Module 9: Requirements Quality

1. Quality Measures (IEEE-830)

   o Correctness: All requirements must be met by the software; no extra "nice-to-haves"(Module 9 - Requirements...).

   o Completeness: Describes all necessary functionality, performance, constraints, and interfaces(Module 9 - Requirements...).

   o Unambiguous: Each requirement should only have one interpretation (Module 9 - Requirements...).

   o Consistency: No conflicts among requirements(Module 9 - Requirements...).

   o Ranked: Prioritize by importance and stability(Module 9 - Requirements...).

   o Verifiability: Requirements must be testable(Module 9 - Requirements...).

   o Modifiability: The structure must allow easy changes(Module 9 - Requirements...).

   o Traceability: Clear origins and referable in future development(Module 9 - Requirements...).

2. Key Challenges and Solutions

   o Correctness: Requires peer and customer reviews, traceability(Module 9 - Requirements...).

   o Completeness: Joint reviews with users, prototyping to ensure all functional and non-functional requirements are captured(Module 9 - Requirements...).

   o Unambiguity: Use multi-level reviews, prototypes, and measurable criteria (Module 9 - Requirements...).

   o Consistency: Extensive manual reviews to prevent contradictions(Module 9 - Requirements...).

   o Ranked Requirements: Prioritize based on scope, stability, and importance (Module 9 - Requirements...).

   o Verifiability: Testing-aware wording and process-focused validation(Module 9 - Requirements...).

- o Modifiability: Embrace change through tools and a change management process(Module 9 - Requirements...).

- o Traceability: Use unique identifiers for all requirements and artifacts(Module 9 - Requirements...).

3. Requirements Validation

- o Validation: Ensures the system meets the user's needs. It's more than just testing; it checks for completeness, ambiguity, and omissions(Module 9 - Requirements...).

- o Prototyping: Helps with validating concepts but must be careful about shortcuts(Module 9 - Requirements...).

- o Requirements-Based Testing: Validates use cases and ensures all scenarios are covered(Module 9 - Requirements...).

Module 10: Requirements Standards

1. General Standards

- o IEEE 29148: Focused on requirements process, documents (SRS, StRS, SyRS), and traceability. An extended and detailed version of IEEE 830(Module 10 - Requirement...).

- o CMMI (Capability Maturity Model Integration): Emphasizes process maturity. Includes Requirements Development (RD) and Requirements Management (REQM), focusing on elicitation, validation, and traceability (Module 10 - Requirement...).

- o SWEBOK: Software Engineering Body of Knowledge, covering all essential knowledge for software engineers, including requirements(Module 10 - Requirement...).

- o SEBOK: Systems Engineering Body of Knowledge, guiding systems design and requirement decomposition(Module 10 - Requirement...).

- o BABOK: Business Analyst Body of Knowledge, focusing on people, collaboration, and communication in requirements processes(Module 10 - Requirement...).

2. Industry-Specific Standards

- o MIL-STD-498: Defense standard focused on comprehensive documentation like SRS, design descriptions, and interface descriptions(Module 10 - Requirement...).

- o DO-178C: Avionics software standard with design assurance levels based on failure consequences. It requires extensive traceability and verification (Module 10 - Requirement...).

- o IEC 62304: Medical device software standard focused on risk management, development, and requirements traceability(Module 10 - Requirement...).

- o EN 50128: Railway system standard focused on safety integrity levels and full traceability(Module 10 - Requirement...).

3. Why Standards Matter

- o Standards provide a common approach to requirements engineering, ensuring rigor, traceability, and compliance, particularly in industries with critical systems like defense, aerospace, medical devices, and transportation(Module 10 - Requirement...).