IJITCE

ISSN 2347- 3657

International Journal of Information Technology & Computer Engineering

Email : ijitce.editor@gmail.com or editor@ijitce.com

International Journal of Information Technology & Computer Engineering

ISNNO: ISSN 2347–3657

Volume 4,Issue 4 ,2017

ESSENTIAL FEATURES OF GOOD SOFTWARE

E Ravi
Dr J Rajaram
J Rajaram,

ABSTRACT

Software engineering attempts to produce systems that are "good systems" in terms of reliability, ease of maintenance etc. We take a broader definition of a good system as any general system that produces benefits that exceedinitial expectations or intended scope or initial investment. There appear to be common characteristics that tie together such systems. These are hypothesized to include functional "goodness", good infrastructure, reliability, connect-ability, versatility and benefits that overflow/overwhelm the system's scope or initial investment. A case study approach involving fourexamples of what are regarded as "good systems" and four examples of what are regarded as "bad systems" fully supports this hypothesis. But support for the converse hypothesis, a bad system not having these characteristics was only 68.7%. The implications of these findings are discussed.

Keywords: **good systems, system characteristics, strategic systems, systems theory, software engineering philosophy, information systems philosophy, complex systems analysis.**

While developing any kind of software product, the first question in any developer's mind is, "What are the qualities that good software should have?" Well before going into technical characteristics, I would like to state the obvious expectations one has from any software.

First and foremost, a software product must meet all the requirements of the customer or end-user. Also, the cost of developing and maintaining the software should be low. The development of software should be completed in the specifiedtime-frame.
Well these were the obvious things which are expected from any project (and software development is a project in itself). Now lets take a look at Software Quality factors. These set of factors can be easily explained by Software QualityTriangle.

The three characteristics of good application software are :-

1) Operational Characteristics

2) Transition Characteristics

3) Revision Characteristics

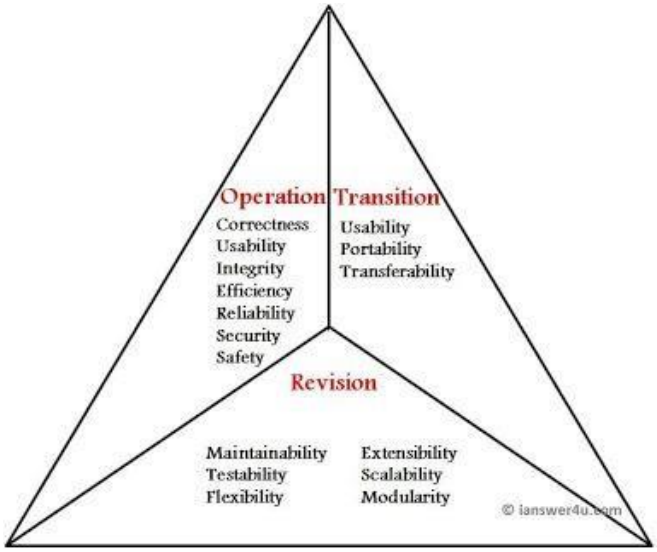
Assistant Professor M.Tech—CSE
Professor Ph.D ----CSE
Associate Professor
Nagole institute of engineering and technology

International Journal of Information Technology & Computer Engineering

ISNNO: ISSN 2347–3657

Volume 4,Issue 4 ,2017

Software Quality Triangle



Software Quality Triangle with characteristics

16 Characteristics of a Good Software

What Operational Characteristics should a software have ?

These are functionality based factors and related to 'exterior quality' of software. Various Operational Characteristics of software are :

a) **Correctness:** The software which we are making should meet all the specifications stated by the customer.

b) **Usability/Learnability:** The amount of efforts or time required to learn how touse the software should be less. This makes the software user-friendly even for IT-illiterate people.

c) **Integrity :** Just like medicines have side-effects, in the same way a software may have a side-effect i.e. it may affect the working of another application. But

ISNNO: ISSN 2347–3657

Volume 4,Issue 4 ,2017

quality software should not have side effects.

d) **Reliability :** The software product should not have any defects. Not only this, it shouldn't fail while execution.

e) **Efficiency :** This characteristic relates to the way software uses the available resources. The software should make effective use of the storage space and executecommand as per desired timing requirements.

f) **Security :** With the increase in security threats nowadays, this factor is gaining importance. The software shouldn't have ill effects on data / hardware. Proper measures should be taken to keep data secure from external threats.

g) **Safety :** The software should not be hazardous to the environment/life.

What are the Revision Characteristics of software ?

These engineering based factors of the relate to 'interior quality' of the software like efficiency, documentation and structure. These factors should be in-build inany good software. Various Revision Characteristics of software are :-

a) **Maintainability :** Maintenance of the software should be easy for any kind ofuser.

b) **Flexibility :** Changes in the software should be easy to make.

c) **Extensibility :** It should be easy to increase the functions performed by it.

d) **Scalability :** It should be very easy to upgrade it for more work(or for morenumber of users).

e) **Testability :** Testing the software should be easy.

f) **Modularity :** Any software is said to made of units and modules which are independent of each other. These modules are then integrated to make the finalsoftware. If the software is divided into separate independent parts that can be modified, tested separately, it has high modularity.

Transition Characteristics of the software :

a) **Interoperability :** Interoperability is the ability of software to exchange information with other applications and make use of information transparently.

b) **Reusability :** If we are able to use the software code with some modifications for different purpose then we call software to be reusable.

c)**Portability :** The ability of software to perform same functions across all environments and platforms, demonstrate its portability.

Importance of any of these factors varies from application to application. In systems where human life is at stake, integrity and reliability factors must be givenprime importance. In any business related application usability and maintainability are key factors to be considered. Always remember in Software Engineering, quality of software is everything, therefore try to deliver a product which has all these characteristics and qualities.

CONCLUSIONS AND IMPLICATIONS

Based on the cases presented, there is strong support for the hypothesis that good systems have the characteristics as identified in this research, viz. fulfills its primary function, good infrastructure, is easily connectable, is adaptable/ versatile, reliable and finally produces benefits that exceed the system's scope. A primary limitation of the study is its qualitative nature. Evaluation of whether or not the criteria was fulfilled was subjective. There was also no attemptto control for the size or complexity of the system although all systems presented have their own levels of complexity. There is no assurance that the criteria are complete. Some criteria such as modularity, configurability, flexibility, evolvability, architecture, feedback and self organization have not been considered. These, it was felt would not

ISNNO: ISSN 2347–3657

Volume 4,Issue 4 ,2017

be applicable to all types of systems. For example, an airplane is a self- contained system and modularity would not be relevant to whether or not it is a good system fromthe point of view of transporting passengers. Similarly the notion of an architecture is not appropriate to systems such as the healthcare system. There is obviously opportunity for further development.

REFERENCES

1. Amaravadi, C. S., (2004). "The Laws of Information Systems," Journal of Management Research, 4(3), pp 129-137, December.
2. Anonymous (2010). "Walmart makes fashion statement with item-level RFID". Material Handling Management July 26th (no volume, issue or page information).
nderson, G. F., and Squires, D. A. (2010). Measuring the US health care system: a cross-national comparison. Issue Brief (Commonwealth Fund), 90, 1-10.
3. App Store (2015). Apple Inc., https://itunes. apple.com/us/genre/ios/id36?mt=8.
4. Barroso, L. A., Dean, J., and Hölzle, U. (2003). Web search for a planet: The google cluster architecture. IEEE Micro, 23(2), 22-28.
5. Boeing (2014). Statistical Summary ofCommercial Jet Airplane Accidents Worldwide Operations|1959–2013 (August, 2014). Retrieved from http://www.boeing. com/news/techissues/pdf/statsum.pdf current July 2015.
6. Bertalanffy, L. V. (1950). An Outline of General Systems Theory," The British Journal for the Philosophy of Science, 1:2 August, pp. 134-165.
7. Brin, S., and Page, L. (2012). Reprint of: The anatomy of a large-scale hypertextual web search engine. Computer Networks, 56(18), 3825-3833.
8. Brodwin. E., 2014. 11 Charts that show exactly what's wrong with the US Healthcare System. Business Insider, September 23. http://finance.yahoo.com/news/11-charts-show-exactly-whats-133000491.html.
9. Chatterley, T., & Chojecki, D. (2010). Personal digital assistant usage among undergraduate medical students: exploring trends, barriers, and the advent of

ISNNO: ISSN 2347–3657

Volume 4,Issue 4 ,2017

smartphones. Journal of the Medical Library Association : JMLA, 98(2), 157-160.

10. Clemons, E. K., & Row, M. (1988). McKesson Drug Company: a case study of Economost— a strategic information system. Journal of Management Information Systems, 5(1), 36-50.