CREATING AN OPEN CERTIFICATION PROCESS: THE PROBLEM

CEM KANER & TIM COULTER
FLORIDA INSTITUTE OF TECHNOLOGY

What does it say about a profession if you can (allegedly) become an expert in that field with a few short courses and no industry experience? Over the long term, how does acceptance of this type of certification impact the respect others have for the field?

For example, Villanova University markets a 3-course ($4995.00) sequence that yields the Master Certificate in Software Testing in preparation for the ISTQB certification exam.

- The Villanova brochure for this program encourages the reader to **"Become an Expert—Earn a Master Certificate!"**
- In an online discussion with Tim Coulter (we have a transcript), the Villanova representative claimed
  - that the exam is **"kind of like the bar exam, but for Software Testers to become an ISTQB Certified Tester."**
  - that you need this certificate to get a job as a tester, **"especially if you have no experience" and**
  - that ***this program*** is "**Regionally accredited. The highest accreditation a school can get"** (their marketing literature for this certificate mentions the Middle States Association of Colleges and Secondary Schools).

THE UNDERLYING PROBLEM:

- Software testers—25% to 60% of software project's staff
- No undergraduate degrees in testing
- Few graduate programs offer a specialization in testing
- Relatively few universities even offer testing courses
  - Many of these are inapplicably theoretical
  - Very few schools offer more than one course
  - We think this is unlikely to change in the next decade

Many employers and individual testers want independent certification of testers' knowledge and skill. In the absence of academic credentialing, they look to industrial certifiers.

THE COMMERCIAL OPPORTUNITY:

Certification exams offered by:

- Professional societies, such as the American Society for Quality (ASQ)
- For-profit corporations, such as the Quality Assurance Institute (QAI)
- Entities formed specifically to create / administer these exams, such as the International Software Testing Qualifications Board (ISTQB) and national affiliates (American affiliate is ASTQB).

Profit potential

- From the exam itself
  - ASQ charges $360 for a 75-question multiple choice test.
  - QAI charges $350 for a mixed multiple-choice and essay exam (we understand that this is nonprofit for QAI because the fee compensates essay-exam graders who are not necessarily employees or officers of QAI).
  - ISTQB charges $250 for a 40-question multiple choice test.
- From training courses:
  - ASQ charges $1800 per person for a basic 5-day software quality course.
  - QAI charges $895 for a 2-day exam prep course
  - Private organizations pay ASTQB about $4000 for 3-years' right to offer the Foundation Training Course and typically bill about $2000 for a 3-5 day exam prep course.
  - Some organizations offer certification based on taking a series of courses, without a formal certification exam. For example, the International Institute for Software Testing offers the *Certified Software Test Professional* certificate on completion of 10 one-day courses ($395-$495 each).

Consider the following question (from one of the exam's study guides):

In prioritizing what to test, the most important objective is to:

1. **find as many faults as possible.**
2. **test high risk areas.**
3. **obtain good test coverage.**
4. **test whatever is easiest to test.**

The supplied-correct answer is (2).

We see testing as an empirical investigation, a service that provides stakeholders with quality-related information about the software under test. In terms of prioritization, the primary objective of this service must be to meet the information needs of the stakeholders.

The question presents a factual statement (the most important objective is X) but in practice, in different situations, (1), (2), (3) and (4) have each had their turn as the most important objective.

The answer that we would argue is correct (any of these could be most important, depending on context), is not even an option.

While appearing to call for a simple factual response, the question is actually presenting someone's opinion about how to test, while assuming (without stating) the conditions under which this opinion would be correct.

Here's another sample question:

Which of the following requirements is testable?

1. **The system shall be user friendly.**
2. **The safety-critical parts of the system shall contain 0 faults.**
3. **The response time shall be less than one second for the specified design load.**
4. **The system shall be built to be portable.**

In a Requirements course, if any of these would be considered correct, it would probably be (3), which is indeed the supplied-correct answer. But this is not a Requirements exam, it is a Testing exam.

(a) Answer (3) is arguably ***untestable*** because "design load" probably doesn't specify the caching settings or other performance-critical configuration information about the system under test.

(b) Answer (1) must be testable, because we have a whole field called ***usability testing*** that exists for the assessment of user-friendliness.

The assumption that underlies this question is that a requirement is untestable if it does not specify an oracle, but it is a very controversial question in our field whether you can test without a pre-agreed fully detailed oracle. In practice, testers almost never test with the benefits of such an oracle, therefore any question that insists on such an oracle must be incorrect.

Pay attention to the fundamental difference between requirements specifiers and testers. Specifiers control what the specification says. Testers do not.

In our view, calling this specification untestable is giving the tester an excuse to refuse to test a product or test poorly when s/he would be better served (more respected and less likely to be fired for incompetence) by doing the best testing possible with the product as it is.

Meaningful assessment operates congruently with the bundle of knowledge, attitudes and skills it is intended to assess.

Software testers are professional skeptics. To require them to adopt a compliance mentality, in which they set aside issues of ambiguity, oversimplification, unstated assumptions or controversial conclusions in order to provide the answer expected by an examiner is to demand conduct so far removed from what testers should do as to be invalid on its face.

ENTRENCHING ORTHODOXY IN A RAPIDLY CHANGING FIELD

We don't mind that some people hold (and teach) views or techniques that we consider antiquated. We do mind that in prep courses that teach you how to pass "objective" exams, there is no place for presentation of controversy or thoughtful analysis of the fundamentals.

- Certification is commonly done against a standard and so most of the groups that have developed certification programs have created their own Body of Knowledge as a standard.
- Several of these are so traditional that Kaner considered the ideas dated when he started writing about testing in 1983.
- The conservativism of these materials fosters a conservativism in the courses and so in the attitudes of the trainees. In a field as rapidly changing as computing, for a service provider to be this conservative is a liability.