Food for Thought™

An e-newsletter published by
Software Quality Consulting, Inc.

April 2008
Vol. 5 No. 4

Welcome to **Food for Thought™**, an e-newsletter from **Software Quality Consulting**. I've created free subscriptions for my valued business contacts. If you find this newsletter informative, I encourage you to continue reading. Feel free to pass this newsletter along to colleagues by using this **Forward Email** link. If you've received this newsletter from a colleague and would like to subscribe, please use this **Enter New Subscription** link. If you don't wish to receive this newsletter, use the **SafeUnSubscribe** link at the bottom of this newsletter, and you won't be bothered again.

Your continued feedback on this newsletter is most welcome. Please send your comments to **steve@swqual.com**

In this issue

In **This Months' Topic** I discuss the personality traits and skills required to be a good Tester...

Regular features to look for each month are:

- **Monthly Morsels** Hints, tips, techniques, and references related to this month's topic
- **Calendar**
  Conferences, workshops, and meetings of interest to software engineers, QA engineers, and anyone interested in software development

This months' topic

Do You Have the Right Stuff to be a Good Tester?

IEEE Software magazine published an article on software testing a few years ago titled "What is Software Testing and why is it so hard?" by Jim Whittaker, who said:

> "Software companies face serious challenges in testing their products, and these challenges are growing bigger as software grows more complex. The first and most important thing to be done is to recognize the complex nature of testing and take it seriously.
>
> My advice: Hire the smartest people you can find, help them get the tools and training they need to learn their craft, and listen to them when they tell you about the quality of your software. Ignoring them might be the most expensive mistake you ever make." [1]

Testing is also a very important part of the software development process. Under the best conditions, however, testing can be difficult, frustrating, stressful, and time consuming. In a recent e-newsletter, I discussed the unique challenges that SQA/Test Team Managers face and the special skills required to be an effective manager of such a team. One of the many challenges facing SQA /Test Team managers is hiring and retaining qualified people. For example,

- When interviewing candidates for a testing position, what skills should you look for?
- Are certain people more adept at testing than others?
- Are people with certain personality traits better suited for testing work?
- What kinds of questions should you ask people looking for an entry-level job as a Tester?
- Is testing a path to software development?

Part of the problem facing the SQA profession is lack of consistency. Across the software development industry, the role of SQA varies significantly. In more enlightened companies, SQA engineers and Testers are involved in many activities starting from project inception and continuing through release and maintenance phases. In other companies, Testers are only involved in testing software that has been "thrown over the wall" from development.

I have discussed whether **Software Quality Assurance** is an "engineering profession". I also believe there is a wide variety of tasks that come under the SQA Umbrella. I've identified two distinct roles - **Software Quality Professionals** and **Testers** - and identified the tasks each could be responsible for. Here is what I said about these different roles...

Software Quality Professionals

Examples of tasks that Software Quality Professionals would be involved in could include:

- Define Software Quality Processes and Procedures
- Review Software Engineering Processes and Procedures
- Define, develop, and analyze Quality Metrics
- Identify appropriate Peer Review techniques
- Identify appropriate Software Risk Management practices
- Identify appropriate Static Analysis techniques
- Actively participate in Peer Reviews and Formal Inspections
- Prepare Software Quality Assurance Plans
- Prepare and review Software Test Plans
- Prepare Software Risk Management Plans
- Audit compliance with software engineering processes
- Review Configuration Management and Change Control processes
- Prepare Defect Management procedures
- Participate in software document reviews
- Assess Test Automation needs
- Plan Manual Testing activities
- Review Regression Testing rationales
- Review Test Reports
- Lead Process and Product Audits
- Define Software Release Procedures
- Establish effective working relationships with Software Development
- Establish procedures for Triage and Root Cause Analysis
- Identify and provide appropriate training for Testers
- Actively mentor and coach Testers and provide leadership

Software Quality professionals should have an extensive background in basic quality principles as applied to software development. They should be skilled at applying proven quality principles such as auditing, statistical analysis, root cause analysis, etc. They should also have a considerable amount of domain knowledge of the company's products. They need excellent written and interpersonal communication skills. Certifications from organizations such as ASQ or QAI should be required along with several years of experience.

Training for Software Quality professionals and SQA / Test Team Managers...

Testers

Some of the tasks that Testers could be involved in would include:

- Participate in document reviews (especially the Requirements)
- Participate in writing Test Plans
- Write Test Cases
- Execute Test Cases and perform Regression Testing
- Prepare Test Reports
- Track status of testing activities
- Acquire automated testing skills
- Work with Defect Tracking tools
- Participate in process and product audits

- Acquire domain knowledge
- Build effective working relationships with software engineers and programmers

Testers also need training in basic testing skills as well as domain knowledge of the company's products. The more domain knowledge Testers have, the more likely they are to be able to find defects that customers are likely to find...

Basic training for Testers...

A Tester should be an apprentice to a more experienced Software Quality Professional who can also serve as a mentor.

So what does it take to be a good Tester?

Personality Traits for Testers...

In my experience testing software and as an SQA Manager, I have found that good Testers have the following personality traits (in no particular order):

- **Skeptical**

  Good Testers are skeptical of everything and generally of everyone, especially developers. You see, some developers believe that their code works and sometimes view testing as a waste of time. Fortunately, there are fewer and fewer of these people around today. Most developers now have a strong appreciation for testing and try to work collaboratively with Testers in order to achieve a common goal - which is a higher quality product.

- **Methodical, systematic and somewhat anal**

  Testing is a job that demands methodical, systematic approaches. Good Testers know they need to cover all the bases when testing applications. For example, this means developing test cases that exercise expected as well as unexpected situations.

  Testers also need to have some anal tendencies. By not overlooking the obvious, Testers can help ensure that customers will have a positive experience.

- **Able to put yourself in your customer's shoes**

  I've talked at length about how important **domain knowledge** is for a Tester. Domain knowledge comes from many sources:

  - actually being a customer and using the software you are now testing
  - spending time observing real customers using your software
  - spending time with people who support customers

  By basing tests on domain knowledge, Testers can be much more effective at finding defects that customers would likely find.

- **Instinctive and intuitive**

  Good Testers have a sixth sense about their craft. They seem to know where many of the problems are before they run any tests. This intuition comes from experience and from working with developers over several years. This is not a skill that can be taught - you either have this or you don't.

- **Patience, perseverance, and tactfulness**

  Good Testers need to be patient. Often, the software they are testing is very buggy. As a result, the same tests may need to be repeated several times.

Testers also need to be tactful in their relationship with developers. Why - because Testers criticize developers' work. Handling this role with tact can go a long way to maintaining a positive and productive relationship between the Test Team and Development.

- **Good at solving puzzles**

  People who enjoy the challenge of solving puzzles generally make good Testers. The same thought process that is used to help solve puzzles can be very helpful in trying to find defects in complex software applications...

- **Integrity and a strong commitment to Quality**

  This goes without saying...

Okay, we have identified some important personality traits for good Testers. Now let's look at basic testing skills.

Basic Testing Skills...

To be effective, all Testers need a set of basic, core skills. These are the essential skills I would look for when interviewing candidates for a testing job:

- **Review Requirements**

  Requirements are the grist for the testing mill. Every Tester needs to be able to read, understand, and challenge requirements. **Requirements are often poorly written.** As a result, Testers are often faced with making sense out of a document that doesn't make sense. Identifying requirements that are ambiguous, conflicting, and missing is a critical task that's often overlooked.

  So, one of the most important skills that Testers need to have is being able to determine if requirements, as they are written, are testable...

  Information on training in writing requirements...

- **Estimating the Testing Work**

  As a Tester, you will be facing a significant amount of schedule pressure primarily due to the fact that testing is the last step in the development process before release. As a result, you need to be able to accurately estimate three things:

  - How many tests are required?
  - How long will it take to write the tests?
  - How long will it take to run the tests once?

  This information is essential to prepare an accurate Test Plan - an overall road map for the testing process.

  Familiarity with using a **Requirements Trace Matrix** can be helpful in estimating the work and in providing traceability between requirements and specific test cases.

- **Creating Effective Test Cases**

  Good test cases have a reasonable probability of finding problems that haven't yet been found. Creating good tests requires training in basic testing skills. Testers need to be aware of:

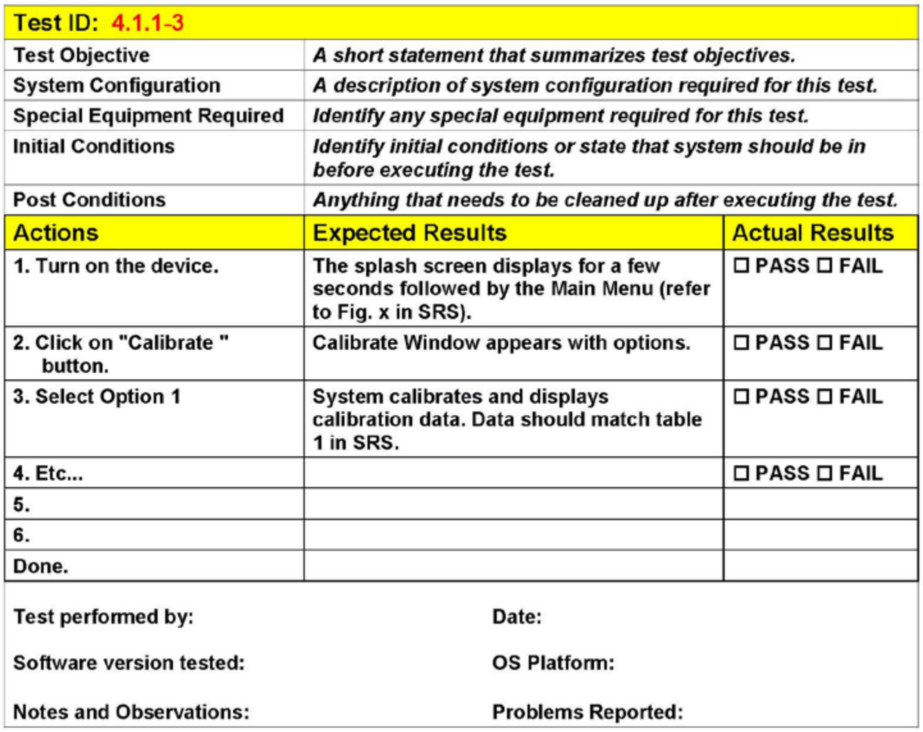
  - levels of testing: unit, integration and system testing
  - methods of testing: black box, white box, and Act Like a Customer ™

Testing)

- testing strategies such as: Structured Basis Testing, Data Flow Testing, Error Guessing, Boundary Testing, Classes of Good and Bad Data, Equivalence Class Partitioning

- test types, such as: positive, negative, boundary, stress, performance, etc.

Testers also need good templates for documenting tests. Here's an example...



Test ID: 4.1.1-3		
Test Objective	***A short statement that summarizes test objectives.***	
System Configuration	***A description of system configuration required for this test.***	
Special Equipment Required	***Identify any special equipment required for this test.***	
Initial Conditions	***Identify initial conditions or state that system should be in before executing the test.***	
Post Conditions	***Anything that needs to be cleaned up after executing the test.***	
Actions	**Expected Results**	**Actual Results**
1. Turn on the device.	**The splash screen displays for a few seconds followed by the Main Menu (refer to Fig. x in SRS).**	**☐ PASS ☐ FAIL**
2. Click on "Calibrate " button.	**Calibrate Window appears with options.**	**☐ PASS ☐ FAIL**
3. Select Option 1	**System calibrates and displays calibration data. Data should match table 1 in SRS.**	**☐ PASS ☐ FAIL**
4. Etc...		**☐ PASS ☐ FAIL**
5.		
6.		
Done.		

Test performed by: **Date:**

Software version tested: **OS Platform:**

Notes and Observations: **Problems Reported:**

A good way to determine if someone has basic testing skills is with a simple test...

Take a Tester Aptitude Test...

Grove Consultants developed the Tester Aptitude. If you are interested in the answers, **send me an e-mail** and I'll share the answers with you.

- **Excellent communication skills**

  Testers must be able to communicate their findings and their questions both verbally and in writing. For example, one of the most important task Testers perform is documenting defects. The ability to describe a problem and what was done to expose the problem in a manner that is clear and succinct is absolutely critical. Sometimes Testers will be asked to review requirements. Here, the ability to verbally explain why a requirement is not testable as it is currently written is also very critical.

- **Dealing with change and multi-tasking**

  In the software business, change is constant. As a result, Testers need to be highly adaptable. Decisions about features are often made late into a project. These decisions often can impact testing work. Also, when testing an application, Testers are often blocked due to defects. Testers need to be able to multi-task and continue testing in another area (or on another project) until the blocking defects are fixed.

Learn how to improve effectiveness of testing...

Summary

Testing is a challenging profession that requires patience, skill, and a keen awareness of the customer. Good Testers need to be acknowledged and rewarded. People coming into the testing profession need training and mentoring if they are to become good Testers. The result of this work will definitely pay off by improving your bottom line and by increasing customer satisfaction.

'Til next time...

Monthly Morsels

Every month in this space you'll find additional information related to this month's topic.

- **References**

  1. Whittaker, J., "What Is Software Testing? And Why Is It So Hard?" *IEEE Software*, Jan-Feb 2000, p. 70-79.

  2. **Marick, B., "Classic Testing Mistakes", presented at STAR EAST, 1997.**

- **Books**

  The following books are required reading for every good Tester:

  Kaner, C. et. al., Testing Computer Software, 2nd edition, Thompson Computer Press, 1993.

  Marick, B., The Craft of Software Testing, Prentice-Hall, 1995.

  Kaner, C., et. al., Lessons Learned in Software Testing, Wiley, 2002.

  Kit, E., Software Testing in the Real World, Addison-Wesley, 1995.

  Beizer, B., Black-Box Testing: Techniques for Functional Testing of Software and Systems, Wiley, 1995.

  Copeland, L., A Practitioners Guide to Software Test Design, Artech House, 2004.

  Myers, G., The Art of Software Testing, Wiley Interscience, 1979.

  Rakitin, S. R., Software Verification and Validation for Practioners and Managers, 2nd edition, Artech House, 2001.

Calendar

Every month you'll find news here about local and national events that are of interest to the software community ...

- **Software Quality Calendar**

  There are many organizations that sponsor monthly meetings, workshops, and conferences of interest to software professionals. **Find out what's happening…**

- **Workshops Offered by Software Quality Consulting**

  Software Quality Consulting offers workshops in many topics related to software process improvement. **Get more info...**

About SQC

Software Quality Consulting provides consulting, training, and auditing services tailored to meet the specific needs of clients. We help clients fine-tune their software development processes and improve the quality of their software products. The overall goal is to help clients achieve Predictable Software Development™ – so that organizations can consistently deliver quality software with promised features in the promised timeframe.

To learn more about how we can help your organization, **visit our web site** or **send us an email**.

Graphic design by **Sarah Cole Design**