



MARRI LAXMAN REDDY INSTITUTE OF TECHNOLOGY AND MANAGEMENT

(AN AUTONOMOUS INSTITUTION)

(Approved by AICTE, New Delhi & Affiliated to JNTUH, Hyderabad)

Accredited by NAAC with 'A' Grade & Recognized Under Section 2(f) & 12(B) of the UGC act, 1956

COURSE CONTENT

SOFTWARE QUALITY ENGINEERING								
I Semester: CSE								
Course Code	Category	Hours / Week			Credits	Maximum Marks		
2415817	Foundation	L	T	P	C	CIA	SEE	Total
		3	0	0	3	40	60	100
Contact Classes: 45	Tutorial Classes: Nil	Practical Classes: Nil			Total Classes: 45			
Prerequisites: A course on software quality engineering								

Course Overview:

Software Quality Engineering focuses on principles, techniques, and standards used to ensure the development of reliable and high-quality software systems. The course covers software quality models, verification and validation, testing strategies, reviews, audits, and quality assurance processes. It emphasizes defect prevention, process improvement models like ISO and CMMI, and automated testing tools.

Course Objectives:

1. To understand the fundamental concepts, perspectives, and importance of software quality in software development.
2. To study quality assurance techniques including defect prevention, defect reduction, and defect containment methods.
3. To learn software quality engineering activities, quality planning, assessment, and process improvement techniques.
4. To develop knowledge of software testing methods, test management, automation, and quality measurement practices.
5. To analyze coverage-based and usage-based testing techniques for improving software reliability and performance.

Course Outcomes: After Completion of the Course, Students should be able to

1. Analyze different perspectives and frameworks of software quality (including ISO-9126) and their role in identifying, measuring, and improving correctness and defect properties in software systems.
2. Evaluate techniques for defect prevention, reduction, and containment within software quality assurance to improve fault tolerance, safety, and risk management.
3. Formulate quality planning strategies, assess software processes, and propose improvements through quality engineering practices.
4. Develop and manage effective test strategies by planning, executing, measuring, and automating test activities to ensure software quality and reliability.
5. Apply coverage and usage-based testing techniques, including operational profile construction, to evaluate and improve the reliability of software systems through case studies.

UNIT - I:

Software Quality

Quality: perspectives and expectations, Quality frameworks and ISO-9126, correctness and defects: Definitions, properties and Measurements, A historical perspective of quality, software quality.

UNIT - II:

Quality Assurance

Classification: QA as dealing with defects, Defect prevention- Education and training, Formal method, other defect prevention techniques, Defect Reduction - Inspection: Direct fault detection and removal, Testing: Failure observation and fault removal, other techniques and risk identification, Defect Containment- software fault tolerance, safety assurance and failure containment

UNIT - III:

Quality Engineering: Activities and process, Quality planning: Goal setting and Strategy formation, Quality assessment and Improvement, Quality engineering in software process.

UNIT - IV:

Test Activities, Management and Automation

Test planning and preparation, Test execution, Result checking and measurement, Analysis and follow- up, Activities People and Management, Test Automation

UNIT - V:

Coverage and usage testing based on checklist and partitions

Checklist based testing and its limitations, testing for partition Coverage, Usage based Statistical testing with Musa's operational profiles, Constructing operational profiles

Case Study: OP for the cartridge Support Software

TEXT BOOKS:

1. Jeff Tian, Software Quality Engineering, Testing, Quality Assurance, and Quantifiable improvement.
2. Richard N. Taylor, Software Architecture: Foundations, Theory, and Practice

REFERENCE BOOKS:

1. Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement *by* Jeff Tian.
2. Foundations of Software Testing: ISTQB Certification *by* Rex Black, Erik van Veenendaal, and Dorothy Graham.
3. Software Quality Assurance: From Theory to Implementation *by* Daniel Galin
4. Managing Software Quality: An Essential Guide for Software Practitioners *by* Lloyd Roden.

ELECTRONIC RESOURCES:

1. <https://www.guru99.com/software-quality-assurance.html>
2. <https://www.softwaretestinghelp.com/software-quality-assurance-sqa-tutorial>
3. https://www.tutorialspoint.com/software_quality_assurance/index.htm
4. <https://www.edx.org/learn/software-quality-assurance>

MATERIALS ONLINE:

1. Course template
2. Tutorial question bank
3. Tech talk and Concept Video topics
4. Open-ended experiments
5. Definitions and terminology
6. Assignments
7. Model question paper –
8. Model question paper – II
9. Lecture notes
10. E-Learning Readiness Videos (ELRV)

