Abe's Questionnaires

Q1 – What's software Engineering?

From Wiki: "Software engineering (SE) is the profession of people who create and maintain software applications by applying technologies and practices from computer science, project management, engineering, application domains and other fields.

Software engineering deals with matters of cost and reliability, like traditional engineering disciplines…"

What does it mean?

The term software engineering has been commonly used with a variety of distinct meanings:

As the usual contemporary term for the broad range of activities that was formerly called programming and systems analysis;

As the broad term for all aspects of the practice of computer programming, as opposed to the theory of computer programming, which is called computer science;

Software engineering is "(1) the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software, that is, the application of engineering to software," and "(2) the study of approaches as in (1)." – IEEE Standard 610.12

Why Software Engineering?

When computer software succeeds-when it meets the needs of the people who use it,

When it performs flawlessly over a long period of time,

When it is easy to modify and even easier to use –

It can and does change things for the better.

But when software fails-

When its users are dissatisfied,

When it is error prone,

When it is difficult to change and even harder to use-bad things can and do happen.

We all want to build software that makes things better, avoiding the bad things that lurk in the shadow of failed efforts.

To succeed, we need discipline when software is designed and built.

We need an engineering approach.

What are the State requirements to become a software engineer?

There is no state requirement for ‘Software Engineer’ in Ohio.

Each organization has their own requirement aspiring to their immediate needs. Broad knowledge of software Development, System Design, Software Architecture, Software process Management are helpful.

SEI (Software Engineering Institute at Carnegie Mellon) has a certification program that can be of value to you:

Why get certified?

As an SEI-certified individual, you'll - Gain measurable knowledge and skills - Increase your marketability throughout the industry -Become knowledgeable and skilled in the latest practices –Capability to produce high-quality work -Trained and assessed through industry leading assessment methods -Have committed to a code of professional conduct that separates them from all other practitioners in the field

Q2 – Why is software ‘requirement gathering’ an important phase of software application design?

Software requirement gathering is one piece of the overall software development methodology. It can be group in the following phases:

User profiling: researching user characteristics and needs

Task Analysis: researching user tasks and workflow and their mental models of the work

Usability goal setting: Establishing quantitative, measurable goals from the requirements that are subsequently used in acceptance and usability testing.

User involvement may lead to reduced costs in several areas, including increased employee efficiency and productivity, reduced training and support, and less software development and maintenance effort by not wasting time on features users don’t need or want or won’t use.

In addition to cost savings, improved usability can create more customer traffic and increase sales. It can dramatically reduce development costs by catching problems earlier in the product cycle, where each phase can add an order of magnitude to the cost of changes.

Good requirement gathering and great user’s involvement may:

- Reduce product development risk by about 67%

- Reduce development bench cost by about 50%

- Reduce engineering development time by about 25%

- Improve product quality by 50%

Customer behavior:

The key isn’t to treat customers like idiots who must be prevented from hurting themselves, but instead to treat them like people who will grow and mature in their use of your products.

Q3 – What techniques or methodologies do you recommend? Why?

One technique that is easy to use is: Software Prototyping

Prototypes combine the most representative attributes of a category. A prototype is the stage before the final design. They are the best examples among the members of a category and serve as benchmarks against which the surrounding instances are categorized

In many fields, there is great uncertainty as to whether a new design will actually do what is desired. New designs often have unexpected problems. A prototype is built to test the function of the new design before starting production of a product.

Building the full design is often expensive and can be time-consuming. Sometimes, rather than building the full design, figuring out what the problems are, then building another full design, "rapid-prototyping" or "rapid application development" techniques are used for the initial prototypes, which implement part, but not all, of the complete design. This allows manufacturers to rapidly and inexpensively test the parts of the design that are most likely to have problems, solve those problems, and then build the full design.

Software prototypes are ideally low-cost mock-ups, which I note may be nothing more than sketches that designer can quickly modify before committing to code.

DESIGN AXIOMS

1 – Design should be modular.

2 – Design should lead to program modules (e.g. subroutines or procedures).

3 – Design should be driven by the analysis.

4 – Design should not reinvent the wheel

5 – Minimize coupling between modules and maximize cohesion within modules.

6 – Rule of thumb – a module should not exceed a page.

7 – Design should be structured to accommodate change.

8 – Design is not coding, coding is not design.

9 – Analysis is the act of understanding a problem while design is the act of developing a solution for the problem.

10 - Programming (or other technique) is the implementation of the solution.

Q4– What qualities are needed to be successful manager in the field of IT?

There are so many characteristics which a manager should possess, but in general some of these essential qualities a manager should have in order to be successful in the field of IT.

- A good subject knowledge: Whatever the portfolio a manager is handling, he should be completely expert at it. An IT manager obviously needs to know about the server, network management, database management etc.

- Ability to correlate and coordinate with other departments: He does not only need to know how to manage his department but should also know how his department can be effective in the performance of other departments.

- Ability to handle the teamwork: A successful IT manager should know what motivates a team and how to extract the best out of them. He should assess the team by gauging their weaknesses and hence should be able to remove those weaknesses by getting on top of the work.

- A keen approach towards problem-solving and decision-making: If a manager is not logical, he is of no use. If a particular team member lacks certain skills, he should train towards achieving the organizational goals. In entirety, a sound judgment is essential.

- Ability to cope with pressure: There are very few managers who can cope with the demands of the organization. A manager who can take up extra work pressure and complete the tasks at hand on time, he is a successful IT manager.

Above all, an IT manager needs to be proactive and should not be prompted by his superior. The field of IT requires to be updated as innovations take place and the IT industry witnesses change. A balanced frame of mind, being always watchful of the events and a continuous learning habit will ensure success.

Deepak Kashyap/ITvoir Network

Q5– What’s Web 2.0?

The term "Web 2.0" is commonly associated with web applications that facilitate interactive information sharing, interoperability, user-centered design and collaboration on the World Wide Web. Examples of Web 2.0 include web-based communities, hosted services, web applications, social-networking sites, video-sharing sites, wikis, blogs, mashups and folksonomies. A Web 2.0 site allows its users to interact with other users or to change website content, in contrast to non-interactive websites where users are limited to the passive viewing of information that is provided to them.

Q7 – What's a virtual social network?

A virtual community, e-community or online community is a group of people that primarily interact via communication media such as newsletters, telephone, email, internet social network service or instant messages rather than face to face, for social, professional, educational or other purposes. If the mechanism is a computer network, it is called an online community. Virtual and online communities have also become a supplemental form of communication between people who know each other primarily in real life. Many means are used in social software separately or in combination, including text-based chat rooms and forums that use voice, video text or avatars. Significant socio-technical change may have resulted from the proliferation of such Internet-based social networks.

Q8 – What's twitter?

Twitter is a free social networking and micro-blogging service that enables its users to send and read messages known as tweets. Tweets are text-based posts of up to 140 characters displayed on the author's profile page and delivered to the author's subscribers who are known as followers. Senders can restrict delivery to those in their circle of friends or, by default, allow open access. Users can send and receive tweets via the Twitter website, Short Message Service (SMS) or external applications. While the service itself costs nothing to use, accessing it through SMS may incur phone service provider fees.

Q9 – What are your experience working in the Public sector versus Private sector?

What do I mean by the public and private sector? The public sector is composed of federal, state, and local government entities. The government hires engineers and scientists for a variety of jobs, mostly in the area of R&D.

The private sector is the heartbeat of a capitalistic society. It is composed of any entity providing goods or services for a price; the objective of the private sector is to produce and market a product that returns a profit to investors.

Objective: There are fundamental differences in the objectives of the public and private sectors. The role of government in the scientific field is to expand the knowledge base and provide for the public good. This is usually referred to as high-risk research because the payback on the initial expenditures is neither guaranteed nor near-term. Government R&D programs are set-up such that the public-at-large benefits from the results (e.g., the space program, medical research on genetics and diseases such as AIDS, and defense-related technologies that have civilian uses). Private-sector R&D is focused on short-term results for developing a new product or improving an existing product to provide a competitive edge for the company or to increase profits. Private-sector research in the Information Technology industry is not normally high risk or long term.

Environment: Both sectors can provide exhilarating and satisfying jobs, depending on what you want. The government is better suited to provide for the overall long-term R&D needs of an industry, whereas, a company is better suited at meeting company-specific R&D needs.

Earning potential: President Bush earns a salary of $400,000 plus a $50,000 nontaxable expense allowance. By contrast, the average Standard & Poor's 500 company CEO took home $11.75 million in 2005, according to a Corporate Library report cited by the AFL-CIO.

Q10– Is IT a high paying career?

It’s wise to pay too much, but it’s worse to pay too little.

When you pay too much, you lose a little money – that is all. When you pay too little, you sometimes lose everything … because the thing you bought was incapable of doing the thing it was bought to do. The common law of business balance prohibits paying a little and getting a lot – it can’t be done.

If you deal with the lowest bidder, it is well to add something for the risk you run, and if you do that you will have enough to pay for something better.