

1. What are the four important attributes that all professional software should have? Suggest four other attributes that may sometimes be significant.

Answer

Maintainability, dependability and security, efficiency, acceptability

Correctness, reusability, portability, usability, interoperability

2. Develop your own answers to the five questions asked on slide 10, software development challenges. Discuss them with your fellow classmates.

Answer

- Why does it take so long to get software finished?
  - Most software has a handful of user requirements, each requirement has to be carefully study and understood. For some software engineering methods, documentation may take up a lot of effort. Requirement change prolong the development schedule. A single change may affect many other areas of the software.
- Why are development costs so high?
  - The development of a software required a lot of skillful and experience people. They are highly educated and expensive work force. The development of software required not only to study the user requirements carefully, it also need to design the solution and implement the design in term of software. After the implementation, the produced software has to be tested. As you can see, the development of a software is a lengthy process which, spend a lot of expensive manpower.
- Why can't we find all errors before we give the software to our customers?
  - Software evaluation technique cannot spot all potential errors. Some errors occur only with the input of some specific data or specific situation where it might happen only in the production environment.
  - Errors in software also occur due to the change of production environment which is not possible to be simulated in the development environment. (For example, change of hardware,
- Why do we spend so much time and effort maintaining existing programs?
  - If the existing system were in use for some time, it will be difficult (risky and costly) to replace with another system. In the case where are there bugs or new requirement, it is more feasible to maintain and upgrade the existing system.
- Why do we continue to have difficulty in measuring progress as software is being developed and maintained?
  - Unlike civil engineering, software engineer builds software which is intangible. It is hard to measure the progress of the virtual product which is still in

progress. Unlike building a skyscraper, when you say you have built 20 floors, others can count and verify that you have 20 floors. In software development, sometime the halfway done software cannot be verify because unfinished software cannot be executed. Even if you have a halfway done software, it may consist of serious bug which requires unknown amount of time to fix.

3. What is the most important difference between generic software product development and custom software development? What might this mean in practice for users of generic software products?

ANSWER

The essential difference is that in generic software product development, the specification is owned by the product developer. For custom product development, the specification is owned and controlled by the customer. The implications of this are significant – the developer can quickly decide to change the specification in response to some external change (e.g. a competing product) but, when the customer owns the specification, changes have to be negotiated between the customer and the developer and may have contractual implications.

For users of generic products, this means they have no control over the software specification so cannot control the evolution of the product. The developer may decide to include/exclude features and change the user interface. This could have implications for the user's business processes and add extra training costs when new versions of the system are installed. It also may limit the customer's flexibility to change their own business processes.

4. Many modern applications require modification frequently (before they are presented to the end user and then after the first version has been put into use). Suggest a few ways to build software to stop deterioration due to change.

Answer

- The software should be modular so that changes will not have a lot of side effects to other part of the software
- The software must be maintainable
- Comprehensive testing should put in place to reduce errors.
- Work closely with the stakeholder to ensure requirements are correctly defined
- Improve requirement study approach to achieve better requirements study

5. Discuss whether professional engineers should be certified in the same way as doctors or lawyers.

Answer

Unprofessional lawyer may cause some people to be suffering financially, physically, or even mentally. Unprofessional doctor may cause his patients to suffer physical damage and maybe financially as well, worst still may cause life. The same reasoning may be applied for engineers.

For example, a civil engineer may cause damage if he build a collapsing bridge or skyscraper. Software engineer may cause damage to users financially. Software engineer who writes code for high-speed train may cause hundreds if not thousands of lives if the code is not properly written and tested.

It is necessary to demand certification of engineers just in the same way doctors and lawyers are certified.

Advantages of certification for engineers:

1. Certification make the field more competitive since they have been trained according to the standard and binding law, the source of certification can ensure that the software engineer is trusted.
2. The public opinion of the engineering profession will be better when some form of certification is in place
3. Certification signals to the potential employers a minimum level of competence and this make is easy for them to screen only engineers that meet the minimum requirement.

It is important to note that certification does not guarantee competence but does guarantee that a minimum standard was reached at the time of the certification.