Code Complete

Steve McConnell

20. The Software-Quality Landscape

20.1 Characteristics of Software Quality

Software has both external and internal quality characteristics. External characteristics are characteristics that a user of the software product is aware of, including

● Correctness. The degree to which a system is free from faults in its specification, design, and implementation.

● Usability. The ease with which users can learn and use a system.

● Efficiency. Minimal use of system resources, including memory and execution time.

● Reliability. The ability of a system to perform its required functions under stated conditions whenever required—having a long mean time between failures.

● Integrity. The degree to which a system prevents unauthorized or improper access to its programs and its data. The idea of integrity includes restricting unauthorized user accesses as well as ensuring that data is accessed properly—that is, that tables with parallel data are modified in parallel, that date fields contain only valid dates, and so on.

● Adaptability. The extent to which a system can be used, without modification, in applications or environments other than those for which it was specifically designed.

20.1 Characteristics of Software Quality

● Accuracy. The degree to which a system, as built, is free from error, especially with respect to quantitative outputs. Accuracy differs from correctness; it is a determination of how well a system does the job it's built for rather than whether it was built correctly.
● Robustness. The degree to which a system continues to function in the presence of invalid inputs or stressful environmental conditions.

External characteristics of quality are the only kind of software characteristics that users care about. Users care about whether the software is easy to use, not about whether it's easy for you to modify. They care about whether the software works correctly, not about whether the code is readable or well structured.