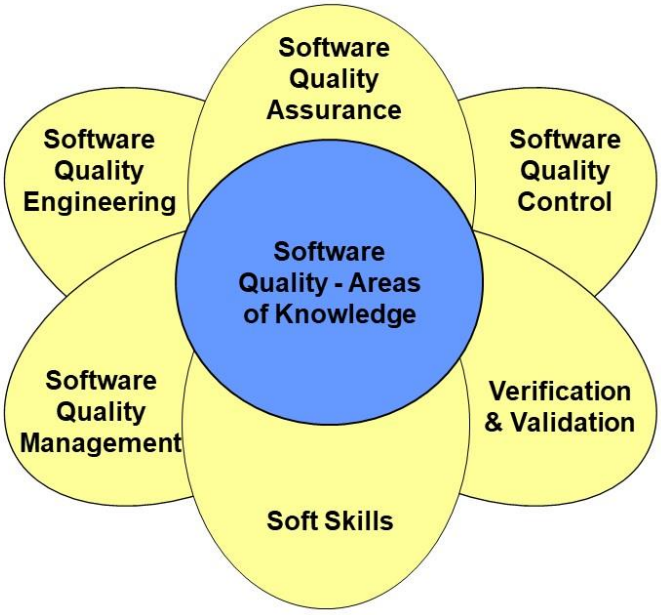
6 Quality Knowledge Areas

By Linda Westfall

https://www.softwareexcellenceacademy.com/



Software quality has emerged as a multifaceted discipline that requires us, as software quality engineers, to be able to understand and apply the knowledge that encompasses:

- Software quality management. The processes and activities involved in setting the organization's strategic quality goals and objectives, establishing the organizational, project, and product quality planning, and providing the oversight necessary to ensure the effectiveness and efficiency of the organization's quality management system. Software quality management provides leadership and establishes an integrated, cross-functional culture where producing high-quality software is "just the way we do things around here."
- Software quality engineering. The processes and activities needed to define, plan and implement the quality management system for software-related processes, projects, and products. This includes defining, establishing, and continuously improving software-related systems, policies, processes, and work instructions that help prevent defects and build quality into the software.
- Software quality assurance. The planned and systematic set of all actions and activities needed to provide adequate confidence that the:

- Software work products conform to their standards of workmanship and that quality is being built into the products
- The organization's quality management system (or each individual process) is adequate to meet the organization's quality goals and objectives, is appropriately planned, documented, and improved, is being followed, and is effective and efficient.
- Software quality control. The planned and systematic set of all actions and activities needed to monitor and measure software projects, processes, and products to ensure that special causes have not introduced unwanted variation into those projects, processes, and products.
- Software verification and validation. The processes and activities used to ensure that software products meet their specified requirements and intended use. Verification and validation help ensure that the "software was built right" and the "right software was built."
- Soft skills. A software quality engineer also needs what are referred to as the "soft skills" to be effective in influencing others toward quality. Examples of "soft skills" include leadership, team building, facilitation, communication, motivation, conflict resolution, negotiation, and more.

References:

Westfall 2017 - Westfall, Linda, *The Certified Software Quality Engineer Handbook*, Second Edition. Milwaukee: ASQ Quality Press. 2017.