SELECTED swe523 EXAM QUESTIONS IN THE PAST YEARS

…….. is not one of the main project management activities.

a. Project planning

b. Risk management,

c. People management,

d. Proposal writing

**e. Testing activity**

.…………….is not of the main classes of attribute used to compute the multiplier M?

a. Product b. Computer c. Personnel d. Project **e. Business**

List the four main classes of attribute used to compute the multiplier M?

1. Product Attributes
2. Computer Attributes
3. Personnel Attributes
4. Project Attributes

critical path question???

exam question: The slacktimes at the activity C is .....days

In a TEYDEB project, ............... is a technical progress report prepared by company

a. AGY100 b. AGY200 **c. AGY300** d. AGY400 e. AGY350

2-3 NUMERIC QUESTIONS : For example,

A decision tree analysis is one of the powerful techniques for analyzing and quantifying the risks. Assume …. I will give a scenario and ask question ☺

The European Union (EU) project framework program does not include d. Military Technologies.

Here's why the other options are included in the EU framework program:

Early Research Council (ERC): Supports frontier research by individual researchers.

Network Excellence Centers (Networks of Excellence): Funds collaboration between research institutions across Europe.

Marie-Curie Actions: Promotes international mobility of researchers at all career stages.

FET Flagships: Supports large-scale, long-term research initiatives addressing societal challenges.

The EU focuses on civilian research and innovation

**Six additional possible risks that could arise in software projects:**

1. **Communication breakdowns:** Miscommunication between stakeholders (clients, developers, etc.) can lead to misunderstandings about requirements, deadlines, or functionalities.
2. **Third-party dependency issues:** Delays or problems with external libraries, frameworks, or services can impact project timelines and functionality.
3. **Data security vulnerabilities:** Unidentified or unaddressed security flaws can expose sensitive data or compromise system integrity.
4. **Scope creep without proper change management:** Uncontrolled addition of new features or functionalities can lead to schedule delays, budget overruns, and potential quality issues.
5. **Infrastructure limitations:** Insufficient hardware, software, or network capacity can hinder development, testing, or deployment of the software.
6. **Integration challenges:** Difficulty integrating different software components or systems can cause unexpected issues and delays.

These are just a few examples, and the specific risks will vary depending on the nature and complexity of the software project.

Fixed-price contracts, where the contractor bids a fixed price to complete a system development, may be used to move project risk from client to contractor. If anything goes wrong, the contractor has to pay. Suggest how the use of such contracts may increase the likelihood that product risks will arise.

Under what circumstances might a company justifiably charge a much higher price for a software system than the software cost estimate plus a reasonable profit margin?

Cost estimates are inherently risky, irrespective of the estimation technique used. Suggest four ways in which the risk in a cost estimate can be reduced.

Figure sets out a number of tasks, their durations and their dependencies. Draw a bar chart showing the project schedule. ...find ....

Explain how standards may be used to capture organizational wisdom about effective methods of software development. Suggest four types of knowledge that might be captured in organizational standards.

Describe 4 major factors that should be taken into account by engineers during the process of building a release of a large software system.

Describe 4 essential features that should be included in a tool to support change management processes.

Describe the 3 major difficulties that may arise when building a system from its components. What particular problems might occur when a system is built on a host computer for some target machine?

Describe three types of software process metric that may be collected as part of a process improvement process. Give one example of each type of metric.

Under what circumstances would you recommend the use of the staged representation of the CMMI?

What are the identified levels in the CMMI staged model

The CMMI staged model defines five levels of process maturity:

Initial: Processes are informal and reactive.

Repeatable: Basic project management practices are established to ensure consistent project performance.

Defined: Processes are documented and standardized across the organization.

Managed: Process performance is measured and controlled.

Optimizing: Continuous process improvement is a focus based on quantitative data.

By progressing through these levels, organizations can achieve higher levels of process maturity, leading to improved software quality, reduced development costs, and faster time-to-market.

Selected other questions

**Chapter 22: Project management**

1. What are important differences between software project management and other types of project management?

2. List 4 fundamental project management activities.

* Proposal writing
* Project planning
* People management
* Reporting
* Risk management

3. What are three related categories of risk?

* Project risks
* Product risks
* Business risks

4. Suggest 4 risks that may threaten the success of a software project?

* Risk identification
* Risk analysis
* Risk planning
* Risk monitoring

5. Give 2 examples of technology risks that may arise in a software project.

* The database used in the system cannot process as many transactions per second as expected. (1)
* Reusable software components contain defects that mean they cannot be reused as planned. (2)

6. What is involved in risk monitoring?

* Assess each identified risks regularly to decide whether or not it is becoming less or more probable. (Change in degree)
* Also assess whether the effects of the risk have changed. (Change in effect)
* Each key risk should be discussed at management progress meetings. (Discussion)

7. What are the four critical factors in people management?

* consistency,
* respect,
* inclusion,
* honesty

8. What are the different levels in the human needs hierarchy?

* Basic needs,
* Human needs,
* Social needs

9. What factors might be considered when selecting people for a software development team?

* Budget
* Group cohesion
* Technical skill variation

10. What are the key factors that influence the effectiveness of group communications?

* status of group members,
* the size of the group,
* the gender composition of the group,
* personalities and available communication channels.

**Chapter 23: Planning and estimation Your name:**

1. What estimates have to be produced by software project managers?

* *Cost*
  + *product, project and process attributes*
* *Effort*

2. What factors should be considered when deciding on a price for software?

* *Market opportunity, cost estimate uncertainty, contractual terms, req. volatility, financial health*

3. What is included in a quality plan and a validation plan?

* Quality procedures and standards
* Approach, resources, and schedule

4. What are the principal activities in the project scheduling process?

* Identify activities
* Identify dependencies
* Estimate resources for activities
* Allocate people to activities
* Create project charts

5. How are bar charts used in project scheduling?

* They show the schedule as activities or resources against time.

6. What are the most important differences between agile planning and plan-based development?

* Unlike plan-driven approaches, the functionality of these increments is not planned in advance but is decided during the development

7. Briefly describe two types of cost estimation techniques?

* Experience-based techniques
* Algorithmic cost modeling

8. What is the generic formula used in algorithmic cost modelling to compute the effort required to develop some software?

* Effort = A ´ SizeB ´ M

9. What are the estimation models used in COCOMO II.?

* Application composition model
* Early design model
* Reuse model
* Post-architecture model

10. What are the four classes of attribute used to compute the multiplier M?

* Product Attributes
* Computer Attributes
* Personnel Attributes
* Project Attributes

**Chapter 24: Quality management**

1. What are the three main quality management activities?

* Quality planning,
* quality control,
* quality assurance ?

2. What sections does Humphrey suggest should be included in a quality plan?

a. Introduction

b. Standards and Procedures

c. Reviews and Audits:

d. Tools and Techniques:

e. Training:

f. Responsibilities:

3. Briefly describe the two types of standard that may be defined during the quality management process?

* **Process Standards:** These define the activities, procedures, and methods that should be followed during software development to achieve quality. Examples include coding standards, testing procedures, and defect tracking processes.
* **Product Standards:** These define the characteristics that the software product itself should possess. They can be functional (what the software should do) or non-functional (performance, usability, reliability, etc.).

4. What is ISO 9001 and what does it describe?

ISO 9001 is a generic quality management system (QMS) standard that can be applied to any organization, including those developing software. It doesn't specify software development practices directly; instead, it focuses on establishing a framework for managing processes to ensure consistent quality output. However, organizations can adapt ISO 9001 to include software-specific quality practices within their QMS.

A QMS framework that includes software development

**Software Quality Management Standards (Sommerville)**

Based on Ian Sommerville's Software Engineering book, here's a breakdown of your questions:

**3. Types of Standards in Quality Management:**

Sommerville outlines two main types of standards used in quality management:

* **Process Standards:** These define the activities, procedures, and methods that should be followed during software development to achieve quality. Examples include coding standards, testing procedures, and defect tracking processes.
* **Product Standards:** These define the characteristics that the software product itself should possess. They can be functional (what the software should do) or non-functional (performance, usability, reliability, etc.).

**4. ISO 9001 and Software Development:**

ISO 9001 is a generic quality management system (QMS) standard that can be applied to any organization, including those developing software. It doesn't specify software development practices directly; instead, it focuses on establishing a framework for managing processes to ensure consistent quality output. However, organizations can adapt ISO 9001 to include software-specific quality practices within their QMS.

**5. Stages in Software Inspection Process:**

Sommerville describes a formal software inspection process with several stages:

1. **Preparation:** The inspection team is briefed on the software and receives relevant documentation.
2. **Overview Meeting:** The team gets a high-level overview of the software and identifies potential areas of focus.
3. **Individual Examination:** Each inspector independently examines the assigned code or documentation, looking for defects.
4. **Meeting:** The team discusses identified defects, reaching consensus on their validity and severity.
5. **Rework:** The developer responsible for the code addresses the identified defects.
6. **Follow-up Meeting:** The team verifies that the defects have been properly corrected.

**6. Classes of Faults in Inspection Checklists:**

Inspection checklists typically categorize faults based on their type or impact:

* **Logic Errors:** Mistakes in program logic that lead to incorrect behavior.
* **Data Errors:** Errors in data structures or manipulation that can cause unexpected results.
* **Control Flow Errors:** Errors in program flow that lead to incorrect execution sequences.
* **Interface Errors:** Incompatibility issues between components or with external systems.
* **Usability Errors:** Issues that make the software difficult or frustrating to use.

**7. Uses of Software Product Measurements:**

Sommerville highlights two key uses of software product measurements:

* **Process Improvement:** By measuring various aspects of the development process (e.g., defect rates, testing time), organizations can identify areas for improvement and enhance their software development practices.
* **Project Management:** Metrics can be used to estimate project effort, track progress, and identify potential risks early on. This allows for better project planning and resource allocation.

**8. Limited Use of Software Metrics:**

Several factors contribute to the limited widespread use of software metrics in industry, according to Sommerville:

* **Difficulty in Defining Meaningful Metrics:** Defining metrics that accurately reflect software quality or development effort can be challenging.
* **Data Collection Overhead:** Tracking and collecting relevant metrics can be time-consuming and resource-intensive.
* **Lack of Standards:** The lack of standardized metrics across the industry makes it difficult to compare data and benchmark performance against competitors.
* **Focus on Short-Term Goals:** Short-term project pressures often overshadow the long-term benefits of process improvement using metrics.

By addressing these challenges, the use of software metrics can be enhanced to provide valuable insights for improving software development practices.

9. What are the key stages in the product measurement process?

1. *Quality Management Activities*
2. *Standards in Quality Management*
3. *Software Inspection Process*
4. *Classes of Faults in Inspection Checklists*
5. *Uses of Software Product Measurements*
6. *Limited Use of Software Metrics*
7. *Key Stages in the Product Measurement Process*

10. What are the object-oriented metrics used in the CK OO metrics suite.

* Weighted Methods per Class (WMC)
* Depth of Inheritance Tree (DIT)
* Number of Children (NOC)
* Coupling Between Objects (CBO)
* Response For a Class (RFC)
* Lack of Cohesion in Methods (LCOM)