external components are used, one becomes dependent on outside suppliers for patches, upgrades, and support. There could be major setbacks to the project if the vendor decides not to maintain the product any longer or closes their doors.

2. **Integration Issues:**

Implication: Integrating pre-made components with the current system can be very difficult, particularly if the coding standards or APIs are not completely compatible. This could result in more overhead related to development and debugging, which could reduce some of the early cost and time advantages.

3. **Security Vulnerabilities:**

Implication: Pre-made software components may provide security issues, particularly if their source is unreliable or if they are not updated on a regular basis. Any vulnerability in the component's security puts the project at risk, increasing the risk of data breaches and compliance problems.

4. **Limited Customization:**

Implication: When using off-the-shelf software, the contractor frequently has to modify their specifications to match the capabilities of the component rather than creating a solution specifically tailored to the particular requirements of the project. User experience and functionality may suffer as a result.

5. **Licensing and Compliance Issues:**

Implication: Readymade software components are subject to licensing agreements that may impose continuing charges or particular restrictions. Legal and financial ramifications may result from misinterpreting these terms, particularly if the program is utilized in a way that is not permitted under the license.

Question 4: Chapter 21 – Software Process Quality Metrics

Section 21.4 and Section 21.5 list metrics for USC (User Support Center) and corrective maintenance services.

a. **Explain the difference between these services.**
b. **Justify the separate metric categories and actions (based on their differences).**

Answer:

a. Differences Between USC and Corrective Maintenance Services:

1. **User Support Center (USC):**

- **Purpose:** USC's primary goal is to give end consumers rapid support and assistance. This covers assistance with utilizing software programs, resolving user problems, and fixing typical operational issues that don't always call for changes to the code.
- **Activities:** User assistance, password resets, answering user inquiries, software usage advice, and small configuration changes are typical tasks. From a development perspective, the assistance is typically non-technical and does not entail changing the software's code.

- **Response Type:** The USC strives to minimize delay or inconvenience for its users by providing prompt responses that frequently include direct user contact.

2. **Corrective Maintenance Services:**

- **Purpose:** This service handles finding, diagnosing, and fixing software bugs that are usually found after the program has been deployed. These mistakes could be design flaws, code faults, or other flaws affecting the security, performance, or functioning of the product.
- **Activities:** In order to fix bugs that keep the software from working as intended, activities entail altering the code. Updates and patches may also be necessary for this in order to enhance software performance or adjust to new specifications.
- **Response Type:** Corrective maintenance responses are more structured and could not happen right away because they frequently call for more thorough testing and analysis to make sure that changes don't create new problems.

b. Justification for Separate Metric Categories and Actions:

1. **Different Objectives and Stakeholder Impact:**

- **USC Metrics:** User satisfaction, resolution efficacy, and response times are measured for each individual user by means of specific metrics for USC services. By gauging user satisfaction with the software and support services, these measures contribute to the users' continued productivity.
- **Corrective Maintenance Metrics:** Corrective maintenance metrics center on the efficacy, timeliness, and quality of the fixes for software flaws. These include metrics for bug fix timeframes, incident frequency associated with known problems, and failure recurrence rates for certain defects.

2. **Nature of Services Rendered:**

- **USC Actions:** The USC is taking steps to increase productivity and decrease downtime for users. It is essential to monitor metrics like average handle time, initial call resolution, and customer feedback scores.
- **Corrective Maintenance Actions:** The actions taken here are aimed at preserving and gradually raising the software's quality. Mean time to repair (MTTR), defect density, and the proportion of faults fixed prior to live deployment are examples of metrics.

3. **Impact on Software Lifecycle and Development Process:**

- **USC Metrics Influence:** Metrics have an impact on helpdesk resource management and direct user interaction tactics. They guarantee that users can use the software efficiently, which minimizes interruptions to operations.
- **Corrective Maintenance Metrics Influence:** These measurements affect software development life cycle procedures, software maintenance plans, and the long-term stability and quality of software.

Question 5: Chapter 22 – Software Change Control (SCC) Processes

The SCC committee assigns the task of examination of a SCR (Software Change Request) to one of the committee members.

a. **Why is there a need to examine the submitted SCR?**
b. **List types of expected examination findings that need to be corrected.**

Answer:

a. Why is there a need to examine the submitted SCR?

The examination of a submitted SCR is critical for several reasons:

1. **Ensure Relevance and Necessity:**

• To confirm that the suggested modification is pertinent and essential for improving the software product, making sure that it complies with the project's overarching objectives and operational requirements.

2. **Validate Completeness and Clarity:**

• To determine if the SCR is comprehensive and accurately outlines the change, including its goals, scope, and anticipated effects. This aids in properly comprehending the change and organizing its execution.

3. **Identify Impact:**

• To ascertain whether the suggested modification may have an effect on other system components, such as user experience, system functionality, and technological requirements. This aids in evaluating the hazards connected to the modification.

4. **Resource and Cost Evaluation:**

• Assessing the manpower, money, and time needed to put the change into effect. Analyzing if the expenses and resource allocation of the change are justified by its benefits is part of this.

5. **Compliance and Consistency Check:**

• To guarantee that the modification adheres to the current system architecture and coding standards and does not contravene any compliance requirements.

6. **Prioritization:**

• To rank the change request according to its urgency and strategic significance in relation to other SCRs and ongoing projects.

b. List types of expected examination findings that need to be corrected.

The examination of SCRs might reveal various issues that need to be addressed before approving the change:

1. **Ambiguities or Incompleteness:**

- SCRs that are imprecise, don't provide thorough justifications for the modifications, or don't accurately identify the application's impacted sections.

2. **High Impact on Existing Functionality:**

- Modifications that might impair current functionality or result in serious regression faults. Comprehensive impact analyses and risk management techniques are needed for this.

3. **Non-Compliance with Standards:**

- Before being implemented, proposed changes that don't adhere to technical, organizational, or regulatory criteria must be amended to do so.

4. **Resource Constraints:**

- The request may need to be modified or resources may need to be reallocated if the SCR calls for more resources than what is currently available.

5. **Poorly Defined Outcomes:**

- SCRs that are unclear about the anticipated results or benefits might need to be clarified in order to show how the modification would support the software's overarching goals.

6. **Overlapping Changes:**

- To reduce conflict and redundancy, SCRs that overlap with other planned or existing changes must be integrated or coordinated.

7. **Security Risks:**

- Modifications that bring about possible security flaws need to be made in order to sufficiently mitigate these risks.

8. **Performance Issues:**

- To guarantee adherence to performance benchmarks, proposed modifications that can impair software performance must be examined and optimized.

Question 6: Chapter 23 – Staff Skills and Knowledge – Training and Certification

Consider the certification requirement “mentor supervision.”

a. **Explain the unique contribution of supervision to the success of the certification process.**

b. **Suggest certification requirements that can be replaced, wholly or partially, by a mentor's supervision. List your arguments.**

Answer:

a. Mentor Supervision Contribution:

1. **Expert Guidance**: Mentors offer mentees specific knowledge and experience, assisting them in navigating challenging material and real-world applications. The successful integration of theoretical information from certification courses into practical skills is ensured by this individualized mentoring.
2. **Feedback and Improvement**: It takes constant feedback to learn and get better. Mentees can identify their areas of strength and growth with the help of mentors' constructive criticism and feedback, which is essential for both professional and personal growth.

3. **Experience Sharing**: Beyond the confines of conventional educational resources, mentors impart insightful knowledge, practical difficulties, and invaluable experiences. Through this experience, mentees are better able to comprehend and get ready for real-world scenarios that they may encounter in their careers.

4. **Networking and Professional Growth**: Mentees may be exposed via mentors to professional networks, which can present chances not normally found in traditional educational settings. For professional success and career progress, networking can be quite important.

5. **Motivation and Support**: The certification procedure can provide difficulties and demands. As a source of motivational support, mentors assist mentees in overcoming obstacles and staying goal-focused.

6. **Ethical and Professional Standards**: Mentors serve as role models for professional conduct and ethics, which are essential elements of many certification programs. Mentees pick up these ideals through imitation and observation.

b. Certification Requirements Influenced by Mentor Supervision:

1. **Practical Experience Requirement**:

   - **Replacement Argument**: A lot of certificates call for hours of verified real-world experience. This experience can be obtained in an organized setting with mentor supervision, which guarantees that the practice is carried out with an emphasis on learning objectives and quality.

2. **Project Submission**:
   - **Replacement Argument**: Continuous assessment and mentor feedback, with an emphasis on iterative improvements and real-time learning adjustments, might partially replace the requirement for project submissions in certifications involving project work (such as software development or engineering certifications).
3. **Ethics Training**:
   - **Replacement Argument**: Standardized assessments may not have the same impact as mentorship when it comes to assisting candidates pursuing certifications that involve an ethics component. Mentorship offers a dynamic and context-based approach to comprehending ethical challenges and acceptable professional behavior.
4. **Continuous Professional Development (CPD)**:
   - **Replacement Argument**: Mentor-guided professional development can offer continuous learning customized to the mentee's unique needs and career path in place of traditional CPD courses, making the learning process more applicable and relevant.
5. **Examinations**:
   - **Partial Replacement Argument**: Mentor-led assessments, in which mentors assess competence through discussions or practical demonstrations, can replace some test components, but not all of them. This gives a comprehensive picture of the mentee's abilities.