**Discuss the assessment of software quality according to the quality attributes shown on slide number eleven. Choose five of these attributes and discuss how they might be assessed.**

Quality control is aimed at ensuring that software products meet some requisite level of quality. According to Sommerville (2011), since the developers may be biased by software development issues, often an independent quality assurance team is needed to assess software systems. He also lists numerous software system quality attributes that can be assessed, including:

|  |  |  |
| --- | --- | --- |
| * safety * security * reliability * resilience * robustness | * understandability * testability * adaptability * modularity * complexity | * portability * usability * reusability * efficiency * learnability |

(Sommerville, 2011).

It is beyond the scope of this discussion to review each of the quality attributes listed above. However, five attributes have been selected for discussion of how they can be assessed:

1. **Learnability** - can be assessed by determining the extent to which new users of the software can get some work done (Barbacci, 2004). For example, what percentage of new users are able to complete some predetermined task within some predetermined time frame? If, for example, a kiosk will only be used once, there is an expectation that the user should immediately be able to use the kiosk. On the other hand, word processing applications, like Microsoft Word, may have specific functionality that only expert users need learn about, like using formatting stylesheets or mail merge operations.
2. **Efficiency** - This answers the question, “How efficient is the system to use?” This can be assessed by measuring the productivity of expert users (Barbacci, 2004). The more productivity, the more efficient the system.
3. **Security** - is the extent to which the system is secure from intrusion, or at least able to maintain some level of service in the presence of attacks. How are various system attacks handled? Does the system have mechanisms for preventing a system crash when attacked? (Barbacci, 2004).
4. **Usability** - is a measure of how easily a user can operate and take advantage of the functionality of a system (Barbacci, 2004). This could be assessed by noting the error rate by users. How many errors do users make and can they recover from those errors?
5. **Modularity** - is the extent to which a system is partitioned into distinct modules that have distinct functionality (Barbacci, 2004). Can specific functionality be added or removed from the system without disrupting other system functions? To what extent are modules inter-dependent?

According to Barbacci (2004) quality attributes may or may not be relevant to a particular software system. Moreover, attributes may be in conflict and therefore require prioritizing. Having some sort of metric for assessing each ranked attribute can give us a sense of the quality of the software being assessed (Barbacci, 2004).

**References**

Barbacci, M. R. (2004). “Software Quality Attributes: Modifiability and Usability.” Carnegie Mellon University. Retrieved from <http://www.ieee.org.ar/downloads/Barbacci-05-notas1.pdf>

Sommerville, I. (2011). Software Engineering. Edition 9.