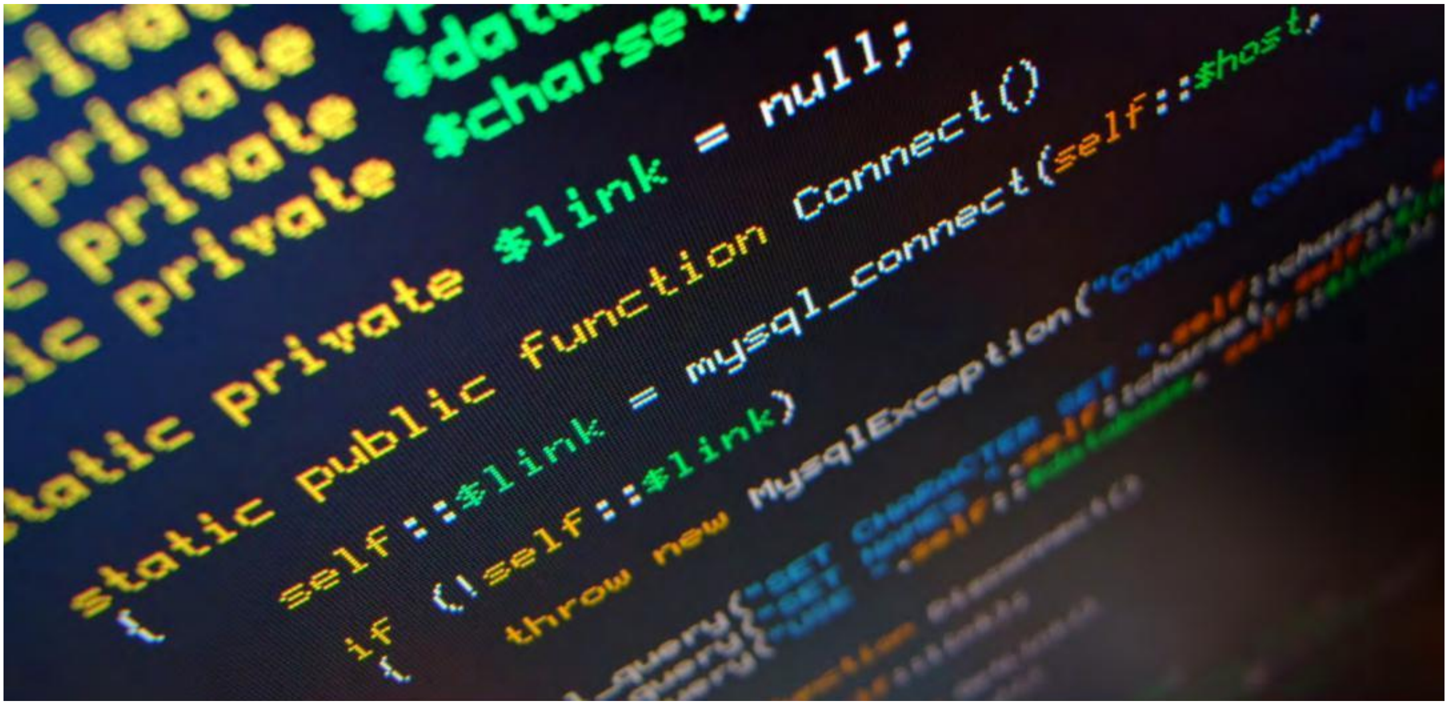
SOFTWARE QUALITY– IT IS A POWERFUL DIFFERENTIATOR IN THE MARKETPLACE



WHITEPAPER

ECONOMICS OF QUALITY

The cost of software quality has been ever increasing and it is directly related to the stage of the development lifecycle where it is detected. The cost of removing a defect grows prohibitively for each downstream phase of the development lifecycle in which it remains undiscovered.

Typically, a bug that costs $1 to fix on the programmers desktop costs $100 to fix, once it is incorporated into a complete program and many thousands of dollars, if it is identified after the software has been deployed in the field. National Institute of Standards and Technology (NIST) research study showed that 80% of software development dollars are spent correcting software defects. The same NIST study estimated that software defects cost the US economy alone, $60 billion every year. The Standish Group report suggested that cancelled projects cost $55 billion of lost revenues due to quality.

SOFTWARE QA VS SOFTWARE TESTING

- No specific tool that is a direct fit for the customer needs
- They had daily builds for the app
- Insufficient regression coverage
- Predictability and confidence of the releases
- Cost of rework was very high

OUR ROLE

Software testing cannot be totally eliminated, for it has to remove the bugs that have already entered the software under test. Testing is one activity that can remove software bugs and can provide the confidence to deploy in customers production environment. Having said that, Software QA ensures removal of bugs early in the development lifecycle and it is a part of the development process itself. After all, it reduces the number of bugs to be detected and re-moved from the software during the testing phase.

An improved software QA process has the potential to affect software developers and users by:

- Removing more bugs before the software is released
- Detecting bugs earlier in the software development lifecycle
- Locating the source of bugs faster and with more precision

Essentially, the number of bugs entered into a software code is dependent on the skill and techniques employed by the programmer/developer during different phases of the development lifecycle.

In the larger sense, Software Quality is a part of each phase of the development lifecycle including testing and encompasses a whole load of factors and not just bug-free and aesthetic software. These factors that are a combination of tangible and intangible sum of various factors like functionality, usability, reliability, performance, scalability etc. and this will be the focus of this paper.

DEFINING SOFTWARE QUALITY

The easiest definition would be that Software Quality is a defined process for creating useful software for the target audience, which adds value for both the user and the producer. Adding to that definition, a bug free, aesthetically appealing product that fails to solve business problems for which it was designed is a poor quality product. Software Quality is a sum of all these quality attributes defined by ISO:

- Functionality – suitability, accurateness, interoperability, compliance and security
- Reliability – maturity, fault tolerance and recoverability

- Usability – understandability, learnability and operability
- Efficiency – time behavior and resource behavior
- Maintainability – analyzability, changeability, stability and testability
- Portability – adaptability, installability, conformance and replaceability

This definition of quality ensures that it is not just about eliminating bugs but it is about meeting user needs; it is about usability, security, scalability, reliability, maintainability, performance and portability. It is the total result of what the team produces.

Now, it perhaps makes sense to see quality as a differentiator and this applies to both commercial software as well as internal applications. Throughout the development lifecycle, there needs to be constant attention to quality.

QUALIT Y IS FREE;
EVERY THING ELSE COSTS A LOT OF $$$

Typical software development nowadays demands faster development at lower costs with added functionality. In this framework, development managers give quality a miss, thinking that it will allow them to do development faster and at lower costs, and anything and everything will get captured during the testing phase. This really is a misconception and focus on quality will actually expedite development, reduce costs and allows new features to be added with greater ease.

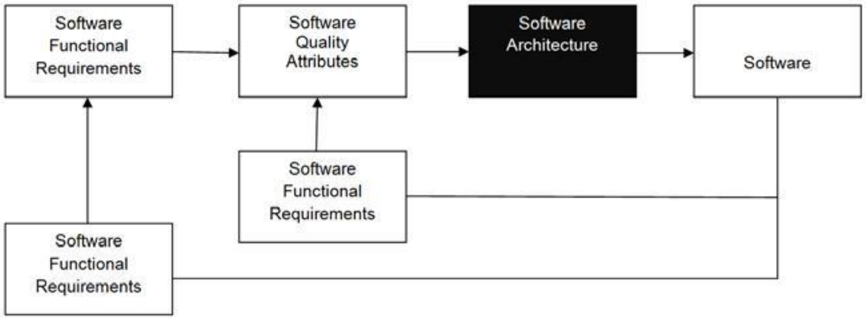
High-quality software will have reduced reworking costs and will allow the organization to in-novate and pursue newer opportunities. This certainly can be a real differentiator in the market place because even today, high-quality software is an exception rather than a rule.

HOW DO WE GET IT RIGHT?

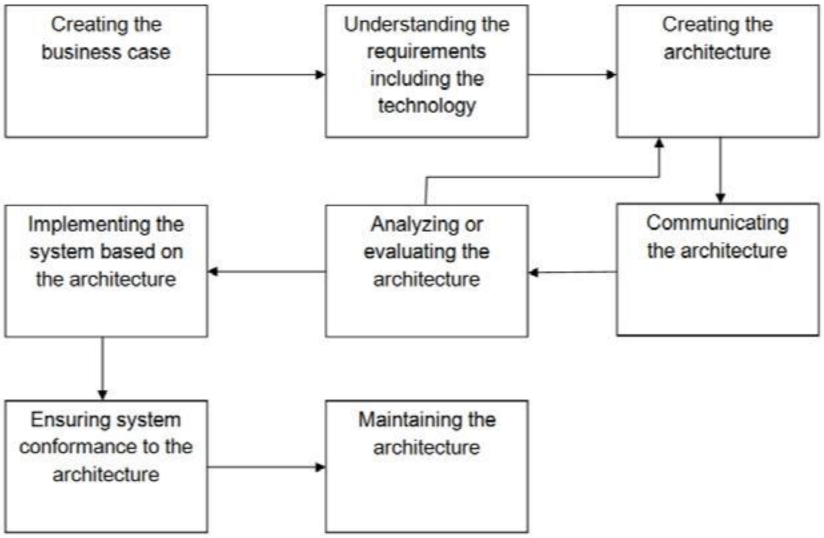
Having said that the software has to have all the quality attributes defined by ISO standards, in addition to being bug-free and aesthetic, the one most important thing that can influence the quality of the software is the architecture used. Software architecture plays a key role, and it is a collection of design decisions intended to ensure functionality and the other quality attributes of the software.

Tools, technology, frameworks and processes used for development have to work together for the software to produce the desired results. Architecting it right ensures that these work together satisfying the business goals of high quality, quick time- to-market, effective use of limited re-sources, low cost production, low cost maintenance and mass customization.

SOFTWARE ARCHITECTURE - PART OF THE SOFTWARE DEVELOPMENT PROCESS


Software Functional Requirements
Software Quality Attributes
Software Architecture
Software
Software Functional Requirements
Software Functional Requirements

ARCHITECTURE DRIVEN SOFTWARE DEVELOPMENT


Creating the business case
Understanding the requirements including the technology
Creating the architecture
Implementing the system based on the architecture
Analyzing or evaluating the architecture
Communicating the architecture
Ensuring system conformance to the architecture
Maintaining the architecture

Software architecture represents earliest design decisions that are hardest to change and is a communication vehicle among stakeholders. It also addresses the quality attributes as desired by the target users of the system.

In a rush to shorten the learning curve and to get the development completed, very often basic development guidelines and quality attributes are ignored. Several transversal functions are un-stated and are not met; basic principles of object – orientation and modularity are ignored. This is discovered too late in the cycle and this result in loss of quality of the software.

Architecting the software right allows you to create a template for significant parts of the code, say the primary types of components, thus ensuring that the common concerns such as modularity, error handling, logging and other quality attributes are met while ensuring that the developing team has a shorter learning cycle. This will allow them to get started quickly ensuring the business goals of quick time to market at lower costs are met.

Architecture driven development will ensure that software developed will be useful for the target audience accounting for the quality needs while adding value to both the user and the producer.

CONCLUSION

Whenever we mention that software quality has to be improved, the knee jerk reaction is to add resources to the testing team. It can certainly ensure that the number of bugs in the system is reduced but cannot ensure the needs of the business.

Software quality is a comprehensive view of the software that takes into account the domain needs, stakeholder needs and various quality attributes that are critical for the user, including functional and performance testing.

ABOUT ZADO

Zado is an independent QA Solution provider with specific focus on testing web, mobile and cloud applications. Our framework-driven approach to test automation ensures reliability and performance of your applications in diverse environments and complexities.

Our Center of Excellence works towards ensuring the success of every test automation initiative of our customers, irrespective of the stage that they are in – startup, transitional or mature. We have successfully helped Startup, e-Commerce and Independent Software Vendors with their automation needs. Our goal is to ensure quality of your software using test automation optimally.

We are open to doing POCs and Pilots that prove our credibility. We also have an innovative engagement model, Enhance – Optimize – Transfer (EOT), where we implement automation testing and transition it to your local teams. Our points of intervention after that, will be only towards enhancing the automation framework.

Zado automation frameworks help manual testers write their own test scripts without the necessary automation expertise. This qualifies manual testers into automation testers, providing better economies of scale and faster ROI of your automation efforts.

Infotech Solution

Zado Infotech Solutions India Pvt Ltd
85/3 1st Floor Orchid Plaza, Razaak Garden Road,
Arumbakkam, Chennai - 106.

contact@zadoinfotech.com
www.zado-tech.com