SRM VALLIAMMAI ENGINEERING COLLEGE

(An Autonomous Institution) SRM

Nagar, Kattankulathur – 603 203

DEPARTMENT OF CYBER SECURITY

QUESTION BANK

VIII SEMESTER

1923803 Software Quality

Regulation – 2019

Academic Year 2025 – 2026(Even Semester)

Prepared by

E.Rajkumar, Assistant Professor / CYS

SRM VALLIAMMAI ENGINEERING COLLEGE

(An Atonomous Insstitution)

SRM Nagar, Kattankulathur – 603 203.

DEPARTMENT OF CYBER SECURITY

QUESTION BANK

SUBJECT: 1908301 Software Quality

SEM / YEAR VIII Sem / IV Year

UNIT – I INTRODUCTION			
Software Process Assessment overview – Assessment phases – Assessment principles – Assessment conduct – Implementation consideration – Quality management – Quality assurance plan – Considerations – Verification and Validation.			
PART – A			
Q.No	**Questions**	**BT Level**	**Competence**
1.	Define Software Process Assessment.	BTL1	Remembering
2.	List the phases involved in Software Process Assessment.	BTL1	Remembering
3.	Mention the key principles guiding Software Process Assessment.	BTL1	Remembering
4.	What is the typical conduct during Software Process Assessment?	BTL1	Remembering
5.	What are the factors should be considered during the implementation phase of Software Process Assessment?	BTL1	Remembering
6.	List the key aspects of Quality Management addressed in Software Process Assessment.	BTL1	Remembering
7.	Distinguish between Software Process Assessment and Quality Assurance.	BTL2	Understanding
8.	Interpret the key phases involved in a Software Process Assessment.	BTL2	Understanding
9.	Differentiate between Verification and Validation in the context of software development.	BTL2	Understanding

10.	Discuss the importance of Quality Management in the software development life cycle.	BTL2	Understanding
11.	What are the main phases of a Software Process Assessment?	BTL3	Applying
12.	What are the assessment principles commonly applied in Software Process Assessments interpret it with examples.	BTL3	Applying
13.	Examine how is the assessment conduct structured to ensure objectivity in Software Process Assessments?	BTL3	Applying
14.	Compare the assessment conduct in formative and summative Software Process Assessments.	BTL4	Analyzing
15.	Analyze the role of organizational culture in influencing implementation considerations for Software Process Assessments.	BTL4	Analyzing
16.	Examine how Quality Management contributes to continuous improvement in Software Development Processes through assessment.	BTL4	Analyzing
17.	Why is the assessment phase crucial in Software Process Assessment, and how does it contribute to informed decision making for process improvement?	BTL5	Evaluating
18.	Compare the roles of verification and validation in the software development lifecycle, highlighting their distinct contributions to ensuring Software Quality.	BTL5	Evaluating
19.	Describe two key phases in Software Process Assessment and their roles in enhancing software development practices.	BTL6	Creating
20.	Explain the significance of a Quality Assurance Plan in maintaining high quality standards throughout the software development lifecycle.	BTL6	Creating
21.	Compare the key distinctions between Quality Management and quality assurance in software development.	BTL3	Applying
22.	Explain the significance of verification and validation in Software Process Assessment and compare their roles in ensuring process effectiveness.	BTL4	Analyzing
23.	Interpret the principles that guide the conduct of a Software Process Assessment.	BTL2	Understanding
24.	What are the considerations are important when implementing a Quality Assurance plan in software development?	BTL5	Evaluating
	PART B		
1.	Define Software Process Assessment and list its key phases. (13)	BTL1	Remembering
2.	List and explain the principles guiding Software Process Assessment.(13)	BTL1	Remembering
3.	Mention key considerations for Quality Assurance Plan implementation in Software Process Assessment. (13)	BTL1	Remembering
4.	(i) What is a Software Process Assessment, and how does it contribute to the overall software development lifecycle? (7)	BTL1	Remembering

	(ii) Enumerate the key phases involved in a typical Software Process Assessment, outlining the sequence of activities undertaken. (6)		
5.	Distinguish between Software Process Assessment and Quality Assurance Plan.(13)	BTL2	Understanding
6.	Interpret the Assessment Phases in Software Process Assessment.(13)	BTL2	Understanding
7.	Differentiate between Verification and Validation in the context of Quality Management.(13)	BTL2	Understanding
8.	Identify and examine key phases in a Software Process Assessment, highlighting their significance in evaluating and enhancing Software Development Processes.(13)	BTL3	Applying
9.	(i) Identify the crucial considerations that need to be addressed during the implementation of a Software Process Assessment to ensure its effectiveness. (7) (ii) Examine the role of Verification and Validation in the context of software development, and how they contribute to ensuring the quality of the final software product. (6)	BTL3	Applying
10.	Compare the assessment phases in Software Process Assessment and analyze their respective roles in ensuring effective Quality Management during software development.(13)	BTL4	Analyzing
11.	Analyze the assessment principles employed in Software Process Assessment, and explain how they contribute to the establishment of a robust quality assurance plan in the software development lifecycle.(13)	BTL4	Analyzing
12.	Explain the considerations involved in the implementation of Software Process Assessment, and compare the key factors influencing successful verification and validation processes in the context of quality management.(13)	BTL4	Analyzing
13.	Why is the assessment phase crucial in the Software Process Assessment, and what justifies its significance in improving Software Development Processes?(13)	BTL5	Evaluating
14.	In designing a Quality Management framework, what key considerations should be prioritized to establish a robust and adaptable system for continuous improvement in Software Development Processes?(13)	BTL6	Creating
15.	(i) Discuss the fundamental principles that guide a Software Process Assessment and explain their significance in improving Software Development Processes. (7) (ii) Distinguish between Quality Management and Quality Assurance in the context of software development, highlighting their respective roles and objectives. (6)	BTL2	Understanding

16.	Identify and examine crucial considerations in the implementation of a Software Process Assessment, exploring factors that contribute to its effectiveness in driving process improvement. (13)	BTL3	Applying
17.	Why should assessment principles be established before conducting a Software Process Assessment, and what justifies their role in ensuring a comprehensive and unbiased evaluation? (13)	BTL5	Evaluating
	PART – C		
1.	How does the assessment phase differ from the implementation consideration phase in a Software Process Assessment, particularly in terms of their objectives and activities? (15)	BTL5	Evaluating
2.	(i) How does the assessment conduct phase differ from the assessment Principles phase in a Software Process Assessment? (8) (ii) Why is Quality Management an integral part of software development, and how does it contribute to overall project success? (7)	BTL6	Creating
3.	(i) Develop a brief outline of key considerations that should be included in a Quality Assurance plan for a software development project. (8) (ii) Design considerations play a crucial role in software implementation. Why is it essential to carefully plan and design before initiating the implementation phase in software development? (7)	BTL5	Evaluating
4.	In the context of Software Process Assessment, elaborate on the key principles that organizations should focus on during the assessment conduct phase, emphasizing their role in ensuring a thorough and effective evaluation. (15)	BTL6	Creating
5.	When designing a quality assurance plan for software development, what key considerations should be taken into account to ensure its alignment with overall Quality Management goals and standards? (15)	BTL6	Creating
	UNIT – II CONFIGURATION MANAGEMENT		
	Need for configuration Management – Software product nomenclature – Configuration Management functions – baselines – responsibilities – need for Automated Tools – plan – SCM support functions – The requirement phase Design control – The implementation phase – Test phase – SCM Tools – Configuration accounting and audit.		
	PART – A		
Q.No	**Questions**	**BT Level**	**Competence**
1.	Define Configuration Management.	BTL1	Remembering
2.	List Configuration Management Functions.	BTL1	Remembering
3.	Distinguish between Software Configuration Management (SCM) and Software Product Nomenclature.	BTL2	Understanding
4.	Interpret the role of baselines in the context of Configuration Management functions.	BTL2	Understanding

5.	Differentiate the responsibilities of various stakeholders in the requirement phase of Configuration Management.	BTL2	Understanding
6.	Identify the primary purpose of Configuration Management in software development.	BTL3	Applying
7.	Compare the functions of Configuration Management in the requirement phase to those in the design control phase of software development.	BTL4	Analyzing
8.	Analyze the significance of configuration accounting in contrast to configuration audit within the context of Software Configuration Management.	BTL4	Analyzing
9.	Why is Configuration Accounting and Audit considered essential in Configuration Management?	BTL5	Evaluating
10.	What are responsibilities in Configuration Management?	BTL6	Creating
11.	Define the need for Automated Tools in SCM.	BTL1	Remembering
12.	List SCM support functions.	BTL1	Remembering
13.	Interpret the significance of Design Control in the overall Configuration Management plan during the Test Phase.	BTL1	Remembering
14.	Why do we need the baselines fundamental in Configuration Management responsibilities during the Implementation Phase?	BTL1	Remembering
15.	Discuss the need for Automated Tools in Software Configuration Management during the Implementation Phase.	BTL2	Understanding
16.	Illustrate the significance of Software Process Assessment in the context of Configuration Management functions.	BTL3	Applying
17.	Examine the responsibilities associated with Configuration Management in the requirement phase of software development.	BTL3	Applying
18.	Examine the need for Automated Tools in managing software product nomenclature and configuration changes, emphasizing their advantages.	BTL4	Analyzing
19.	Why is Configuration Management crucial in the requirement phase of software development?	BTL5	Evaluating
20.	Design a role and importance of software product nomenclature in the design phase of configuration management.	BTL6	Creating
21.	Compare the need for Automated Tools in Configuration Management during the implementation phase and the test phase.	BTL3	Applying
22.	Explain the critical role of Software Process Assessment in configuration management, specifically outlining their responsibilities in maintaining stability during the software test phase.	BTL4	Analyzing
23.	Distinguish between Configuration Accounting and Audit in the context of SCM support functions.	BTL2	Understanding
24.	Why Automated Tool utilization is preferable in Configuration Management planning compared to manual methods?	BTL5	Evaluating

PART – B			
1.	Define Configuration Management and list its key functions in the Software Development Process. (13)	BTL1	Remembering
2.	List and mention the key responsibilities of individuals involved in Configuration Management during the software product development life cycle. (13)	BTL1	Remembering
3.	Distinguish between Software Product Nomenclature (SPN) and baselines in Configuration Management, highlighting their respective roles in maintaining software integrity. (13)	BTL2	Understanding
4.	Interpret the functions of Configuration Management (CM) during the Test Phase, emphasizing its impact on ensuring accurate testing environments and the successful execution of test procedures. (13)	BTL2	Understanding
5.	Why is the concept of Software Process Assessment important in the context of Configuration Management during software development? (13)	BTL3	Applying
6.	Can you provide practical examples of how Software Process Assessment are applied to establish stable reference points in different phases of the software life cycle? (13)	BTL3	Applying
7.	Why is Configuration Management essential in software development? (13)	BTL5	Evaluating
8.	What is the need for Automated Tools in Configuration Management, and how do they support the overall Software Development Process plan? (13)	BTL1	Remembering
9.	(i) Define Configuration Management (CM) and its significance in software development. (7) (ii) List and briefly explain the key functions of Configuration Management in a software development lifecycle. (6)	BTL1	Remembering
10.	Differentiate the roles and responsibilities assigned in Configuration Management from those in Design Control, discussing how effective assignment of these roles contributes to successful software development projects. (13)	BTL2	Understanding
11.	Compare the roles of Configuration Management in the requirement and implementation phases of software development, highlighting distinct focuses and activities. (13)	BTL4	Analyzing
12.	Analyze the significance of configuration Software Process Assessment in software development, emphasizing their contributions to version control, change management, and overall product stability. (13)	BTL4	Analyzing
13.	Examine the impact of Automated Tools on Configuration Management during the design control phase, emphasizing their role in enhancing efficiency, accuracy, and collaboration among development teams. (13)	BTL4	Analyzing

14.	Develop a concise question probing the essential aspects of software product nomenclature and its impact on maintaining consistency and clarity in communication throughout the software development lifecycle.(13)	BTL6	Creating
15.	(i) Discuss the concept of Software Process Assessment in Configuration Management and their role in version control during software development. (7) (ii) Distinguish between the responsibilities of software developers and Configuration Management personnel in maintaining code integrity and version control. (6)	BTL2	Understanding
16.	(i) Compare the necessity of Automated Tools in Configuration Management with manual approaches, highlighting advantages and potential drawbacks. (7) (ii) Identify and briefly discuss the SCM support functions that Automated Tools provide throughout the Software Development Process. (6)	BTL3	Applying
17.	Discuss the significance of Software Product Nomenclature in Configuration Management and its role in enhancing communication and understanding within development teams. (13)	BTL5	Evaluating
	PART C		
1.	Develop a strategy for Automated Tools in Software Configuration Management (SCM) and explain their significance in the requirement and design phases of software development. (15)	BTL6	Creating
2.	What is the primary purpose of Configuration Management in software development, and how does it contribute to the overall Software Development Process? (15)	BTL5	Evaluating
3.	(i) Explain the significance of Configuration Management in the software development life cycle, emphasizing its role in ensuring version control, traceability, and systematic handling of changes. (8) (ii) Compare and contrast manual Configuration Management processes with Automated Tools, highlighting the advantages and disadvantages of each approach in terms of efficiency, accuracy, and scalability. (7)	BTL5	Evaluating
4.	Design a Configuration Management approach highlighting the critical role of Software Process Assessment in ensuring stability, focusing on SCM support functions during the test phase. (15)	BTL6	Creating
5.	(i) Elaborate on the need for establishing Software Process Assessment in Configuration Management and how they contribute to project stability, consistency, and risk mitigation throughout the software development life cycle. (8) (ii) Compare the responsibilities of Configuration Management teams in the requirement phase, design control, implementation phase, and test phase, emphasizing the evolving nature of their roles and the impact on project success. (7)	BTL5	Evaluating
	UNIT – III SOFTWARE STANDARDS AND INSPECTION		

Definitions – Reason for software standards – Benefits – Establishing standards – Guidelines – Types of reviews – Inspection of objectives – Basic inspection principles – The conduct of inspection – Inspection training.			
PART – A			
1.	Why are software standards essential? Justify.	BTL 5	Evaluating
2.	Why is inspecting objectives critical in software reviews? Justify.	BTL 5	Evaluating
3.	What is the primary purpose of software standards?	BTL 1	Remembering
4.	Develop a definition of software standards and outline one primary reason why they are essential in software development.	BTL 6	Creating
5.	How do software standards differ from guidelines in terms of enforceability and flexibility?	BTL 4	Analyzing
6.	Discuss the importance of Inspection Training in the software development lifecycle.	BTL 2	Understanding
7.	Define the benefits associated with adhering to software standards.	BTL 1	Remembering
8.	List two methods for establishing software standards.	BTL 1	Remembering
9.	Mention the different types of reviews in the context of software development.	BTL 1	Remembering
10.	Design two distinct benefits that arise from the establishment and adherence to software standards in a development environment.	BTL 6	Creating
11.	Discuss the reason for implementing software standards in development projects.	BTL 2	Understanding
12.	Differentiate between Benefits and Establishing Standards in the context of software development.	BTL 2	Understanding
13.	Discuss the types of reviews in the Software Development Process.	BTL 2	Understanding
14.	Analyze the distinct advantages of software standards in ensuring consistency and guidelines in fostering adaptability.	BTL 4	Analyzing
15.	Identify the primary reason for implementing software standards.	BTL 3	Applying
16.	What are the basic principles underlying software inspection in development?	BTL 1	Remembering
17.	How does inspection training contribute to the effectiveness of the Software Development Process?	BTL 1	Remembering
18.	Illustrate a key benefit associated with the following software standards in development projects.	BTL 3	Applying
19.	Examine the differences in approaches to establishing software standards through formal documentation and enforcement compared to guidelines relying on communication and voluntary adherence.	BTL 4	Analyzing
20.	Examine the steps involved in establishing software standards within an organization.	BTL 3	Applying
21.	Distinguish between Inspection of Objectives and Basic Inspection Principles in software inspection.	BTL 2	Understanding
22.	Compare the formalized process of inspection with the broader term "review" in software development, highlighti key distinctions in their nature and scope.	BTL4	Analyzing

23.	Compare the objectives of different types of reviews in the Software Development Process.	BTL 3	Applying
24.	What is inspection training crucial in the context of software development?	BTL5	Evaluating
	PART B		
1.	(i) Define software standards and discuss their significance in the software development life cycle. (7) (ii) List two key benefits of adhering to software standards and distinguish between their internal and external impacts on a development project. (6)	BTL 1	Remembering
2.	Compare the benefits of software standards and the reasons for establishing them in the development process, highlighting their impact on Software Quality and maintenance. (13)	BTL 4	Analyzing
3.	Define software standards and provide a brief list of their essential components. (13)	BTL 1	Remembering
4.	Discuss the reasons for implementing software standards and outline the benefits they bring to the Software Development Process. (13)	BTL 1	Remembering
5.	(i) Discuss the guidelines that organizations should follow when establishing software standards and identify the challenges they may encounter in the process. (7) (ii) Compare the types of reviews commonly used in software development, emphasizing their roles and significance in ensuring Software Quality. (6)	BTL 2	Understanding
6.	Why are software standards essential in the development process, and how do they contribute to improved Software Quality? Justify. (13)	BTL 5	Evaluating
7.	Explain the role of design in realizing benefits like increased interoperability and easier maintenance within the context of adhering to software standards. (13)	BTL 6	Creating
8.	Distinguish between guidelines and standards in the context of software development, highlighting their respective roles. (13)	BTL 2	Understanding
9.	What is the significance of inspection training in ensuring the effectiveness of Software Development Processes? (13)	BTL 1	Remembering
10.	Analyze the types of reviews in the software development lifecycle, emphasizing their respective roles and significance in detecting defects and ensuring software reliability. (13)	BTL 4	Analyzing
11.	Compare different types of software reviews, emphasizing the unique characteristics of formal and informal review processes. (13)	BTL 2	Understanding
12.	Examine the basic inspection principles and guidelines for conducting effective software inspections, emphasizing their role in improving code quality and preventing errors in the development process. (13)	BTL 4	Analyzing
13.	(i) Identify and discuss three basic inspection principles that underpin the effectiveness of the inspection process in software development. (7)	BTL 3	Applying

	(ii) Discuss the importance of inspection training for development teams, and compare it with other forms of training in the software industry, highlighting its unique contributions. (6)		
14.	Discuss the basic inspection principles in the context of software development and their significance in ensuring Software Quality. (13)	BTL 3	Applying
15.	Interpret the key principles that guide the inspection of software objectives, emphasizing their role in quality control. (13)	BTL2	Understanding
16.	How can organizations implement effective training programs for software inspection to enhance overall development quality? (13)	BTL3	Applying
17.	Why it is important to establish clear guidelines for software development standards? Justify. (13)	BTL5	Evaluating
	PART C		
1.	Why is the conduct of inspections crucial in ensuring Software Quality, and how do basic inspection principles contribute to the overall success of the inspection process? (7) Justify the necessity of inspection training in the context of software development, and develop key criteria for effective training programs aimed at improving inspection proficiency. (8)	BTL 5	Evaluating
2.	Explain the significance of design in software development, highlighting its role in ensuring compliance with established software standards. (15)	BTL 6	Creating
3.	Explain in detail about the various types of standards available in Software Quality Standards. Discuss in detail about SEPG Groups and what standards it recommends in the software standards.(15)	BTL 5	Evaluating
4.	(i) Develop a concise definition of software standards and explain why they are essential in the Software Development Process. (8) (ii) Design a set of guidelines for establishing effective software standards, and justify the importance of adhering to these standards in the development lifecycle. (7)	BTL 6	Creating
5.	Why is the establishment of software standards necessary for effective project management, and how does it contribute to successful software development projects? Justify. (15)	BTL 5	Evaluating
	UNIT – IV TESTING AND MANAGING SOFTWARE QUALITY		
Testing principles – Types – Planning – Development – Execution and reporting – Tools and methods – Real Time testing – Quality Management paradigm – Quality motivation – Measurement criteria – Establishing a Software Quality program – Estimating Software Quality.			
	PART – A		
1.	Define Software Testing Principles.	BTL 1	Remembering
2.	Develop and design two key principles that guide effective software testing.	BTL 6	Creating
3.	Identify two principles of software testing.	BTL 3	Applying

4.	Develop and design the types of tools commonly used in software testing, emphasizing their role in the testing process.	BTL 6	Creating
5.	Interpret the concept of Real Time Testing and its relevance in ensuring the responsiveness and reliability of software applications.	BTL 2	Understanding
6.	List two types of software testing.	BTL 1	Remembering
7.	Discuss the role of Tools and Methods in software testing, and how they contribute to the efficiency of the testing process.	BTL 2	Understanding
8.	Compare white box and black box testing techniques in software testing.	BTL 4	Analyzing
9.	Illustrate two types of testing in the software development life cycle.	BTL 3	Applying
10.	Examine the importance of planning in software testing.	BTL 3	Applying
11.	Mention the key components of testing planning.	BTL 1	Remembering
12.	Differentiate between Development and Execution phases in the software testing life cycle, highlighting their specific roles.	BTL 2	Understanding
13.	What is the role of development in the software testing process?	BTL 1	Remembering
14.	Analyze the significance of test planning in the software development life cycle, highlighting its impact on overall project success.	BTL 4	Analyzing
15.	Examine the role of tools and methods in software testing, emphasizing their contribution to efficient and effective test execution.	BTL 4	Analyzing
16.	Why is planning an essential principle in software testing?	BTL 5	Evaluating
17.	Why is real time testing significant in the software development life cycle?	BTL 5	Evaluating
18.	Define the concept of Real Time testing.	BTL 1	Remembering
19.	Interpret the significance of Planning in the software testing process and its impact on overall project success.	BTL 2	Understanding
20.	What tools and methods are commonly used in software testing?	BTL 1	Remembering
21.	Distinguish between Testing Principles and Types in software testing.	BTL 2	Understanding
22.	Explain the key differences between Real Time testing and traditional testing approaches, considering their implications for system reliability and responsiveness.	BTL4	Analyzing
23.	Compare manual and automated testing methods.	BTL 3	Applying
24.	Why is the execution and reporting phase crucial in the software testing process?	BTL5	Evaluating
PART B			
1.	List and briefly explain three principles of software testing. (13)	BTL 1	Remembering
2.	Examine the role of Quality Management paradigms in ensuring Software Quality, and discuss their impact on the development process. (13)	BTL 3	Applying
3.	Mention four types of testing techniques and provide a concise description of each. (13)	BTL 1	Remembering

4.	Compare testing principles and the Quality Management paradigm, highlighting their roles in ensuring software robustness. (13)	BTL 4	Analyzing
5.	Analyze testing types' contribution to Software Quality, emphasizing their diverse methodologies and effectiveness. (13)	BTL 4	Analyzing
6.	(i) Distinguish between test planning and test development, highlighting the key activities involved in each phase. (7) (ii) Interpret the significance of real time testing in the context of software development, emphasizing its impact on system performance. (6)	BTL 2	Understanding
7.	(i) Examine the role of tools and methods in software testing, providing examples of popular tools and their applications. (7) (ii) Compare the Quality Management paradigm with traditional project management approaches, outlining key differences in their focus and objectives. (6)	BTL 3	Applying
8.	What are the key elements to consider when planning a software testing process? Illustrate with examples. (13)	BTL 1	Remembering
9.	(i) List and briefly explain three principles of software testing that guide the testing process. (7) (ii) Mention and describe two types of testing methodologies commonly used in software development. (6)	BTL 1	Remembering
10.	Distinguish between development testing and acceptance testing, highlighting their respective purposes in the software development life cycle. (13)	BTL 2	Understanding
11.	Why is real time testing crucial? Justify its impact on Software Quality by promptly addressing and resolving issues during development**.** (13)	BTL 5	Evaluating
12.	Develop concise measurement criteria for evaluating a Software Quality program, focusing on key metrics for effective assessment. (13)	BTL 6	Creating
13.	Examine the significance of planning in testing, elucidating its role in risk mitigation and process efficiency. (13)	BTL 4	Analyzing
14.	Interpret the significance of using tools and methods in software testing, and provide examples of commonly used tools. (13)	BTL 2	Understanding
15.	Differentiate between manual and automated testing, discussing scenarios where each approach is most effective. (13)	BTL2	Understanding
16.	Illustrate the importance of real time testing in the context of software development, and provide examples of real time testing scenarios. (13)	BTL3	Applying
17.	Justify the use of specific testing tools, explaining their role in enhancing accuracy and reliability. (13)	BTL5	Evaluating
PART C			
1.	Compare the principles of black box testing and white box testing, justifying when and why each approach is more suitable for specific types of software applications. (15)	BTL 5	Evaluating

2.	Design a strategy for real time testing in a mission critical system, considering factors such as data integrity, response time, and system reliability, and explaining how this strategy aligns with overall project goals. (15)	BTL 6	Creating
3.	(i) Design a comprehensive Software Quality program, outlining the key components and steps involved in ensuring a high level of Software Quality throughout the development life cycle. (7) (ii) Develop a strategy for estimating Software Quality, considering measurement criteria and factors that contribute to the overall success of a Software Quality assurance initiative. (8)	BTL 6	Creating
4.	Justify the integration of risk-based testing into the software testing process, outlining how it enhances the identification of critical areas and contributes to effective resource allocation. (15)	BTL 5	Evaluating
5.	(i) Compare manual testing and automated testing, justifying when each approach is more suitable in the software development life cycle. (7) (ii) Justify the importance of incorporating real time testing in the Software Development Process, highlighting its impact on overall system reliability and user satisfaction. (8)	BTL 5	Evaluating
UNIT – V DEFECT PREVENTION			
Principles of Software Defect Prevention – Process changes for Defect Prevention – Defect prevention considerations – Managements role – framework for Software Process Change – Managing resistance to Software Process Change – Case studies.			
PART- A			
1.	Define Software Defect Prevention.	BTL1	Remembering
2.	List key process changes for Defect Prevention.	BTL1	Remembering
3.	Mention Defect Prevention Considerations.	BTL1	Remembering
4.	What is management's role in Defect Prevention?	BTL1	Remembering
5.	Define framework for Software Process Change.	BTL1	Remembering
6.	Managing resistance to Software Process Change How?	BTL1	Remembering
7.	Distinguish between Defect Prevention and Defect Detection.	BTL2	Understanding
8.	Interpret the management's role in Software Defect Prevention.	BTL2	Understanding
9.	Differentiate between process changes for Defect Prevention and defect resolution.	BTL2	Understanding
10.	Discuss considerations for effective Defect Prevention in software development.	BTL2	Understanding
11.	Identify two principles of Software Defect Prevention.	BTL3	Applying
12.	Illustrate a process change for Defect Prevention in software development.	BTL3	Applying
13.	Examine the role of management in Software Defect Prevention.	BTL3	Applying
14.	Compare Software Defect Prevention principles and process changes, highlighting their synergies in enhancing overall Software Quality.	BTL4	Analyzing

15.	Analyze the role of management in Defect Prevention, emphasizing its pivotal influence on creating a proactive quality culture within the development team.	BTL4	Analyzing
16.	Examine considerations in Defect Prevention, exploring their impact on managing resistance to Software Process Change.	BTL4	Analyzing
17.	Distinguish between a framework for Software Process Change and a framework for Defect Prevention.	BTL5	Evaluating
18.	Why are the principles of Software Defect Prevention crucial in the development lifecycle?	BTL5	Evaluating
19.	Develop two key principles of Software Defect Prevention, emphasizing their impact on enhancing overall Software Quality.	BTL6	Creating
20.	Design a framework for Software Process Change by highlighting two specific actions that management can undertake to effectively manage resistance within the development team.	BTL6	Creating
21.	Interpret strategies for managing resistance to Software Process Change.	BTL 2	Understanding
22.	Explain the framework for Software Process Change, illustrating its practical application through concise case studies.	BTL4	Analyzing
23.	Compare two case studies highlighting successful Software Process Changes for Defect Prevention.	BTL 3	Applying
24.	Justify the necessity for process changes in Software Defect Prevention.	BTL5	Evaluating
	PART B		
1.	List three principles of Software Defect Prevention and briefly explain each. (13)	BTL1	Remembering
2.	Mention key process changes that can effectively contribute to Software Defect Prevention. (13)	BTL1	Remembering
3.	What considerations should be taken into account for effective Defect Prevention in software development? (13)	BTL1	Remembering
4.	(i) Define the Principles of Software Defect Prevention. (7) (ii) List key process changes for Defect Prevention in software development. (6)	BTL1	Remembering
5.	Distinguish between the roles of individual contributors and management in Software Defect Prevention. (13)	BTL2	Understanding
6.	Interpret the framework for Software Process Change and its significance in improving Software Quality. (13)	BTL2	Understanding
7.	Differentiate between managing resistance to Software Process Change and forcing compliance.(13)	BTL2	Understanding
8.	Illustrate the role of management in fostering a culture of continuous improvement for Defect Prevention. (13)	BTL3	Applying
9.	Examine two case studies highlighting successful implementations of Defect Prevention strategies in software development. (13)	BTL3	Applying

10.	Analyze the core principles guiding Software Defect Prevention, emphasizing strategies for integration into the software development life cycle. (13)	BTL4	Analyzing
11.	Examine the role of process changes in Defect Prevention, focusing on how adjustments in methodologies and workflows contribute to higher Software Quality. (13)	BTL4	Analyzing
12.	Explain the key considerations crucial for successful Software Defect Prevention, addressing factors like proactive risk management and continuous process improvement. (13)	BTL4	Analyzing
13.	Justify the necessity of a structured framework for Software Process Change, outlining its components and illustrating how it facilitates sustained process improvements. (13)	BTL5	Evaluating
14.	Develop a concise strategy to manage resistance to Software Process Change , considering communication and incentive mechanisms for smoother implementation within the team. (13)	BTL6	Creating
15.	(i) Distinguish between Defect Prevention considerations and Defect Detection strategies in software development. (7) (ii) Interpret the role of management in Software Defect Prevention.(6)	BTL2	Understanding
16.	(i) Identify the components of a framework for Software Process Change in the context of Defect Prevention. (7) (ii) Illustrate strategies for managing resistance to Software Process Change with relevant examples. (6)	BTL3	Applying
17.	Analyze a specific instance of Defect Prevention implementation, outlining the measures taken and the resultant impact on Software Quality and project outcomes. (13)	BTL5	Evaluating
	PART C		
1.	Why it is imperative to consider Defect Prevention at the design stage of the Software Development Process, and how does this early consideration influence the overall success of Defect Prevention strategies? (15)	BTL5	Evaluating
2.	Design a Defect Prevention oriented Software Development Process, highlighting key checkpoints and measures at each stage to ensure a proactive approach to identifying and mitigating defects. (15)	BTL6	Creating
3.	(i) Design a comprehensive set of considerations for effective Defect Prevention, encompassing elements like code reviews, testing methodologies, and documentation practices. (7) (ii) Develop strategies for managing resistance to Software Process Changes in the context of Defect Prevention, and discuss how these strategies can ensure successful adoption by the development team. (8)	BTL6	Creating
4.	Justify the need for a structured framework in managing resistance to Software Process Change, outlining key elements that can effectively mitigate resistance and ensure successful adoption of new processes. (15)	BTL5	Evaluating

5.	(i) Why is Defect Prevention essential in software development, and how does it contribute to project success? (7) (ii) Justify the need for specific process changes in Defect Prevention and explain how these changes align with continuous improvement principles. (8)	BTL5	Evaluating