AKADEMİK
PLATFORM

The Effects of Critical Thinking On the Success of a Software Testing Process

[1]Fatma Molu* and [2]Ayşe Betül Karagöz*
[1,2] Kuwait Turkish Participation Bank, R&D Center, Kocaeli, Turkey

Abstract

Technology comprise of complex and ephemeral integration that can be perceived as simple, fixed, objective or dependable even when they aren't. Major sectors such as finance, health, insurance, marketing, etc. serve their customers via software products. As a result of that, people are using numerous software applications with or without their choices. With the increasing of usage of web and mobile, software is not only a part of business lives, but also daily lives.

This paper focuses on the testing software activities and methodology that aim to bring to customer high quality, free from errors and usable product. Software testing process is different from software development and it has its own terminology and methodology. Some critical success factors of testing process depend on other processes such as software development, project management, and requirement management. In this paper, it is studied about the effects the critical thinking in software testing process.

"Critical thinking is purposeful, self-regulatory judgment which results in interpretation, analysis, evaluation, and inference, as well as explanation of the evidential, conceptual, methodological, criteriological, or contextual considerations upon which that judgment is based." Critical Thinking: A Statement of Expert Consensus for Purposes of Educational Assessment and Instruction, Dr. Peter Facione. When considered from this point of view, critical thinking is the kind of skill that a great tester must have. Critically thinking testers save projects from dangerous assumptions and ultimately from disasters. Besides, software testing process is a realistic domain to prove that critical thinking is not just innate intelligence or a talent, but it is learnable and improvable skill.

Testing is enactment of critical thinking about software. Passing tests cannot prove that software is good. This paper analyzes the numbers, documentation, judgments, terminology, reporting of software testing with critical thinking perspective and their effects of software testing projects' success. Moreover, strong parallels between a software tester and a scientific researcher are discussed and how to use critical thinking in selecting right method in software testing depending on projects are studied. Additionally, testing against requirements is modelled with applying steps of critical thinking in software testing process.

Key words: Software testing, critical thinking, success of process, testing methodology

1. Introduction

Critical thinking is a rich concept that has been developing throughout the past 2500 years. The term "critical thinking" has its roots in the mid-late 20th century [1]. There are several definitions vary according to motivation underlying it. Michael Scriven and Richard Paul states that critical thinking is the intellectually disciplined process of actively and skillfully conceptualizing,

*Corresponding author: Address: Kuwait Turkish Participation Bank, R&D Center, Cumhuriyet District, Ozgurluk Street, No:11/A Sekerpinar, Kocaeli, TURKEY. E-mail address: fatma.molu@gmail.com, aysebetulkaragoz@gmail.com

applying, analyzing, synthesizing, and/or evaluating information gathered from, or generated by, observation, experience, reflection, reasoning, or communication, as a guide to belief and action [2].

“Critical thinking is a way of thinking, and a set of skills, that encourages an informed, aware, systemic, considered and logical approach to deciding what to believe or do. Critical thinking leads to arguments and conclusions that are valid, substantiated and resistant to criticism” [3].

Testing is enactment of critical thinking about software [4].Critical thinking is the kind of thinking that specifically looks for problems and mistakes. A great tester need to be a great critical thinker unlike most of people. With the skill of critical thinking, a tester saves projects from dangerous assumptions and ultimately from disasters [5]. Study and analyze product behaviors and experience new ways to identify, isolate, and characterize bugs are the ways to practice critical thinking in software testing.

2. Critical Thinking Skills for Testers

Higher order thinking skills like critical thinking, creative thinking and problem solving are considered necessary skills for 21st century individuals [6]. On the other hand, technology competencies like using the Internet and its services effectively bring software applications into people’s lives. This case makes testing process of software more critical and increase the responsibility of software testers.

Testers are people who ponder and probe complexity. Basic testing is a straightforward technical process. However, mature testing is a difficult social and psychological process in addition to the technical stuff [4]. Thinking skills are the most important skill that testers can possess. The job of a tester is to investigate and question everything about a product. The ideas of a product, the processes and tools to build the product, the product itself. To test effectively it is a must to think laterally and critically [7].

Using critical thinking skills helps to develop more reasoned arguments and draw out the inferences in others' arguments. As a tester, with heightened critical thinking to support your advocacy, you will have more of your bugs fixed, you will gain the credibility that developers respect, thereby, developers will incorporate your testability suggestions and your enhancement ideas into the software [8].

Critical thinking facilitates a tester in:

- understanding the logical connections between software components;
- identifying assumptions in the design of the software under test;
- evaluating the correctness of the design of the software;
- evaluating the ease of use of the software;
- detecting inconsistencies in the software design;
- identifying mistakes or bugs in the software;

- recognizing which features are the highest priority and why;
- Unearthing bias in one's own thinking and that of the developers of the software [8].

3. Critical Thinking in Software Testing Process

Topics in critical thinking that relate to software testing include:

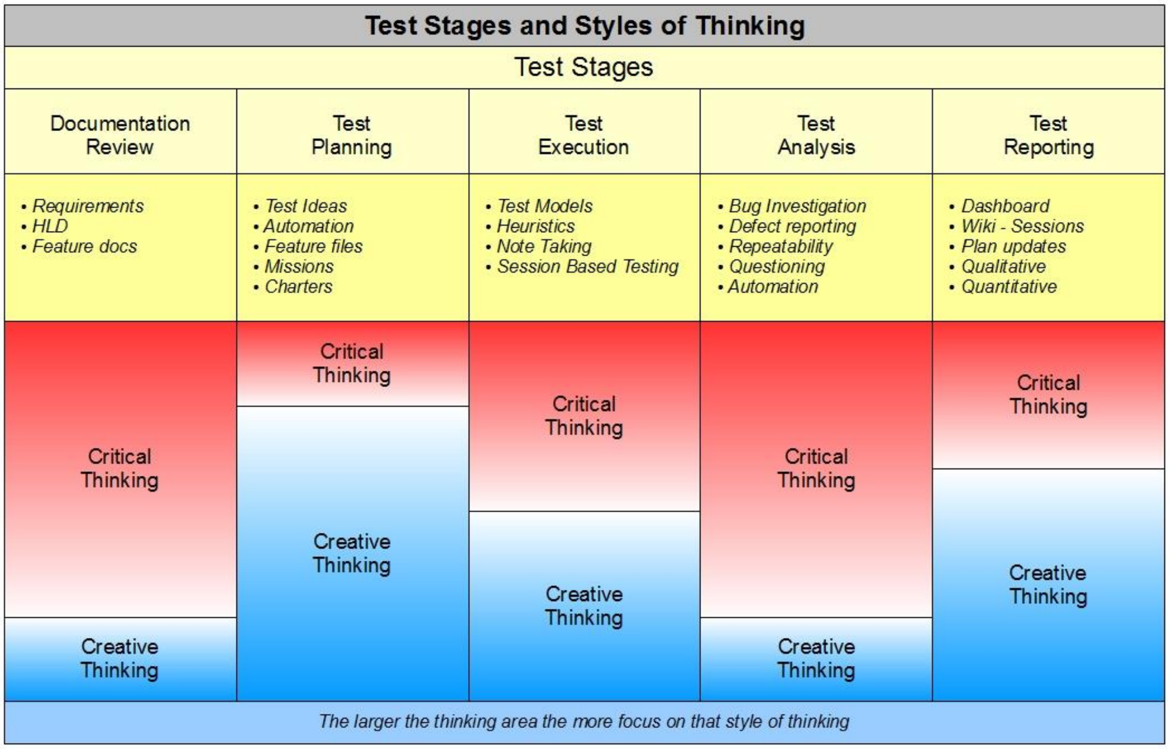
- How to gather and evaluate evidence
- How to make valid inferences.
- How to use different forms of logic.
- What it means to have a justified belief.
- Common fallacies in informal reasoning.
- Meaning and ambiguity in natural language.
- How to make good decision etc [9].

Considering critical thinking in software testing process helps to devise effective testing strategies, better recognize mistakes in the work, know what the testing does and does not prove, and contstruct defensible test reports [9].

3.1. Software Testing Process

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test [10]. There is need different type of thinking methods in different stages of software testing process. Beside technical thinking including domain knowledge, creative and critical thinking takes place in all stages of software testing process. Critical thinking promotes creativity. In this paper, it is focused on critical thinking in software testing process.

In Figure 1, it is displayed test stages and styles of thinking. When reviewing documentation you should be more focused towards critical thinking with a small amount of creative thinking. During the test planning stage this is reversed with more effort placed upon creative thinking rather than critical. When execution your testing there is a balance between being creative to find more stuff to test and thinking critical about what you experiences and the theories you are forming. After you execute your testing you need to be critical of what you have observed and see if there are any problems with the testing you carried out and see if the information you have captured holds up to critical analysis. Finally you need to report on your testing and you need to find creative ways to present your information without overloading the recipient and confusing them with meaningless data. To do this you should have a strong slant towards thinking creatively but also keeping in mind a critical viewpoint of what does that information really represent and how it may or may not be misinterpreted [11].



Test Stages and Styles of Thinking

Test Stages

Documentation Review	Test Planning	Test Execution	Test Analysis	Test Reporting
• Requirements • HLD • Feature docs	• Test Ideas • Automation • Feature files • Missions • Charters	• Test Models • Heuristics • Note Taking • Session Based Testing	• Bug Investigation • Defect reporting • Repeatability • Questioning • Automation	• Dashboard • Wiki - Sessions • Plan updates • Qualitative • Quantitative
Critical Thinking	Critical Thinking	Critical Thinking	Critical Thinking	Critical Thinking
Creative Thinking	Creative Thinking	Creative Thinking	Creative Thinking	Creative Thinking

The larger the thinking area the more focus on that style of thinking

Figure 1 Test Stages and Styles of Thinking [11]

3.2. Test Cases and Assumptions

A test case is the set of steps by which a test is accomplished. One common approach to defining a test case is first to define a test objective; then to define an expected result by which to measure the objective, and then to define a case by which to demonstrate that the expected result confirms the test objective [12].

A tester should start creating test cases by asking questions. Asking questions is a result of critical thinking. Right questions should be asked for right answers. To practice critical thinking, to spark curiosity, to find a way forward when stuck/find different perspectives, to reveal important questions and to generate ideas for improvements [13]. With critical thinking perspective, it is obvious that what you see in a product to write test cases is not all there is.

Assumptions are dangerous in software testing. Because changing assumption would change important behavior. That's why test cases should not base on only specific assumptions. Additionally, assumption ignores controversy. The assumption is counter – intuitive, confusing, obsolete, or has a low probability of being true [4]. As a result of that, assumptions should be avoided when critical things depend on it, the assumption is unlikely to be true, or the assumption is dangerous when not declared.

A critical thinker tester asks himself that how many test cases are needed to test the product represented by the flowchart shown in Figure 2? Traditional test cases include samples for input A such as numbers lower than 70, equal to 70 and more than 70. Simply, a test case should check if it is possible for input A to be a string, varchar, etc. Even it is numeric, its format also should be checked in any case it is binary or decimal format. Moreover, how user interact with the product, how the input is sent to product questions produce more test cases. Additionally, it is possible to write test cases about “Store A” action. Test cases about store functionality, storage capacity, deadline for storing, storing format, security configurations in storing A if it is a critical input, the behavior of the system if same inputs are sent into system, etc. It is clear that, even a simple flowchart needs many test cases with the perspective of critical thinking. When it is a safe critical system or financial system, critical thinking take more significant role in testing those whole systems.

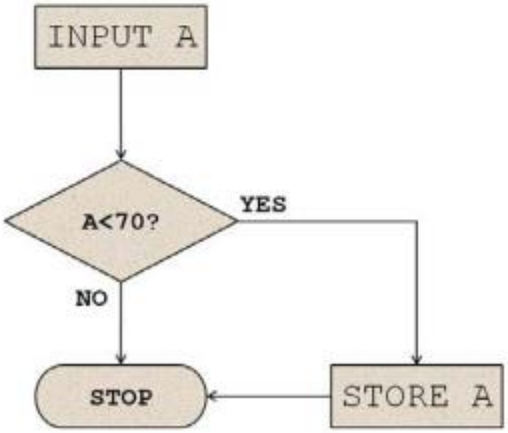


Figure 2 Sample Flowchart to Create Test Case [4]

3.3. Test Execution

It is recommended not to plan or execute tests until you obtain context about the test mission. A tester’s job is to preserve uncertainty when every project team member is certain. It is possible with critical thinking.

In test execution stage, a tester follow critical thinking steps such as observe, evaluate, report. Tester interacts with the product in specific ways to collect specific observations. Then, he applies algorithmic decision rules to those observations and finally reports any failed checks.

While executing test cases, theoretically main aim is to compare the product to its specifications. There are described and actual specifications. In practicing critical thinking in test execution, there are specifications in three categories, which are described, actual, and imagined. Therefore, tester compares the idea of the product to description of it, compares the idea of the product to actual product and also compares the actual product to the description of it. A tester should detect the bugs in missing information, observations not made, poor understanding of risk etc.

3.4. Analysing Test Results

In test results, pass rate is a popular metric. 100% passed test results do not prove that the product is qualified for customers, too. First of all, pass rate does not say anything about test coverage and risk or priority of not passed test cases.

Developer fixes a bug. Tester can’t reproduce it anymore. Must it be fixed? Actually, not. Fixing a bug may cause new bugs in different parts of the product or fixing method can be an unhealthy way for product quality. A critical tester does not make quick positive inferences when the product gives the correct result at the end of testing. Firstly, he checks if it may silently delete system files or not.

It is a fact that, testing all cases in real life is not possible in test environments. Planning to simulate all test conditions and provide all test data is not a realistic plan. Testing everything is never would happen. A tester knows that and prioritizes test cases and analyses test results according to risk analysis and probability to realization.

4. Conclusions

Since the science is credible, curiosity-driven, critical, software testing has also same characteristics. Testing a software product requires continuously challenging and questioning software methods, core beliefs about products. To improve the approach to testing, critical thinking is a necessity. It is a preferred skill for testers.

In software testing process, there many areas to practice critical thinking. Questioning assumptions, staying impartial and being more credible testers is more likely with having critical thinking skill.

Analysing task, identifying assumptions, analysing and classifying, making comparisons, problem solving, questioning and challenging ideas, observing facts versus assumptions and inferences, judging validity of the source and the worth of the evidence are parameters of critical thinking which a successful software testing process includes.

Jerry Weinberg states that “A tester is someone who knows that things can be different.”

References

[1] *Defining Critical Thinking*. The Critical Thinking Community. Available: http://www.criticalthinking.org/pages/defining-critical-thinking/766. 2015.

[2] Scriven M, Paul R. *Critical Thinking as Defined by the National Council for Excellence in Critical Thinking*. 8th Annual International Conference on Critical Thinking and Education Reform, 1987.

[3] Onions, P. E. W. *Thinking critically: An introduction*. 2009.

[4] Bolton M, Bach J. *Critical Thinking for Software Testers*. Star Canada Conference, Toronto. 2014.

[5] Bach J. *Critical Thinking for Software Testers Tutorial*. Satisfice Inc. 2014

[6] Kalelioğlu F, Gülbahar Y. *The Effects of Instructional Techniques on Critical Thinking and Critical Thinking Dispositions in Online Discussion*. Educational Technology & Society, 17 (1), 248-258. 2012.

[7] Ashby D. *Lateral and Critical Thinking Skills for Testers*. 2014

[8] Murphy S. *Why Testers Should Study Critical Thinking?.* Test-soft.com. 2010.

[9] Pettichord B, Kaner C, Bach J. *Lessons Learned in Software Testing: A Context-Driven Approach*, Wiley Publish, 2011

[10] Kaner C. *Exploratory Testing*. Florida Institute of Technology, Quality Assurance Institute Worldwide Annual Software Testing Conference, Orlando, FL. , 2014.

[11] Stevenson J. *Creative and Critical Thinking and Testing Part 1*. http://steveo1967.blogspot.com.tr/2013/03/creative-and-critical-thinking-and.html. 2013.

[12] McMahon C. The importance of critical thinking in software test.

[13] Brickarp E. Exercise: *Critical thinking in software testing*. Test Adventures Blog. 2014.