PODER EXECUTIVO
MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

Oswald Mesumbe Ekwoge

TX-A: UMA ABORDAGEM PARA MELHORAR A EXPERIÊNCIA DO TESTADOR EM PROJETOS DE SOFTWARE

Manaus
2018

OSWALD MESUMBE EKWOGE

TX-A: UMA ABORDAGEM PARA MELHORAR A EXPERIÊNCIA DO TESTADOR EM PROJETOS DE SOFTWARE

Dissertação de mestrado apresentada ao Programa de Pós-Graduação em Informática do Instituto de Computação da Universidade Federal do Amazonas, como requisito parcial para a obtenção do título de Mestre em Informática.

Orientador: Prof. Dr. Arilo Cláudio Dias Neto

Manaus
2018

Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

Ekwoge, Oswald Mesumbe
E36t TX-A: Uma Abordagem para Melhorar a Experiência do Testador em Projetos de Software / Oswald Mesumbe Ekwoge. 2018
85 f.: il. color; 31 cm.

Orientadora: Arilo Claudio Dias-Neto
Coorientadora: Awdren de Lima Fontão
Dissertação (Engenharia de Software) - Universidade Federal do Amazonas.

1. teste de software. 2. testadores. 3. desenvolvedores. 4. qualidade. 5. tester experience. I. Dias-Neto, Arilo Claudio II. Universidade Federal do Amazonas III. Título

PODER EXECUTIVO
MINISTÉRIO DA EDUCAÇÃO
INSTITUTO DE COMPUTAÇÃO

PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

UFAM

FOLHA DE APROVAÇÃO

"TX-A: UMA ABORDAGEM PARA MELHORAR A EXPERIE?NCIA DO TESTADOR EM PROJETOS DE SOFTWARE"

OSWALD MESUMBE EKWOGE

Dissertação de Mestrado defendida e aprovada pela banca examinadora constituída pelos Professores:

Arilo Claudio Dias Neto
Prof. Arilo Claudio Dias Neto - PRESIDENTE

Prof. José Reginaldo Hughes Carvalho - MEMBRO INTERNO

Prof. Eduardo Noronha de Andrade Freitas - MEMBRO EXTERNO

Manaus, 12 de Março de 2018

Av. Rodrigo Otávio, 6.200 - Campus Universitário Senador Arthur Virgilio Filho - CEP 69077-000 - Manaus, AM, Brasil
Tel. (092) 3305 1193 E-mail: secretariappgi@icomp.ufam.edu.br www.ppgi.ufam.edu.br

To God, my Family and friends.

Acknowledgements

I would like to thank God for His grace and mercy upon my life throughout this period, for His wisdom, goodness and mercy. To Him be the glory forever.

I would like to thank my parents, Chief Ekwoge Joseph and Ekwoge Julian for their constant love and support for me, for their encouragements and sacrifices throughout these very challenging years in Brazil. I will forever remain grateful to them.

To my brothers and sisters, Trudy, Hudson, Joel and Persis, my nephew, John-Wesley and niece, Mbolle, for their encouragements, love and company.

To my love Ramona Dana Smith for her unconditional love and support and for being there for me during the good and bad times. Words cannot describe how grateful I am.

To my friends Patrícia Silva, Denize Lima, Tiago Custódio, Marcelo Brito, Romuald Yomkil, Marcineuza Santos, Beatriz Oliveira, Ervili Tarsila, Kevin Takano and the entire PEC-G community, who showed me much love, welcomed me with arms wide open, and made my stay in Brazil wonderful. You will forever have a special place in my heart.

To my friends and laboratory colleagues Silvia Ascate, Josias Lima, Larissa Neves, Karina Villanes, Kariny Oliveira, Paulo Henrique, Laíza Moraes, and Gibson Souza for their help, support, fun, jokes, and all the good moments spent together in and out of the lab. You guys are special!

To my advisor and mentor Professor Dr. Arilo Claudio Dias-Neto for kindly accepting to supervise me and co-orquestrate this research from beginning to end, for the widsom, questions, directions, guidance, rebukes, corrections, and the confidence he had on me. Through him I feel accomplished and a much better person. My gratitude will be forever.

To my co-advisor, Awdren de Lima Fontão for his perseverance, counseling, corrections, advice, wisdom, and for always being there contributing to my professional growth, accompanying my research closely, and for being a friend. I will forever be indebted to him.

To Dr. José Reginaldo Hugues Carvalho and Dr. Eduardo Noronha for accepting to be part of my panel and for their contributions throughout this journey.

Finally, to my professors, Dr. Eduardo Souto, Dr. Eduardo Feitosa, Dr. Eulanda Miranda dos Santos and Dr. Tayana Uchoa Conte, among others, who contributed to making me what I am today. May God Almightly bless them abundantly.

Resumo da Dissertação apresentada à UFAM/AM como parte dos requisitos necessários para a obtenção do grau de Mestre em Informática (M.Sc.)

TX-A: UMA ABORDAGEM PARA MELHORAR A EXPERIÊNCIA DO TESTADOR EM PROJETOS DE SOFTWARE

Oswald Mesumbe Ekwoge

Março / 2018

Orientador: Arilo Claudio Dias Neto

Teste é uma atividade essencial para o desenvolvimento de software. É o ato de executar um produto de software para validar se o mesmo se comporta como previsto e identificar possíveis falhas. Estudos demonstraram que os testes compõem mais de 50% do custo de desenvolvimento. Além disso, muito esforço e ênfase têm sido dedicados em tarefas relacionadas à automação com o objetivo de reduzir custos e a participação do elemento humano em atividades de teste de software. No entanto, teste é também uma atividade humana. Assim, formas eficientes de testar produtos de software para garantia de qualidade exigem uma compreensão melhor e mais abrangente dos sentimentos, percepções e motivações dos testadores, conhecido como *Tester Experience* (TX). Dessa forma, quanto melhor o TX dos desenvolvedores durante o teste, ou qualquer um que assumir o papel de testador, melhor será o resultado. O TX pode ser definido como um meio de capturar como os testadores pensam e sentem sobre suas atividades dentro do ambiente de teste de software, com o pressuposto de que a melhoria da experiência do testador tenha um impacto positivo na garantia de qualidade. Esta dissertação demonstra a importância do TX, propõe conceitos, definições e fatores que afetam o TX. Ela também propõe uma abordagem, denominada, *Tester Experience-Based Approach* (TX-A), composta por diretrizes e atividades que os desenvolvedores (ou testadores) devem seguir para melhorar seu TX, além dos fatores que afetam o TX. Além disso, este trabalho apresenta os resultados de um *survey* realizado com profissionais de desenvolvimento de software para avaliar a TX-A. O estudo avaliou a importância dos fatores e a relevância das diretrizes na melhoria do TX. Os resultados obtidos mostram que mais de 95% dos participantes concordam que os fatores são importantes e as diretrizes são relevantes para melhorar o TX.

Palavras-chave: teste de software, testadores, desenvolvedores, qualidade, Tester Experience, Tester Experience-Based Approach.

Abstract of Thesis presented to UFAM/AM as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

TX-A: AN APPROACH TO IMPROVE TESTER EXPERIENCE IN SOFTWARE PROJECTS

Oswald Mesumbe Ekwoge

March / 2018

Advisor: Arilo Claudio Dias Neto

Testing is a very essential activity for software development. It is the act of executing a software product in order to validate whether it behaves as intended and identify possible malfunctions. Studies have shown that testing makes up more than 50% of the development cost. Besides, much effort and emphasis have now been placed on tasks related to automation with the purpose of reducing cost and the participation of the human element in software activities. Testing is a human-based activity. Therefore, efficient ways of testing software products for quality assurance will require a better and more comprehensive understanding of testers' feelings, perceptions, and motivations, referred to as Tester Experience (TX). Thus, the better the TX of developers during testing, or anyone who takes up the role of a tester, the better the result. TX can be defined as a means of capturing how testers think and feel about their activities within the software testing environment, with the assumption that an improvement of the tester's experience has a positive impact on quality assurance. This thesis motivates the importance of TX, proposes concepts, issues, definition and factors affecting TX. It also proposes an approach, namely, the Tester Experience-Based Approach (TX-A), composed of guidelines and activities that developers (or testers) can follow in order to improve their TX in addition to the factors affecting TX. Moreover, this work presents the results of a survey carried out with software development practitioners in order to evaluate TX-A which evaluates the importance of the factors and the relevance of the guidelines in improving TX. The results obtained show that more than 95% of participants agree that the factors are important and the guidelines relevant for improving TX.

Keywords: software testing, testers, developers, quality, Tester Experience, Tester Experience-Based Approach.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

APP – Mobile Application

DX – Developer Experience

TX – Tester Experience

UX – User Experience

TX-A – Tester Experience-Based Approach

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 - INTRODUCTION

In this chapter, we present the context of this work as well as the motivation for this research. We also present its objectives, methodology and work organization.

1.1 Contextualization and Motivation

Software testing is gaining more and more ground in software development industries. The process occurs in every phase of software development and hence testing must be carried out as early as possible in order to remove faults and stop their propagation, reduce verification cost, and increase the software quality (Fagerholm et al., 2015). In addition, software testing can consume more than 50% of the total development effort (Kanij et al., 2014). A previous study by (Boehm and Basili, 2001) reported that finding and fixing a software problem after delivery is often 100 times more expensive than finding and fixing it during the requirements and design phase. In addition, they report that disciplined personnel practices can reduce defects introduction rates by up to 75%. Given this context, software testers have a very important task in guaranteeing good testing practices.

A software tester is a person whose primary responsibility is to test software before release, helping to increase the reliability of a software product by reporting faults so that they can be fixed (Kanij et al., 2015). The performance of the organization, and the quality of its product, to a large extent, depends on the performance of software testers (Kanij, Merkel, & Grundy, 2011). Most software testing research has focused on the development of standardized, and automated testing methodologies and tools, but the abilities and expertise needed to apply such techniques and tools, by software testers, have attracted a comparatively small amount of attention (Bertolino et al., 2007)(Kanij et al., 2014). In other words, the human-aspects domain of software testing has gotten little attention. Underlying such research is the assumption that software testing should be, for the most part, a systematic, standardized and automated process, meanwhile the personnel who carry out the process are paid little attention. The human impact and experience are important factors in testing-related research, and therefore the most applicable results are gained by observing professional testing personnel (Kasurinen et al., 2009).

Several other studies have shown that human factors are the most important factors for software development, both in terms of performance and quality (Fagerholm and Munch, 2012), (Sackman et al., 1968), (DeMarco and Lister, 1985), and (Mockus, 2010). Hence, when the objective is to analyze and understand software development teams, as related to testing, and all the activities of which it is made up, human factors must be brought to the forefront (Fagerholm et al., 2015). Human factors help to elevate job satisfaction, improve performance, increase productivity of software developers on one hand, and increase user satisfaction on the other hand (Carpretz, 2014). The software testing process is a human-based activity. Testers therefore have a critical role in accompanying each development phase and identifying the required qualities in order to assure software quality.

Despite the importance of the role of the tester in guaranteeing software quality, and despite the importance of human factors in software testing, developers (or anyone who takes up the testing role) face several difficulties and challenges in performing their testing activities: lack of knowledge in testing or on how to use testing tools (Kanij et al., 2014), (Vegas and Basili, 2005). These elements create a berrier that can impede the developer's *testing experience*, and because of this, the quality of the final product is negatively affected. After a software is made available to users, bugs or faults can make users uninstall it and also cause potential new users to look for other software with similar characteristics (Pagano, 2013). Due to the rapid growth of software applications every year, developers and enterprises are losing confidence in to relay on the best testing techniques and practices and adopt economical ways of delivering software to the market (Liu et al., 2010) and (Harty, 2014).

1.2 Problem Description

Nowadays, there are numerous testing tools and techniques, and testers have little (if any) information about the available techniques, usefulness, and generally, how suited they are to the project at hand, upon which to base their decision on which techniques or tools should be used (Vegas and Basili, 2005), (Kanij et al., 2014). However, developers face several challenges in using these testing tools and techniques, which have hindered the proper inclusion of testers in software development. (Kanij et al., 2014) identified several factors that have acted as barriers: lack of knowledge of support tools, high cost, tools difficult to use, time-consuming, and lack of knowledge in testing. Even with information available about the support tools and techniques, the information is generally distributed across different sources of information (i.e. books, articles, internet, people, etc) (Vegas and Basili, 2005). This makes developers lack an overall idea of the techniques and tools and how to use them, and therefore creates a barrier for the proper inclusion of testing activities in software development.

As a means of guaranteeing software quality, there is the need to identify and study factors that reveal testers' experience, and also how these factors can overshadow or minimize the existing negative barriers. These are mostly human factors. This means that if there is a way of introducing good testing practices during software development, one of which involves improving the tester's experience, then this can have a positive influence on quality of the product. One way is to provide a means of improving developers' experience with testing as relates to their perception about testing infrastructure (e.g. platform, techniques, processes, skills and procedures), the value of their contribution (e.g. intention, plans, goals, motivation, commitment and alignment) and how they feel about their work (e.g. respect, team, attachment and belonging). These different kinds of artifacts and activities that a tester might encounter as part of his/her involvement in software development are referred to, in this work, as ***Tester Experience***.

This work thus focuses on investigating good testing practices by means of *Tester Experience*. Our challenge is to provide a definition of the term Tester Experience, as well as its concepts and elements of which it is made up. The concept of Tester Experience is intended to abstract these human characteristics and factors that are intuitive and concrete for practitioners to help them better understand, analyze, design and improve project environments with respect to testers' (or developers when they take up the role of testers) perceptions and feelings. The concept is influenced by similar concepts that aggregate relevant aspects, such as "User Experience" (UX) and "Developer Experience" (DX) concepts, adapted for software testing.

This work, therefore, focuses on proposing and evaluating a *Tester Experience-based Approach* (TX-A), comprising factors and guidelines. That is, TX-A is an instantiation of TX, given that they add meaning to the TX, which left alone, do not offer any help to the tester. For the scope of this study, we define TX, establish factors and their respective guidelines, evaluate them to test our hypothesis whether the factors are important in improving TX, and their guidelines relevant in improving TX. The application of the approach in measuring a tester's TX before, during and after his/her testing activities, as well as measuring the software's quality, are not included in the scope of this work.

1.3 Hypothesis

The hypothesis defined for this work considers the following scenario:

The factors and guidelines that compose TX-A are respectively important and relevant in improving the TX.

1.4 Objectives

In this section, we present the research objectives of this work.

1.4.1 General Objective

Define and evaluate an approach to support the inclusion of testing practices in the software development processes by means of Tester Experience with respect to the types of artifacts and activities that developers may encounter as part of their involvement in software testing, which include the testing infrastructure (i.e. testing tools, programming languages, libraries, platforms, frameworks, processes and methods), feelings about work (respect, attachment and belonging) and value of their contribution (alignment of their goals with those of the project, plans, intentions and commitment).

By defining and evaluating this approach, we hope that the outcome will be useful to software development practitioners in general (when they assume the role of a tester) in improving their TX by providing them with good testing practices.

1.4.2 Specific Objectives

In order to achieve the general objective of this research, we intend to achieve the following intermediary results:

- Define a knowledge body of the software tester's performance and software testing practices;
- Define the concept of Tester Experience and the factors affecting TX;
- Provide a Tester Experience-Based Approach (TX-A) based on the TX factors and their respective guidelines.

1.5 Research Methodology

The research methodology used to achieve the objectives of this research is shown in Figure 1. It is divided into two main phases: the conception phase and the evaluation phase.

1.5.1 Conception Phase

The first part of this phase is aimed at investigating the reasons or factors that reveal the tester's experience. The results obtained would justify the need for a solution to aid developers in their testing activities in order to foster good testing practices in software development. We refer to this solution as TX. Therefore, the first part of the conception phase is a literature review C1, and our solution is described in C2 and C3, as follows:

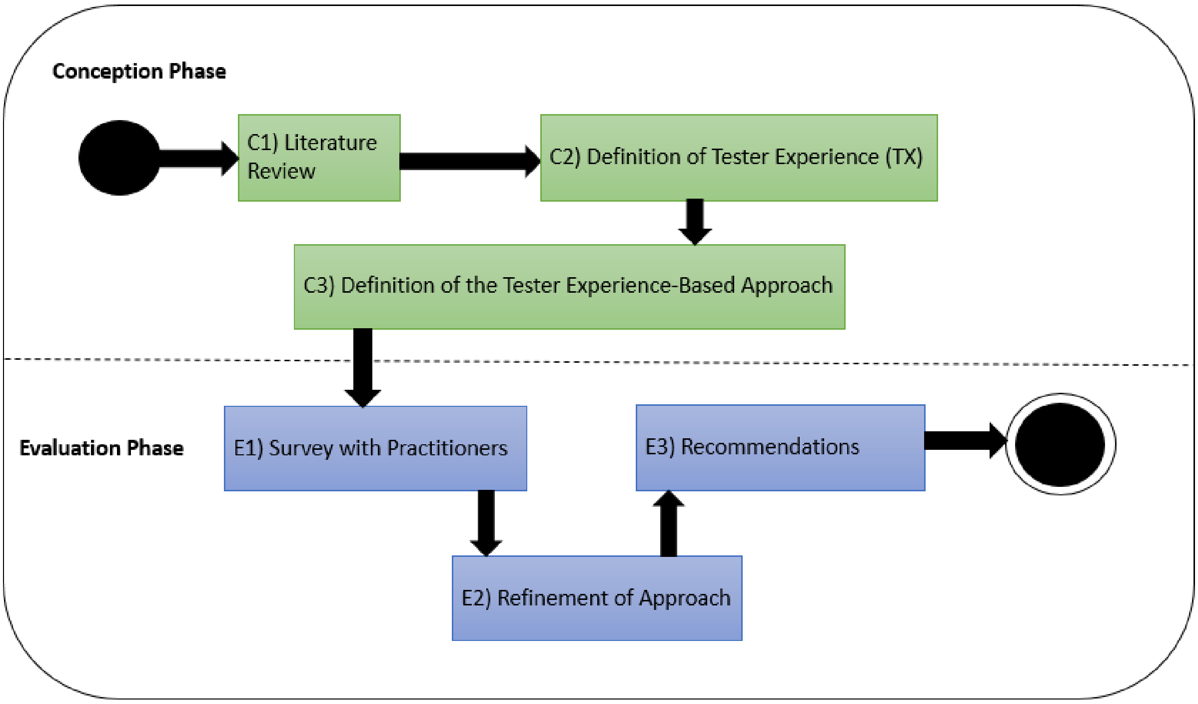

Figure 1: Research Methodology

- **C1)** Literature review:
  - This literature review is based on the software tester's performance and software testing practices in order to identify the factors that affect the tester's experience;
  - As expected results, we hope to identify the factors affecting TX, and also obtain information for the proper definition of TX.
- **C2)** Definition of Tester Experience:
  - Based on the knowledge body constructed from C1, we define a concept referred to as *Tester Experience* which aims at improving the developer's testing experience by providing good testing practices during development;
  - We equally define a TX conceptual framework made up of factors and sub-factors.
- **C3)** Definition of the Tester Experience-Based Approach (TX-A):
  - To facilitate the understanding and the application of the TX factors and conceptual framework, we propose a set of guidelines and activities that accompany these factors.

1.5.2 Evaluation Phase

This phase aims at evaluating the proposed Tester Experience-Based Approach. This will be performed in two main steps:

- **E1)** Survey with software development practitioners, including testers, developers, test managers, project managers, and researchers and professors;
- **E2)** Refinement of the approach based on feedback obtained from survey;
- **E3)** Recommendations for software development practitioners, on which factors and guidelines they should lay focus during their testing activities.

1.6 Document Structure

This document is organized into seven chapters and one appendix, given that this first chapter is organized into: an introduction of this research which includes the motivation of this study, description of the problem which motivates us to carry out this research, research hypothesis, objectives, research methodology and context of this research. The remaining chapters are organized as follows:

- CHAPTER 2 -describes the background of this work, describing important concepts that will aid us achieve our research objectives;
- CHAPTER 3 -describes a literature review on the factors revealing TX;
- CHAPTER 3 -describes the concepts and definition of Tester Experience (TX), providing a definition of, and factors affecting, TX;
- CHAPTER 5 -describes our research proposal of the Tester Experience-Based Approach (TX-A);
- CHAPTER 6 -describes the survey with software development practitioners to evaluate the proposed approach;
- CHAPTER 7 - presents this work's conclusion, limitations and future works;
- Appendix 1 - Presents the questionnaire applied during the survey (Chapter 6).

CHAPTER 2 - BACKGROUND

In this chapter, we present the background of works related to the foundation and evolution of this work, presenting core ideas and justifying the need of our study. It is divided into three sections: Testing in Software Development, Experience in Software Engineering and Final Considerations. These are described below.

2.1 Testing in Software Development

Testing, as defined by (Bach, 2000), (Cem Kaner, 2002) and (Myers et al., 2011), is “the process of executing a program with the intent of finding errors”. In fact, testing is one of the fundamental requirements of software development to measure the quality of a software and to avoid bugs (Amen et al., 2015). Software quality is defined as “the degree to which a system, component or process meets customer or user needs or expectations” (IEEE 730-1989). Due to the rapid growth of software every year, developers are losing confidence in to relays on the best testing techniques and practices (Liu et al., 2010) and (Harty, 2014).

According to (Majchrzak and Schulte, 2015), if the software quality is low, it is partly as a result of problems with testing. In this context, software testing is a main challenge in software development. Testing is a cumbersome task (Majchrzak, 2012) and requires sophistication. For instance, mobile application (app) testing poses several particularities: apps are not developed on the platforms they run (mobile devices) but on a PC (Majchrzak and Schulte, 2015). Testing on emulators will not yield the same results as testing natively. This also makes it laborious and hard to automate. Moreover, tools support currently is limited.

Testing is carried out by testers. The elements of testing which are applied by testers have been defined by ISO/IEC/IEEE 29119-2. Some of these include: testing: set of activities conducted to facilitate discovery and/or evaluation of properties of one or more test items; test process: provides information on the quality of a software product, often comprised of a number of activities, grouped into one or more subprocesses; test subprocesses: include test management and dynamic (and static) test processes used to performa a specific test level (e.g. system testing, acceptance testing) or test type (usability testing, performance testing) normally within the context of an overall test process for a test project; test procedure: sequence of test cases in execution order, and any associated actions that may be required to set up the initial preconditions and any wrap up activities post execution; test case: set of test case preconditons, inputs (including actions, where applicable) and expected results, developed to drive the execution of a test item to meet test objectives, including correct implementation, error identification, checking quality, and other valued information; test condition: testable aspect of a component or system, such as a function, transaction, feature, quality attribute, or structural element identified as a basis for testing; test environment: facilities, hardware, software, firmware, procedures and documentation intended for or used to perform testing of software; test plan: detailed description of test objectives to be achieved and the means and schedule for achieving them; test specification: complete document of the test design, test cases and test procedures for a specific item; test strategy: part of the test plan that describes the approach for testing a specific test project or sub-process; test technique: activities, concepts, processes, and patterns used to construct a test model that is used to identify test conditions for a test item, derive corresponding test coverage items, and subsequently derive or select test cases.

According to (Vegas and Basili, 2005), one of the major problems faced by testers is to select a suitable testing technique or tool to test a software system. Despite the numerous techniques and tools that exist, many are never used and just a few are used over and over again. Moreover, testers have little or no information about the availability of the tools and techniques, as well as their usefulness or their suitability to the projects at hand, and how to base their decision on which technique or tool to use. This difficulty calls for the need of a set of guidelines or an approach that can help testers adopt good testing practices in software development.

According to (Amen et al., 2015), before deciding to adopt any testing techniques on software, it is necessary to have a testing strategy in order to meet users' requirements, specifications and to avoid negative feedback from users. The concept of human factors is emphasized by (Briand and Labiche, 2004), who state that the human impact and experience are important factors in testing-related research, and therefore the most applicable results are gained by observing professional testing personnel. Therefore, a comprehensive testing is crucial to direct high quality of applications and user satisfaction. This can be achieved through the understanding of human factors (Fagerholm and Munch, 2012). In order words, in order to analyze and understand software testing, and all the activities of which it is made up, human factors are considered an essential element (Fagerholm et al., 2015). In the next sub-section, we describe these human factors.

2.2 Experience in Software Testing

The software testing process is a human-based activity carried out by the tester. It occurs in every phase of software development since failures can occur in any development phase. Therefore, software testers must be involved in the development process from the very early development stages. As a means of guaranteeing software quality, there is the need to identify and study the human factors that reveal testers' experience and also how these factors can overshadow the existing negative barriers hindering testers from achieving their testing goals. Human factors involved in software development are vital to a successful completion of a software project (Carpretz, 2014), (Gannon, 1979). They help to elevate job satisfaction, improve performance and increase productivity of software developers. Given this context, human factors have been studied in Software Engineering in the form of "experiences". These experiences include User Experience, Developer Experience, Customer Experience and Brand Experience (Fagerholm and Munch, 2012).

The (Merriam-Webster) dictionary defines "experience" as "the process of doing and seeing things and of having things happen to you; skill or knowledge that you get by doing something or the length of time that you have spent doing something (such as a particular job). In general, it refers to both immediately perceived events as well as the memories of events and the knowledge gained by interpreting and reflecting on remembered events". Human experience is our ability to process data that is limited, maintaining an individual mental state of reality, which can be used to interpret new data (Fagerholm and Munch, 2012).

In the following subsections, we describe the four main types of experiences, mentioned above, that contribute to human factors in software development.

2.2.1 User Experience (UX)

UX involves a person's perceptions and responses resulting from the (anticipated) use of a product, system or service (Hassenzahl, 2008). This includes perceived product properties such as value, desirability, and usefulness. It is a momentary, primarily evaluate feeling (good-bad) while interacting with a product or service (Hassenzahl, 2008). By that, UX shifts attention from the product and materials (content, function, presentation, interaction) to human and feelings – the subjective side of product use. Good UX is the consequence of fulfilling the human needs for autonomy, competency, stimulation (self-oriented), relatedness, and popularity (others-oriented) through interacting with the product or service (hedonic quality). Therefore, the term UX is scoped to products, systems, services, and objects that a person interacts with through a user interface. These can be tools, knowledge systems or entertainment services.

2.2.2 Developer Experience (UX)

DX consists of experiences relating to all kinds of artifacts and activities that a developer may encounter as part of his/her involvement in software development

(Fagerholm and Munch, 2012). The term “experience” refers to involvement, not to being experienced, although the two are interlinked. DX could be divided into experiences regarding (i) development infrastructure (e.g. development and management tools, programming languages, libraries, platforms, frameworks, processes and methods), (ii) feelings about work (e.g. respect, attachment, belonging), and (iii) the value of one’s own contribution (e.g. alignment of one’s own goals with those of the project, plans, intentions and commitment) (Fagerholm and Munch, 2012).

2.2.3 Customer Experience (CX)

CX occurs when a customer interacts with a supplier of goods or services (ISO9241-210:2010, 2010). It can also mean an individual experience over one transaction: the customer experience concept includes both the cumulative experience and episodic experience. It includes the experience of both a product or service, and the process during which the customer interacts with the supplier. This interaction is made up of three parts: the customer journey, the brand touchpoints the customer interacts with, and the environments the customer experiences (including digital environment) during their experience. It is measured by the individual's experience during all points of contact against the individual's expectations (Hassenzahl, 2008). Unlike UX, it does not require a user interface. It could be face-to-face. Moreover, a customer has a stronger relationship with the product or service supplier than a mere user.

2.2.4 Brand Experience (BX)

BX refers to as subjective, internal customer responses and behavioral responses evoked by brand-related stimuli that are part of the brand’s design and identity, packaging, communications, and environment (Brakus et al., 2009). In marketing, a brand is a “name, term, design, symbol or any other feature that identifies one seller’s good or service as distinct from those of other sellers” (American Marketing Association). In creating BX, the goal is to develop or align the expectations behind the brand experience to create an impression that the brand has qualities and characteristics that make it unique or special. A brand is therefore one of the most valuable elements in an advertising theme.

2.3 Final Considerations

In this section, we have discussed the concepts of testing in software development and experiences in Software Engineering. We also observed that testing as a whole is very tedious and cumbersome and with respect to software testing, it comprises of a series of techniques, tools and processes that make the whole process complex. Therefore, as a means of contributing to providing a solution to this complexity, human factors are brought into play with the objective of analyzing and understanding software testing, and all the activities of which it is made up. Human factors have a major impact on the software development process and quality of the software produced, ranging from the software development team to the end users' satisfaction.

Moreover, the quality of the software product can be a reflection of the software testing activities and practices by developers. Therefore, the structure of this research seeks to understand these different aspects, that is, good development (testing) practices within the software organization, namely *Tester Experience*, which investigates ways of improving the developer's testing experience (chapter 4), which can contribute to the quality of his/her product.

CHAPTER 3 - A LITERATURE REVIEW ON FACTORS REVEALING TESTER EXPERIENCE

In this chapter, we present a literature review to extract the factors that reveal testers' experience, testers' performance, or testing activities.

3.1 Introduction

The correlated works identified in this literature review were extracted through a *Snowballing* approach. This approach was chosen due to a recommendation by (Wohlin, 2014) as a first search strategy, and a good alternative to the use of database searches, such as Systematic Mapping studies. We opted for this approach since the concept of TX is new, and it has never been defined. Furthermore, this approach is suitable because very there are few works that research on human factors in software testing.

Snowballing refers to using a reference list or citations of a paper to identify additional papers (Wohlin, 2014). Furthermore, snowballing could benefit from not only looking at the reference list (backward snowballing) and citations (forward snowballing), but to complement it with a systematic way of looking at where papers are actually referenced and where papers are cited. The first step in Snowballing is to carry out a literature search, and (Wohlin, 2014) recommends to use Google Scholar as it helps to avoid bias in favor of any specific publisher. After defining the start set, the backward or forward Snowballing procedures are implemented. In both backward and forward snowballing, we define inclusion/exclusion criteria to include papers in the next set for further investigation. This process goes on and on until no new papers are found, with the final inclusion of a paper based on reading the full paper and not just the abstract.

The rest of this chapter is organized as follows: section 3.2 presents the study plan, and design; section 3.3 presents the discussion; section 3.4 presents the threats to validity, and section 3.5 presents the conclusion.

3.2 Study plan and Design

3.2.1 Goals

The goal of this study is to **analyze** testers' performance, experiences and activities in software development **with the aim of** characterizing them **with respect to** factors affecting TX **from the point of view of** software testers **in the context of** software testing.

To achieve this goal, we defined the following inclusion/exclusion criteria. For inclusion criteria, we defined:

- Papers or technical reports related to testers' performance, activities or experience in software development;

Any paper which does not meet these criteria is excluded from our analysis. As research question, we defined the following: "*What are the factors that reveal the tester's experience with respect to software testing?*".

3.2.2 Execution and Results

Due to the difficulty in identifying relevant papers related to our study goals, our initial start set was comprised of just three papers: (Rodrigues and Dias-Neto, 2016), (Kanij et al., 2011), and (Cunha and Greathead, 2007). From this start set, we obtained a few other papers, including: (Kanij et al., 2014), (Kanij et al., 2015), (Kanij et al., 2013), (Smith et al., 2016), (Patwa and Malvija, 2014), (Merkel and Kanji, 2010). The factors extracted from these papers are summarized in Figure 2 below, which answers our defined research question.

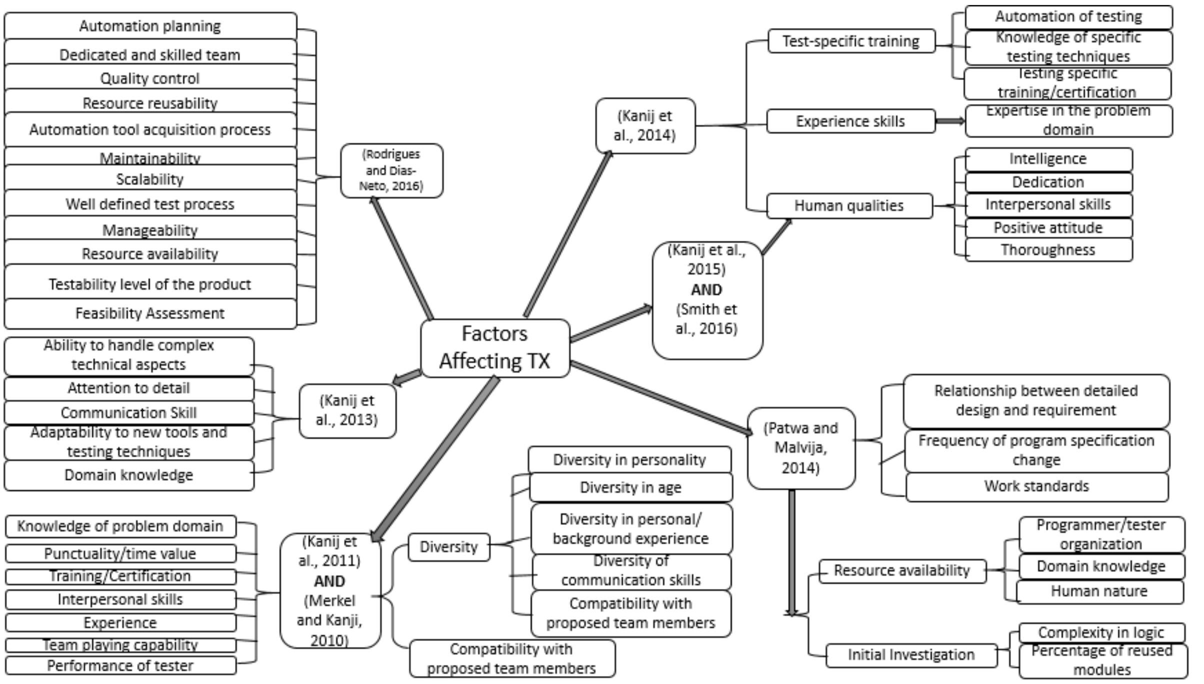

Figure 2: Extraction of Factors Revealing Tester Experience

From the identified papers above, we can observe the following:

According to (Cunha and Greathead, 2007), there is a connection between personality and testing, as measured by psychological testing and debugging performance and some professionals show skill in debugging while others are less successful. If a company organizes its employees according to their personality types and their potential abilities, productivity and quality may be improved. This conclusion can be implicated directly into software testing.

(Kanij et al., 2011), (Kanij et al., 2014) and (Merkel and Kanji, 2010) carried out surveys of factors affecting software testers. They identified the following factors as relevant to their survey:

a) Performance: factors influencing the software testers' performance include knowledge of specific testing techniques, expertise in the problem domain, testing specific training/certification, intelligence, dedication, punctuality/time value, thoroughness, positive attitude and interpersonal skill;
b) Influence on Automated Tools: the most common benefit of automated tools was a time-saver, that it, automated tools lead to increased testing speed, improved productivity and less manual testing effort;
c) Experience in Testing: experience helps testers to prioritize work and is useful for better planning and analysis. Moreover, experience helps to increase knowledge in the domain and the product, and helps to grow adaptability in different situations. Experience is not only fruitful when software testers learn from their past; a variety of new challenges is also important;
d) Personality Characteristics: these characteristics are based on the "Big Five" Factor Model for Personality, which is one of the most popular models for personality traits in modern Psychology research. These factors include Extraversion, Agreeableness, Neuroticism Conscientiousness, and Openness to Experience. In order words, these characteristics could mean good interaction with the outward social world, open-mindedness/openness to new experiences/intellectual curiosity, tendency towards negative emotionality, qualities like trust or modesty, and finally personal organization;
e) Training/Certification: the authors could not come to any definite conclusion because their survey provided mixed feelings. On one hand, certification courses were more theoretical and general, and on the other hand, they helped to better understand the work, find new ways and approaches for testing and helped to safe testing effort.

(Kanij et al., 2015) identified other factors affecting software team performance in addition to those already mentioned above. These include:

a) Knowledge in specific problem domain: this factor is important and different from Experience, because in the dynamic word we live in, new problems arise every day and we are constantly faced with new challenges. Therefore, no matter how experienced the software tester might be, situations might arise where the problem domain is neither known nor completely understood.

b) Compatibility with proposed team members: the authors could not come into any conclusion on how compatibility affects team performance, and therefore they proposed a further research on the topic.

Apart from these factors, the authors identified different types of team diversity that tend to influence performance. These include: diversity of personality, diversity of personal/background experience, diversity of age and diversity of communication skills.

To support the work of (Kanij et al., 2014) as relates to Personality Characteristics, several works have examined the connection between ability in aspects of testing and various personality related factors. (Smith et al., 2016) present results from a survey about beliefs, practices, and personality of software engineers in a large software company. Among the two most commonly used personality inventories that exist in psychometric research, that is, the Meyer-Briggs Type Indicator (MBTI) and the Five Factor or "Big Five" Model, the latter was selected due to its stronger theoretical and empirical basis, as well as a higher test-retest reliability. They observed that managers are more conscientious and more extraverted. The authors emphasize on the need for further research in personality testing, because more work is needed for a conclusive answer, especially with the social aspect of the software developer practitioner.

(Patwa and Malvija, 2014) carried out a survey on the analysis of factors of different phases of software development which affect software testing in object-oriented software with the opinion of people who are engaged in software development phases. They list the top ten factors classified according to the different development phases as follows:

a) Analysis and Design Phase: relationship of detailed design and requirement (rank 1), frequency of program specification change (rank 5), work standards (rank 6) and requirements analysis (rank 9);
b) Coding Phase: Programmer/tester organization (rank 3), programmer/tester skill (rank 4), domain knowledge (rank 7) and human nature (mistake and work omission) (rank 10);
c) Initial Investigation: Complexity in logic (rank 2) and percentage of reused modules (rank 8).

(Kanij et al., 2013) carried out a performance appraisal of software testers with the aim of collecting feedback from project managers. They identified five personality attributes of software testers, which include domain knowledge, adaptability to new tools and techniques of testing, communication skill, attention to detail and ability to handle complex technical aspects.

Finally, (Rodrigues and Dias-Neto, 2016) carried out a survey on the relevance and impact of critical success factors in the software test automation lifecycle. They identified 12 critical success factors collected from technical literature and evaluated their relevance according to practitioners' view. These factors, in order of impact, include: feasibility assessment, testability level of the product, resource availability, manageability, well defined test process, scalability, maintainability, automation tool acquisition process, resource reusability, quality control, dedicated and skilled team and automation planning. These factors are not only important to test automation in particular, but to software testing as a whole because their impact on software test automation will equally influence the tester's experience, since test automation is considered a factor that influences tester experience.

From the works described above, we can conclude that there are several factors that affect TX. However, there is no standard or systematic definition of these factors. Therefore, a formal way of defining TX as well as its factors is necessary. In the next section, we present a brief discussion on the factors obtained from the literature review.

3.3 Discussion

From our research question "*What are the factors that reveal the tester's experience with respect to software testing?*", we were able to identify some of the factors that possibly reveal Tester Experience. These factors range from testing tools and techniques such as automated testing, technical aspects, domain knowledge and skills to the value of the tester's contribution, such as communication skills, attention to details, dedication, positive attitude, thoroughness, and intelligence, to finally how the tester feels about his/her work, such as team playing capabilities, performance, and relationship with other team members. These factors will be elaborated in more details in the next chapter that defines and conceptualizes TX.

3.4 Threats to Validity

a) **Internal validity**: the instrument used for the literature review (Snowballing) was adequate for this research it is indicated as a first search strategy (Wohlin, 2014) and the concept of Tester Experience is new. Even though the number of papers identified were few, we believe they were adequate enough, given that there are very few works that focus on software testers or human aspects of software testing. Furthermore, the selection of studies was based entirely on following the inclusion/exclusion criteria and using Google Scholar;

b) **External validity**: the papers extracted and selected meet our research objectives in identifying the factors that reveal Tester Experience. Even though there are few papers, they represent almost every aspect of the tester's activities and performance within the software development environment;

c) **Constructo validity**: as observed, the literature review went through the process of Snowballing, following a defined protocol which included inclusion/exclusion criteria, and the chosen method was adequate to obtain significant results. Furthermore, the study underwent thorough revision to make sure that bias was avoided and that no important papers were excluded or left out;
d) **Validity of conclusion**: the goals of the literature review were achieved due to well-defined objectives, and the activities that led to this achievement were systematic, revised and monitored.

3.5 Conclusion

In this chapter, we carried out a literature review to identify the factors that reveal Tester Experience. Several factors were identified as shown in Figure 2. Given this context, we can now be able to formally define and contextualize the concept of TX. In the next chapter, we discuss this concept, and in addition to the factors identified in this chapter, we extract other important concepts from User Experience and Development approaches, as well as from psychology. With this, it would be possible to classify the identified factors into categories. These and more are described next.

CHAPTER 4 - TESTER EXPERIENCE: CONCEPTS, ISSUES AND DEFINITION

In this chapter, we present a formal definition of Tester Experience, its concepts as well as its influencing factors.

4.1 Introduction

Some concepts have been defined in order to understand the participation and experience of different actors involved in software development (users and developers, for instance). Two of these concepts are **User Experience** (UX) and **Developer Experience** (DX). UX is a term that captures how people feel about products, systems and services (ISO9241-210:2010, 2010). DX is a means of capturing how developers think and feel about their activities within their software development environments (Fagerholm and Munch, 2012). The concept of TX was influenced by UX and DX concepts, adapted for software testing with an assumption that an improvement of the tester experience would have a positive impact on the software development project outcomes, and consequently on the software project quality. Therefore, this chapter therefore proposes the definition of *Tester Experience* (TX). This concept is intended to abstract human characteristics and factors that are intuitive and concrete for practitioners to help them better understand, analyze, design and improve project environments with respect to testers' perceptions and feelings. This work has been published in the 41st Annual IEEE Computer Software and Applications Conference (COMPSAC), and can be referenced at (Ekwoge et al., 2017).

From the literature review carried out on tester's performance during software testing, we were able to identify the factors affecting testers' experiences as well as establish the definition and concepts of TX. We were also able to extract three main categories of factors affecting TX. These include <u>Cognition</u>, <u>Conation</u> and <u>Affection</u>. Each of these three factors was also divided into sub-factors, which are explained later in this chapter. Therefore, we formally establish a definition of TX and the factors that influence it and also draw a distinguishing line between the three approaches (UX, DX and TX).

The rest of this chapter is divided as follows: in Section 4.2, we contextualize the scenario of TX. In Section 4.3, we define TX, as well as a taxonomy to understand this concept, and also identifies the factors that influence TX. In Section 4.4, we present the final considerations of this charpter.

4.2 Contextualization

To define TX, we assume that several factors that influence UX and DX could also influence TX, which in turn will influence the outcome of the software product quality. The word “tester” refers to anyone who is engaged in testing a software application in order to reveal failures based on the software requirements, including developers, when they have to test their software product (Beer and Ramler, 2008), while “Experience” refers to the involvement in the software testing process and not on the testability of the software.

UX has evolved beyond user interface design and it does not only focus on avoiding usage defects, increasing robustness or ensuring safety. It has shifted to efficiency and ease of use, as well as appropriate use and fitness for purpose and therefore, the entire experience of using the software product or service. Likewise, DX relates to all kinds of artifacts and activities that a developer may encounter as part of his involvement in software development (Fagerholm and Munch, 2012).

Although the software tester is involved in all software development process phases, his/her activities present peculiar differences in characteristics from the UX and DX. That is, Tester Experience presents factors that are peculiar to the functions of the tester for software quality assurance, which are different from factors affecting Developer and User Experiences, even though they might all present some similarities. One justification for the difference is, while UX and DX are psychologically defined as “constructive”, that is, they “build” something to meet customer’s requirements, a tester’s job is often psychologically defined as “destructive”, that is, he/she attempts to “break” the software constructed by programmers in order to encounter faults (Kanij et al., 2015). This, therefore, raises a fundamentally different task set, mindset and work approach for the tester’s experience.

In Table 1, we draw a parallel to the tester perspective, where the end goal is neither to use nor create a product or service, but to “destroy” it or encounter problems in the software product developed by developers. In terms of positive experience, appropriate or efficient use, UX is the outcome of the user after using the product with respect to usability and user-centered design (Law et al., 2009); DX is the outcome of the developer’s perception about the process-product relationship with a specific context or the process models applied during the development process (Fontão et al., 2016). These model processes include: descriptive process models which describe how the process should be carried out in specific settings, and the prescriptive process, which recommends or prescribes guidelines, stages or techniques which, if implemented correctly, are thought to improve performance in specific aspects of the product or project. The end goal of DX is the product or service created which meets user requirements. TX involves the whole experience of activities to ensure identification of bugs (error, flaw or failure in a computer program that causes it to produce an incorrect or unexpected result, or behave in an unintended way (ISTQB Exam Certification), testing tools or techniques with the main objective of testing the product, therefore acting as a preventive process. The end goal of TX is to encounter problems in the software product, or functionalities that do not meet user requirements.

Table 1. Comparison Between UX, DX and TX Perspectives.

Focus	User perspective	Developer perspective	Tester Perspective
Appropriate use, fitness of purpose + efficient use	User-centered design	Understanding the process-product relationship for a specific context	Product oriented activity involving activities to ensure identification of bugs/errors/ defects Quality-based design
Efficiency and ease of use	Usability	Descriptive process models, adaptive process models	Testing tools or techniques such as test automation
Avoid usage defects, increase robustness, safety	User interface design	Prescriptive process models	Preventive process focusing on testing the product to encounter problems
End goal	Use product or service	Create product or service	Evaluate product or service quality

Until now, most software testing research has focused on the development of systematic, standardized and automated testing methodologies and tools (Kanij et al., 2014). However, there has been little attention on the abilities and expertise needed to apply such techniques and tools. Such abilities include personality traits, education and expertise (Kanij et al., 2014). While testing tools and methodologies are important, human factors influencing tester experience are equally important. Testing tools and methodologies all depend on human factors. Other traits considered crucial for tester experience include domain knowledge, experience in testing, skills and human qualities. In this section, we treat each of these factors in details and explain why we think they influence the tester's experience and hence, are software quality factors. Therefore, we believe that good tester experience will result in good practices in software testing techniques, tools and methodologies, and therefore software quality assurance.

4.3 Concepts and Definition of TX

Tester Experience (TX) consists of all the experiences relating to artifacts and activities that a tester may encounter as part of his/her involvement in software development. These experiences have been revealed through factors, which we have been able to identify in the previous subsection through a literature review.

The factors affecting TX can be grouped into one of three categories, as described by (Hilgard, 1980), who carried out a tripartite classification of mental activities. The practice of software testing is intellectual or psychological, relying on the capabilities of the mind and therefore we adopt the classification to our context of TX (which was also adopted by (Fagerholm and Munch, 2012). From here forward, we call the factors identified in Figure 3 as sub-factors, and we call these three categories factors. Hence, we can categorize the three factors as follows:

a) **Cognition** (attention, memory, producing and understanding language, problem-solving and decision making): this category consists of factors that affect how testers perceive their testing infrastructure on an intellectual level. The testing infrastructure can include testing tools, programming languages, platforms, frameworks, processes and methods;
b) **Conation** (impulse, desire, volition, striving): this category consists of factors that affect how testers see the value of their contribution. Intentional, planned activity with personal goals that are properly aligned with the goals of other is likely to increase the sense of purpose, motivation and commitment and hence positively affect TX;
c) **Affection** (feeling, emotion): this category consists of factors that influence how testers feel about their work. Respect belonging is a social factor that creates a feeling of security. Attachment to persons, teams, or even work habits, also belongs to this category.

Each of these categories or dimensions is made up of a number of complex sub-factors (Figure 3).

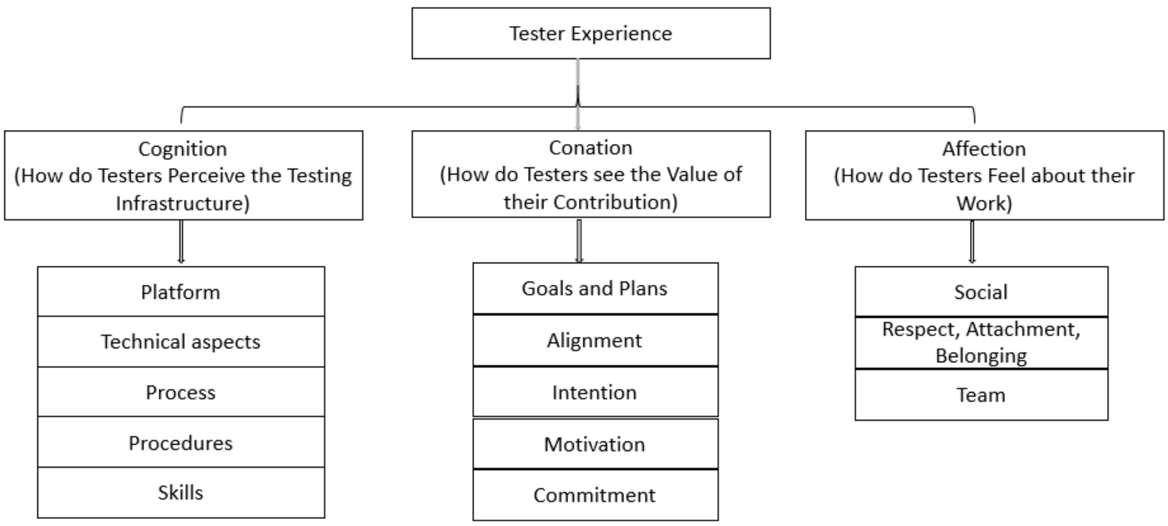

Figure 3. Factors influencing TX divided into Sub-factors

In Figure 4, we present the taxonomy of TX, including the three factors that influence TX, as well as all their sub-factors. We now look at each sub-factor in more details. In the first factor, ***Cognition*** (How do Testers Perceive the Testing Infrastructure), the sub-factors include:

a) **Platform:** common software platforms include, for instance, Windows, Linux and Mac operating systems for desktop/web platforms and Android, iOS, and Windows Phone, for mobile platforms (Gavalas and Economou, 2011). Therefore, the tester's knowledge of these platforms will influence his/her perception about testing applications on these platforms, and therefore influence his/her tester experience;

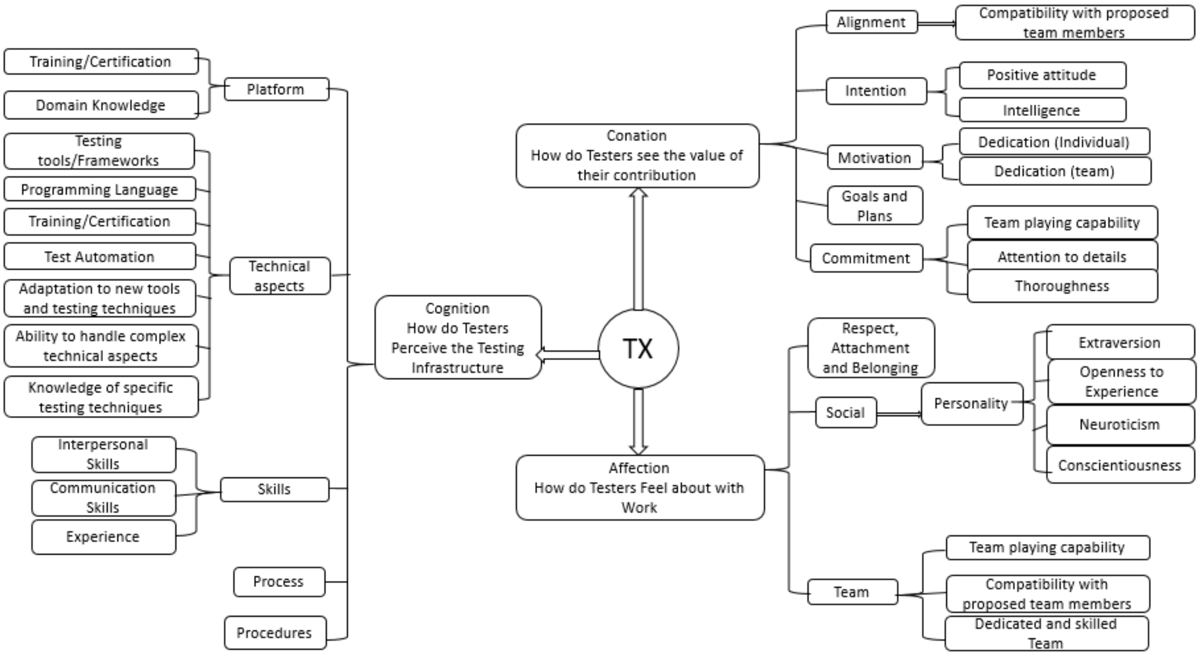

Figure 4. Tester Experience – Conceptual Framework

b) **Technical Aspects:** these include *testing tools, techniques* or *frameworks* to facilitate testing, *test automation* techniques and tools, *programming language*, *training or certification* on testing techniques or tools, *adaptation to new tools and testing techniques*, *ability to handle complex technical aspects and knowledge* about specific testing techniques. Some well-known testing techniques/criteria include: functional techniques (equivalence class partitioning, boundary value analysis and random testing), structural testing (control-flow, such as sentence coverage, decision coverage, path coverage) (Vegas and Basili, 2005). Knowledge of technical aspects will influence the tester's perception about the infrastructure and therefore influence his testing experience;

c) **Procedures:** a test procedure is a fundamental specification of test cases to be applied to one or more target program modules (Panzl, 1976). Test procedures facilitate software testing by allowing individual modules or arbitrary groups of modules to be thoroughly tested outside the environment in which they will eventually reside. They are complete, self-contained, self-validating and execute automatically.

They are used for both initial checkout and subsequent regression testing of target program modifications. Therefore, a good knowledge domain of test procedures by the software tester will greatly influence his/her tester experience;

d) **Skills:** The aim of the software tester is to evaluate the product with respect to user requirements in order to encounter problems, and to do this, good testing skills are required (Rodriguez, 2012). Skills vary according to the types of tests to be carried out as well as the lifecycle phases. Skills include: *knowledge of testing*, *capacity to diagnose and solve problems*, *platform knowledge or application* to be tested. In situations where test automation is to be carried out, *training and experience* in the appropriate use of the test automation tools and programming skills are very important.

In the second factor, ***Conation*** (How do Testers see the Value of their Contribution), there are several sub-factors within the testing environment that can offer value or morale to the software tester. These include:

a) **Alignment:** a well aligned software development team, whose subunits (such as systems development, database management, network operations and architecture planning) are coherent, integrated, congruent and harmony is essential for the ensuring that the software development unit as a whole is able to meet all the emerging strategic responsibilities (Dhaliwal et al., 2011). A large portion of the budget of the software development company is spent on software development and testing, which reflects the importance of these activities. Therefore, alignment of these two subunits is crucial for the success of the IT unit. In general, the relationship between the development and testing subunits stands out as historically exhibiting substantial tension between these two groups (Cohen et al., 2004), and disrupting relations tend to exist among these internal subunits. Both groups play an important role and their cooperation is important in developing software solutions for the company to meet business strategies, and while developers create software functionalities, testers point out functionality flaws in the developers' work. These different roles lead to antagonistic relationships and animosity between the members of these subunits. Therefore, a good alignment of the tester with the development team will imply compatibility with proposed team members, which in turn will lead to valorization of their contribution within the software development team and hence lead to a good tester experience;

b) **Intention:** retaining software developers has been a problem in many organizations for decades (Donahue et al., 2006). When they quit, they depart with critical knowledge of business processes and systems that are essential for maintaining a competitive advantage. Therefore, job satisfaction is directly related to developer's intentions in general and therefore the tester's intentions in particular. In order words, testers' intentions are directly influenced by their perceptions of usefulness, social pressure, compatibility and organizational mandate. A positive attitude of the tester will also influence his intentions;

c) **Motivation:** a *dedicated individual* or a *dedicated and skill team* is one that is motivated and committed (Fagerholm et al., 2015). A tester or the testing team can be motivated when the software being developed is meaningful and should actually make a difference. Furthermore, a project that adds value to the team or one that is challenging but achieving will contribute to motivating the tester and the testing team. Also, an individual or the team is motivated when they are recognized for their efforts. Most people crave for recognition of their abilities and their efforts, hence having a greater impact on morale and motivation. These aspects will make the tester see the value of his contribution and therefore lead to a good tester experience;

d) **Commitment:** this is related to team playing capability, attention to details and thoroughness. Team playing capability is very important when selecting the right technical skill to get some work done. Therefore, some abilities of the tester to contribute to a positive ***team playing capability*** include: reliability, constructive communication, active listening, active participant, willingness and openness to share, cooperation, flexible, team commitment, problem solver and respectful (Dummies.com). (Hase, 2000) carried out a survey and identified the following factors as contributors to human elements of capability: competent people, working in teams, visible vision and values, ensuring learning takes place, managing the complexity of change, demonstrating the human aspects of leadership, performing as change agents, developing management talent and commitment of organizational development. ***Attention to details*** can be included as another element of capability, since all the above-mentioned elements of capability require thoroughness and accuracy when accomplishing a task through concern for all the elements involved in the task. ***Thoroughness***, as defined by the Miriam-Webster's Dictionary (Merriam-Webster), is "the ability to include every possible part or detail, or being careful about something in an accurate or exact way." This is a very important aspect of the software tester. Any detail omitted in his tests can cause severe impacts on the software product. Therefore, the software tester is supposed to execute every test case without negligence or omission. Therefore, the software tester's commitment will prove the value of his contribution and therefore his/her experience as a tester;

e) Goals and Plans: every software testing team or unit needs to understand its role. Testing managers and team leaders need to develop a clear and communicated purpose that is both compelling and that makes members feel important (Hyman, 1993). Employees are more likely to be committed to the purpose of the team if they are involved in creating it. Furthermore, having clear operational goals lets everyone know what is expected. If these goals are communicated and measurable, employees will understand precisely what is important and what is not. Therefore, the kinds of goals or plans of the software testing team will contribute to how software testers see the value of their contribution and therefore, tester experience.

In the third factor, ***Affection*** (How do Testers feel about their Work), we look at the relationship between individuals in an organization. Personalities, ways of working and cultural background all combine to create a workspace that can be vibrant and exciting or stressful and alienating, depending on how people cope with the challenges of working together. This factor is made up of sub-factors which include:

a) **Social:** this is mainly related to the personality traits of the tester. As seen above, these personality traits include: extraversion, openness to experience, neuroticism, conscientiousness and agreeableness (Kanij et al., 2011). Each member of the software testing team falls into one of these personality factors, and therefore, will tend to affect the social interaction and work environment;
b) **Team:** working as a team is very crucial for software testing experience. As seen earlier, team playing capabilities, compatibility with proposed team members, and a dedicated and skilled team all contribute to the tester's experience. The basis of a dynamic, cohesive team is trust (Hyman, 1993). Each team member must implicitly trust teammates to do their part, keeping the team goals paramount in their actions;
c) **Respect, Attachment and Belonging:** a high-performing software testing team has a particular identity and its members have a feeling of team spirit and pride (Fagerholm et al., 2015). Social skills, intrinsic motivation to perform, and a desire for personal development are key traits of members in such teams (Hyman, 1993). Self-motivation and dedication to the team's goals, within an environment of open communication and mutual respect, fosters the commitment necessary for success. With talent, creativity, pride, and passion, plus an environment that uses failures as foundations for successes, chaotic energy created by the team can lead toward a dynamic solution.

TX may be important in several areas of software development. For example, in software test process improvement, it could provide valuable input for analyzing and adjusting processes, procedures, techniques and platforms, as well as evaluating interpersonal, communication and experience skills as relates to the testing infrastructure. In software testing project management, it could offer means to evaluate plans and goals with respect to their alignment and intention with other testers' and developers' motivation and commitment. As relates to alignment, it evaluates compatibility with other team members; as concerns intention, it evaluates positive attitude and intelligence of the tester; as relates to motivation, it evaluates both individual and team dedication. For maintaining testing and team performance, it could offer insight into factors that affect sustainable teamwork. This involves respect, personality (social factors), attachment and belonging with respect to other team members.

4.4 Final Considerations

This chapter has presented a formal definition of TX. As we saw, the concepts were drawn from UX and DX, in addition to the literature review carried out to identify the TX factors.The concept of TX abstracted human factors that are intuitive and concrete for practitioners to help them improve their testing practices and therefore improve their testing perceptions and feelings. Given this context, these factors were grouped into three categories, as proposed by (Hilgard, 1980), which involves the classification of mental or psychological activities: cognition, conation and affection. The factors identified from the literature review, henceforth considered sub-factors, were grouped into the three main categories defined. Each of these factors and sub-factors were described. A composition of all this information has helped us define a conceptual framework for TX, which is our initial definition of the TX approach. In the next chapter, we describe the Tester Experience-Based Approach composed of guidelines, in addition to the TX factors.

CHAPTER 5 - TESTER EXPERIENCE-BASED APPROACH (TX-A)

In this section, we describe our tester experience-based approach, denoted TX-A to support the inclusion of testing practices in the software development processes.

5.1 Introduction

As a means of providing software developers with a tool to facilitate their software testing activities, we propose an approach comprising of activities and guidelines that can be applied at different levels of software testing, referred to as TX-A. TX-A can be applied by the software tester to improve his/her TX. As an illustrative example, let us consider that I am a test manager and intend to carry out testing. First, I would look at the list of factors, then sub-factors and then their respective guidelines for test manager and how they can help me in my activities. Another option would be to provide metrics for each role which would help the professional evaluate his progress and areas where he/she needs improvement. For the scope of this research, we offer guidelines. A future work would be to provide metrics in addition to the guidelines.

Given this context, (ISO/IEC/IEEE 29119-2:2013) provides a three-layer process model, which includes three main test processes: organizational test process, test management process and dynamic test process. TX-A is based on the test management and dynamic test processes of this process model to provide guidelines and activities for the different roles of a tester.

The test management process describes the management of the testing to be performed, based on risk analysis and project constraints. The process defines the types of static and dynamic tests to be performed, the overall staff, scope, time and quality of work to be done. This activity is performed by the ***test manager***, who also monitors activities during testing to ensure that it is progressing well as planned and that risks are being treated appropriately, and also take adequate measures to possible changes. In order words, the test manager ensures that *test planning*, *test monitoring and control* and *test completion* sub-processes are adequate for the success of testing.

The dynamic testing activity is made up of ***Testers*** *and* ***Test Designers***. This testing activity is carried out within a particular testing phase (unit, integration, system and acceptance) or type of testing (performance, security, and usability testing). The tester is involved in four dynamic test processes: test design and implementation, test environment setup and maintenance, test execution, and test incident reporting. The test designer is responsible for the initial identification and posterior definition of necessary tests, the monitoring of the test scope, and the evaluation of the general quality of the target tests.

Therefore, in our approach, for each Tester Experience factor defined in the previous chapter, we provide a set of activities and guidelines to be followed depending on the role of the tester (test manager, test designer or tester).

5.2 Proposal of Guidelines and Activities for the tester and test designer (Dynamic Test Process)

The same factors and sub-factors defined in section 4.3 also apply to this process. The guidelines for the tester and test designer are the same, although their activities vary. The following TX factors are related to the role of the tester and test designer as shown in Figure 5:

- Cognition – platform; technical aspects; skills; process; procedures;
- Conation – alignment; motivation; goals and plans; commitment;
- Affection – respect, attachment, and belonging; social aspects; team.

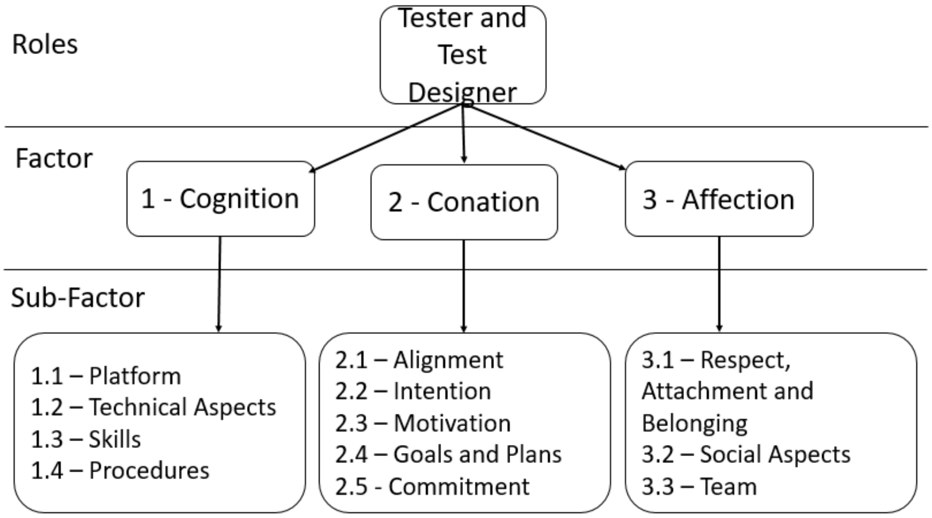

Figure 5: TX Factors and Sub-Factors Related to the Role of the Tester and Test Designer

The activities of the tester and test designer are described as follows:

Role:	**Tester**
Activities:	Identify the most appropriate implementation approach for a given test; implement tests; configure and execute tests; register the results and verify the test execution; analyze execution errors and correct them.

Role:	**Test Designer**
Activities:	Identify the target test items to be evaluated by the test effort; define the appropriate tests as well as associated test data; collect and manage test data; evaluate the results of each test cycle.

The guidelines for each of the factor “Cognition” for the role of the tester (and test designer) are described as follows:

Role:	**Tester, Test Designer**
Factor	**1 – Cognition**
Sub-Factor	**1.1 – Platform**
Guideline:	It is important for the tester and test designer to know the kind of platform to be used and how to use it; knowledge about mobile-based platforms (Android, iOS, etc) is also important. Intermediary programming is also a bonus.

Role:	**Tester or Test Designer**
Factor	**1 – Cognition**
Sub-Factor	**1.2 – Technical Aspects**
Guideline:	Tester must know how to use software testing tools, techniques or frameworks to carry out tests, especially the one(s) chosen for that particular testing activity. In addition, the tester must develop the capacity to handle complex technical aspects when the need arises.

Role:	**Tester or Test Designer**
Factor	**1 – Cognition**
Sub-Factor	**1.3 – Skills**
Guideline:	At each phase of dynamic testing, the tester must show good interpersonal and communication skills. Communication with other team members is necessary. In the case of interpersonal skills, extra training may be necessary depending on the complexity of the task; working with a more experienced tester may make the task easier.

Role:	**Tester or Test Designer**
Factor	**1 – Cognition**
Sub-Factor	**1.4 – Procedures**
Guideline:	Appropriate test case design techniques are to be used; the testing environment should be identical to the production environment in terms of hardware and software; metrics to be collected must be specified (e.g. number of specified test procedures, number of executed test procedures, total number of hours spent on execution and registration of incidents).

Role:	**Teste or Test Designer**
Factor	**2 – Conation**
Sub-Factor	**2.1 – Alignment**
Guideline:	The relationship between the tester and other team members, as well as his/alignment with the testing project is crucial will imply compatibility with team members. This is especially critical in complex projects.

Role:	**Teste or Test Designer**
Factor	**2 – Conation**
Sub-Factor	**2.2 – Intention**
Guideline:	In order to carry out the testing activities, the tester must maintain a positive attitude, and must be willing to cooperate with other team members throughout the whole process (in the case where development is not an individual effort).

Role:	Teste or Test Designer
Factor	2 – Conation
Sub-Factor	2.3 – Motivation
Guideline:	Although most of the motivation comes from the kind of project, or from the managerial and organizational processes, the individual tester has his/her own part to play: dedication and commitment. These two factors may lead to his/her recognition or promotion, or any other aspect that can make him/her more accomplished and satisfied.

Role:	Teste or Test Designer
Factor	2 – Conation
Sub-Factor	2.4 – Goals and Plans
Guideline:	During all the testing activities, the tester must make sure his/her goals and plans are aligned to those of the development team and those of the testing goals and plans. This sub-factor does not apply to individual testers.

Role:	Tester or Test Designer
Factor	2 – Conation
Sub-Factor	2.5 – Commitment
Guideline:	Reliability, constructive communication, active listening, active participation, willingness and openness to share, cooperation, flexibility, team commitment, respectful and problem solver are key characteristics of a software tester. This sub-factor does not apply to individual testers.

Role:	Tester or Test Designer
Factor	3 – Affection
Sub-Factor	3.1 – Respect, Attachment and Belonging
Guideline:	Desire for personal growth and development, mutual respect, feeling of belonging and attachment are expected from the tester. This sub-factor does not apply to individual testers.

Role:	Tester or Test Designer
Factor	3 – Affection
Sub-Factor	3.2 – Social Aspects
Guideline:	It is important for the software tester to know which personality traits he/she possesses: extravert, introvert, open to experiences, neuroticist, conscientious or agreeable. Each of these personality traits can be decisive to the performance of the testing team. This factor does not apply to individual testers.

Role:	Teste or Test Designer
Factor	3 – Affection
Sub-Factor	3.3 – Team
Guideline:	Teamwork is crucial for software testing. As for individual developers, who also need to carry out testing, this factor is not important.

5.3 Guidelines and Activities for the Test Manager (Test Management Process)

The following TX factors influence the *test manager experience*:

- Cognition – platform; technical aspects; skills; process; procedures;
- Conation – alignment; intention; motivation; goals and plans; commitment;
- Affection – respect, attachment, and belonging; social aspects; team.

The following TX factors are related to the role of the test manager as shown in Figure 6:

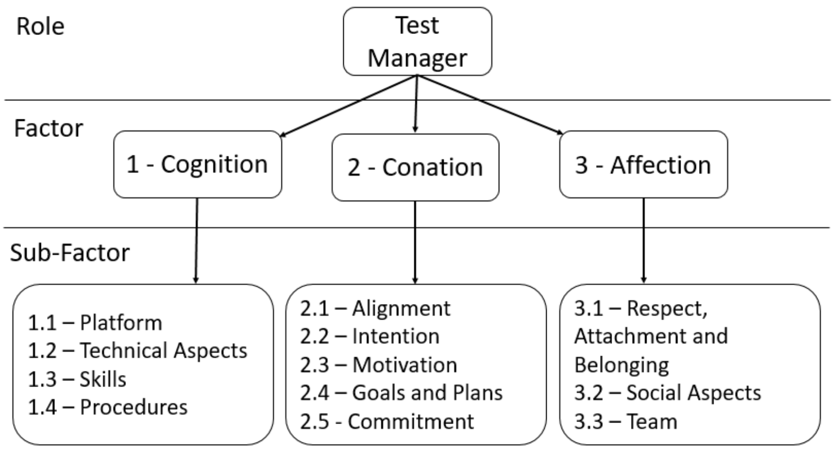

Figure 6: TX Factors and Sub-Factors Related to the Role of the Test Manager

The activities of the test manager are described as follows:

Role:	**Test Manager**
Activities:	Define the test objectives and efforts; define and plan the appropriate test resources; evaluate the progress and efficiency of the tests; define the appropriate levels of quality by correcting important defects; define an appropriate level of focus on testability during software development.

The guidelines for each of the factor “Cognition” for the role of the test manager are described as follows:

Role:	**Test Manager**
Factor	**1 - Cognition**
Sub-Factor	**1.1 – Platform**
Guideline:	The test manager should determine the type of platform to be used during testing, and is mandatory for him/her to have a good mastery of the platform to be used. In the case of a new type of platform, the test manager should offer training to the testers.

Role:	**Test Manager**
Factor	**1 - Cognition**
Sub-Factor	**1.2 – Technical Aspects**
Guideline:	The test manager must know how to use software testing tools, techniques or frameworks to carry out tests, and decide which are best adequate for the project in context. In addition, the test manager must develop the capacity to handle complex technical aspects when the need arises.

Role:	**Test Manager**
Factor	**1 - Cognition**
Sub-Factor	**1.3 – Skills**
Guideline:	At each phase of dynamic testing, the test manager must show good interpersonal and communication skills. Good communication with the testers is necessary. It is the function of the test manager to assign skilled testers for particular testing tasks in order to obtain the expected results. In the case of unskilled testers, the test manager must make sure they receive the appropriate training needed to be become skilled.

Role:	**Test Manager**
Factor	**1 - Cognition**
Sub-Factor	**1.4 – Procedures**
Guideline:	Appropriate test case design techniques are to be determined by the test manager; he/she should make sure the testing environment is identical to the production environment in terms of hardware and software; metrics to be collected must be specified (e.g. number of specified test procedures, number of executed test procedures, total number of hours spent on execution and registration of incidents).

Role:	**Test Manager**
Factor	**2 - Conation**
Sub-Factor	**2.1 – Alignment**
Guideline:	The test manager must make sure the testing team is well aligned to the testing project, since alignment is crucial for the success of testing. He/she is also in charge of making sure the testing team is aligned with the developing team, hence strengthening cooperation between the two entities.

Role:	**Test Manager**
Factor	**2 - Conation**
Sub-Factor	**2.2 – Intention**
Guideline:	Since retaining software developers has been a problem in many organizations, the test manager is to make sure each tester is satisfied with his/her task, since job satisfaction is directly related to the developer's intentions. Intentions are directly related by the testers' perceptions of usefulness, social pressure and compatibility.

Role:	**Test Manager**
Factor	**2 - Conation**
Sub-Factor	**2.3 – Motivation**
Guideline:	The test manager should ensure a motivated testing team, and also make sure the testing project adds more value to the team, including different challenges. Recognition for their work is also important for motivating testers. Furthermore, the testers should be given the opportunity to innovate, and perhaps flexible working hours, and offering empowerment to the testers in terms of decision making.

Role:	**Test Manager**
Factor	**2 - Conation**
Sub-Factor	**2.4 – Goals and Plans**
Guideline:	The test manager should ensure that the testing project's goals and those of individual testers are all aligned in order to obtain the maximum participation of each tester. Clear operational goals should be defined, measurable and understandable by each tester. Furthermore, it is advisable for the testers to be involved in decision making so that the goals and plans should be clearly understood by everyone.

Role:	**Test Manager**
Factor	**2 - Conation**
Sub-Factor	**2.5 – Commitment**
Guideline:	It is important for the test manager to ensure reliability, constructive communication, active listening, active participation, willingness and openness to share, cooperation, flexibility, team commitment, respectfulness of each member of the testing team.

Role:	**Test Manager**
Factor	**3 - Affection**
Sub-Factor	**3.1 – Respect, Attachment and Belonging**
Guideline:	The test manager should ensure that each team member has the feeling of team spirit, belonging and pride. He/she should also encourage the feeling of harmony and equality among the team members.

Role:	**Test Manager**
Factor	**3 - Affection**
Sub-Factor	**3.2 – Social Aspects**
Guideline:	It is the role of the test manager to know which personality traits each tester possesses: extravert, introvert, open to experiences, neuroticist, conscientious or agreeable. This is important for assigning tasks to testers based on their personality traits in order to achieve the testing goals.

Role:	**Test Manager**
Factor	**3 - Affection**
Sub-Factor	**3.2 – Team**
Guideline:	The test manager should ensure the testers work as a team in order to achieve the testing goals. In addition, the team must be skilled and dedicated; there must be trust among members of the team.

5.4 *Final Considerations*

In this chapter, we have proposed TX-A, which is composed of activities and guidelines for each role of the tester, test designer and test manager. Since this proposal is just an initial step, we expect the activities and guidelines to be improved through experimental research, so that they can adequately serve their purpose of improving the TX with respect to testing infrastructure (testing tools, programming languages, libraries, platforms, frameworks, processes and methods), feelings about work (respect, attachment and belonging) and value of their contribution (alignment of their goals with those of the project, plans, intentions and commitment), as we defined in our objectives.

In the next chapter, we present the results of a survey conducted with software development professionals in order to obtain feedback about the proposed TX-A approach.

CHAPTER 6 - SURVEY WITH SOFTWARE DEVELOPMENT PRACTITIONERS WITH RESPECT TO THE TX-A APPROACH

In this chapter, we present the results of a survey carried out with software development practitioners, including developers, testers, test managers, project managers, professors and researchers, to evaluate the TX Guideline-Based approach proposed in the previous chapter.

6.1 Introduction

As part of the evaluation of the TX Guideline-based approach, it is necessary to obtain practitioners' opinions about the importance of the factors and relevance of the guidelines in improving the TX. For a recap, as already described in Chapter 4, the TX factors and respective sub-factors are: *Cognition* – platform, technical aspects, processes, procedures, and skills; *Conation* – alignment, intention, motivation, commitment and goals and plans; *Affection* – social factors, respect, attachment and belonging, and team. To obtain practitioners' opinions, we apply the survey method. From the results obtained, we intend to provide TX recommendations for each practitioner category (developer, tester/test designer or test/project manager). This is because the needs and priorities of the developer, for instance, are different from those of the tester and test manager, so, the recommendations on how to use our approach will be different for these professionals.

The remainder of this chapter is organized as follows: in section 6.2, we present the planning of this study; in section 6.3, we present the study execution; in section 6.4, we present the analysis; the discussion is presented in section 6.5; the threats to validity of this study are presented in section 6.6 while the conclusion and future works are presented in section 6.7.

6.2 Planning of Study

6.2.1 Goals

The objective of this survey is to identify the differences in software development practitioners' perception of TX. Our goals are therefore as follows:

- To **analyze** the factors affecting TX **with the aim of** characterizing them **with respect to** their importance, **from the point of view** of developers, testers, test designers, test and project managers, as well as software development/testing researchers and professors, **in the context of** software testing activities;

- To **analyze** the guidelines of the corresponding factors affecting TX **with the aim of** characterizing them **with respect to** their relevance, **from the point of view** of developers, testers, test designers, test and project managers, as well as software development/testing researchers and professors **in the context of** software testing activities.

Given this context, two research questions are defined:

- **RQ1**: Are the factors affecting TX important in improving the software development practitioner's tester experience?

  Metric:

  List of TX factors important to improving TX per profile (developer, tester/test designer, test/project manager, researcher and professor);

- **RQ2**: Are the guidelines of their corresponding factors affecting TX relevant in improving the software development practitioner's tester experience?

  Metric:

  List of guidelines relevant to guiding the software development professional to improving his/her TX per profile;

It is important to note that for both research questions, we are not interested in the order of importance of the factors or relevance of the guidelines; we just want to determine which factors are important or which guidelines are relevant in improving TX. In the context of importance or relevance of a factor or guideline, any factor in which more than 50% of the software development professionals agree or strongly agree as being important for improving TX is considered important. This also applies to the guidelines. This is explained in details in section 6.2.3.

6.2.2 Target Population

Three main categories were defined as target population:

- Industrial and academic software developers, testers, test designers;
- Industrial software development project managers and test managers;
- Researchers on, and/or professors of, software development/testing.

These three categories were defined because they englobe all the different software development profiles, ranging from academic to industrial, from developers to project managers, and from testers to test managers.

6.2.3 Measuring Instrument

In order to carry out this study, the chosen method was a survey, by the use of a questionnaire. This questionnaire can be found in the following link: https://goo.gl/forms/U2DkC2e4F6iFNvMZ2 or in Appendix 1, and was divided into two main sections: one section for developers/testers/test designers and the other section for test/project managers and researchers/professors. There was an option "other", for those who did not fall into any of the defined categories. In this case, they were directed to the "test/project managers" section. We believe the results would not be influenced by the profile. A survey was chosen because we wanted to obtain the general opinion and perception of the general software development population. This survey was carried out in August 2017.

We used a 5-point Likert scale, ranging from: 1 – Strongly Disagree (SD), 2 – Disagree (D), 3 – Neither Agree nor Disagree (N), 4 – Agree (A), and 5 – Strong Agree (SA). In addition, a subjective, non-mandatory question was included to each objective question, asking the respondents if they have any suggestions for improvement.

For each factor, sub-factor or guideline, it is possible to identify whether it is important or relevant respectively. "Important" is defined by (dictionary.com) as "entitled to more than ordinary consideration or notice), and "relevant" as "bearing upon or connected with the matter in hand; pertinent". For this analysis, we observe the categories that obtained the highest percentage rating by professionals for each factor and guideline. Based on this rating, we are then able to determine whether the factor or guideline is important or relevant, respectively for improving TX as follows (Figure 7):

- Agreement: in cases where more than 50% of the professionals' total number of ratings are "agree" and/or "strongly agree", the factor or guideline will be considered as respectively important or relevant in improving TX;
- Tendency for agreement: in cases where more than 50% the professionals' total number of ratings are "neutral" and "agree" and/or "strongly agree", the factors or guidelines have a tendency for professionals to agree that they are respectively important or relevant in improving TX;
- Neutral: more than 50% of the professionals' total number of ratings are "neutral", the professionals are said to present mixed feelings or opinions about the factor or guideline;
- Disagreement: in cases where more than 50% of the professionals' total number of ratings are "disagree" and/or "strongly disagree", the factor or guideline will be considered as respectively not important or not relevant in improving TX;

- Tendency for disagreement: in cases where more than 50% the professionals' total number of ratings are "neutral" and "disagree" and/or "strongly disagree", the factors or guidelines have a tendency for professionals to disagree that they are respectively important or relevant in improving TX;
- No consensus: when the opinions are distributed among the categories, the factor or guideline will be evaluated as having no consensus in determining whether it is respectively important or relevant.

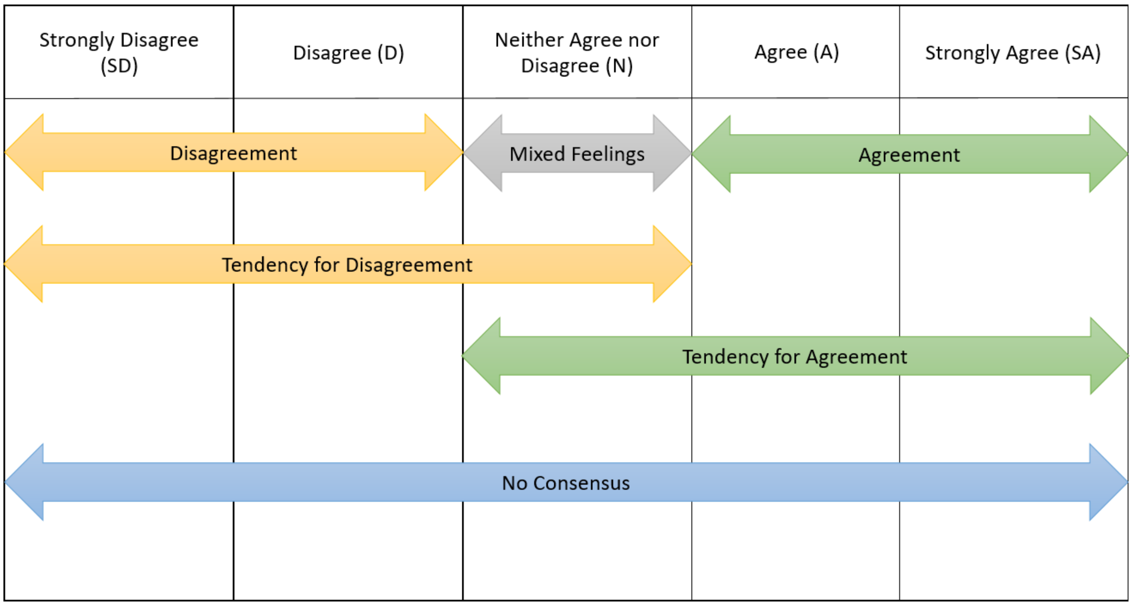

Figure 7: Interpretation of Results

These analyses are explained in more details in the subsequent sub-sections.

6.3 Execution of Study

The questionnaire was distributed among developers, testers, test managers, project managers, professors and researchers, among others, as shown in Figure 8 below. In total, the questionnaire was sent to a total of 566 professionals among which include: LinkedIn (45), professionals who took the software testing certification examination (160), Google Experts (237), conferences or workshops on software testing (59), Twitter and ResearchGate (40), other sources (25). From the 566, we obtained 46 responses. After running the survey, we obtained a sample confidence level of approximately 86% using the following formula (Gardener & Altman, D. J., 1989):

$$n = \frac{N.\frac{1}{{E_0}^2}}{N+\frac{1}{{E_0}^2}}, \text{ where:}$$

N = population size,
n = sample size,

E_0= Confidence level (e.g. 0.05 → 95%)

We believe the confidence level would have been higher if we were able to count the real number of researchers who received and read the survey.

As shown in Figure 8 below, among all the 46 respondents, 20 (43%) are developers, 12 (26%) are testers, 5 (11%) are project managers, 4 (9%) are test managers, and 5 (11%) are researchers and professors. With respect to software development, among the 46 respondents, 3 (6.5%) have never developed software before (but are currently software testers in their respective organizations, so were included in our analysis), 17 (37%) have developed software in individual projects, 29 (63%) have developed software as part of an academic team, and 25 (54.3%) have developed software as part of a team in an industry.

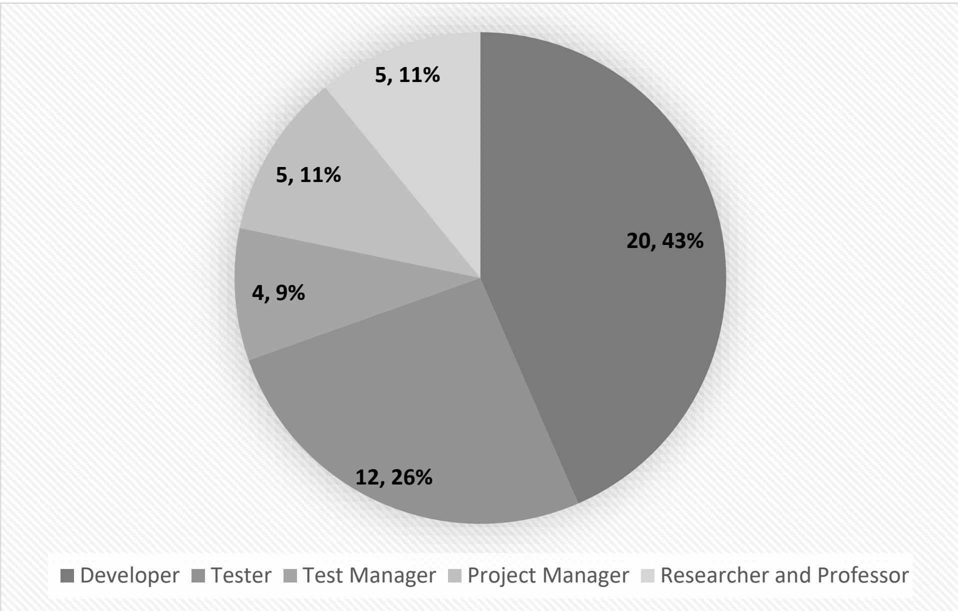

Figure 8: Number of Respondents per Profile

With respect to software testing, among the 46 respondents, 4 (6%) have never tested software before, 11 (18%) have tested software in individual projects, 18 (29.1%) have tested software as part of an academic team, and 29 (47%) have tested software as part of a team in an industry (Figure 9).

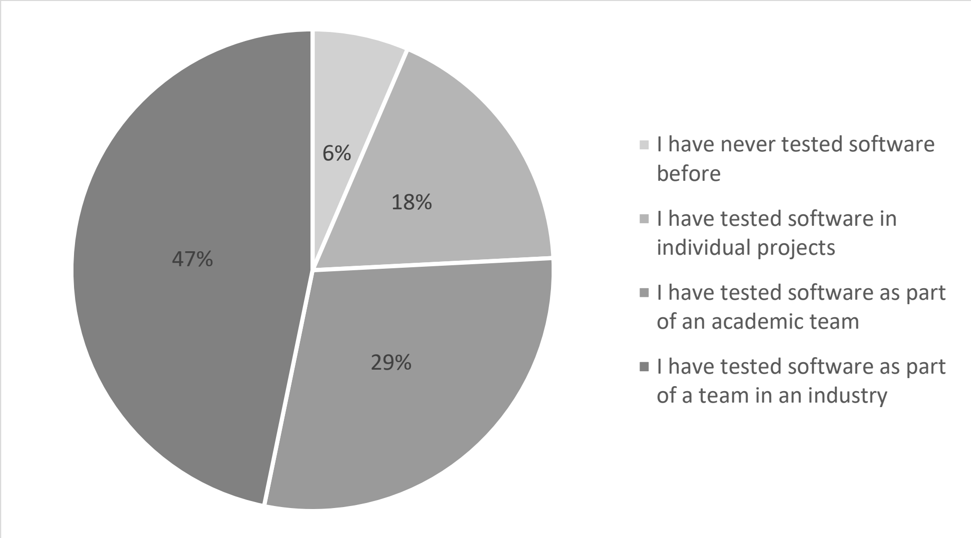

Figure 9: Experience with Software Testing

With respect to geographical distribution, 40 respondents (87%) are from Brazil, 2 (4.3%) are from Canada, 2 (4.3%) are from Spain, 1 (2.2%) from Sweden, and 1 (2.2%) from the United Kingdom.

6.4 Results Analysis

In this section, we present an in-depth analysis of the results based on the two research questions defined, for the factors and their corresponding guidelines.

6.4.1 RQ 1: Are the factors affecting TX important in improving the software development practitioner's tester experience?

The following tables present the ratings (in percentage) of each category (strongly disagree to strongly agree) of the platform sub-factor for each participant profile, presented on a three-color heat map scale, ranging from light-blue (0%), through grey to dark-grey (100%).

Tables 1 to 5 present participants' ratings for the cognitive sub-factors (platform, technical aspects, processes, procedures and skills). Table 2 below shows participants' ratings of the "platform" sub-factor.

Table 2: Participants' ratings (in percentage) of the "Platform" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	15	50	25	Agreement
Tester	0	8	25	67	0	Agreement
Test Manager	0	25	0	25	50	Agreement
Project Manager	0	20	40	20	20	Tendency for agreement
Researcher and Professor	0	20	0	20	60	Agreement

From Table 2 above, we can observe that all professionals present a degree of agreement that "platform" is important for improving TX. There is a tendency for agreement for project managers, where 40% neither agree nor disagree, 20% agree and 20% strongly agree. We also observe a degree of discordance among test managers, project managers, and researchers and professors, with respectively 25%, 20% and 20%. Overall, all professionals agree that "platform" is important for improving TX.

Table 3 below shows participants' ratings of the "technical aspects" sub-factor.

Table 3: Participants' ratings (in percentage) of the "Technical Aspects" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	0	50	40	Agreement
Tester	0	8	8	67	17	Agreement
Test Manager	0	0	0	50	50	Agreement
Project Manager	0	0	0	100	0	Agreement
Researcher and Professor	0	0	0	60	40	Agreement

As can be observed in Table 3, most participants' ratings are concentrated between "agree" and "strong agree". There was a very low discordance for developers (5% strongly disagree and 5 % disagree) and testers (8%). All test managers, project managers and researchers and professors agree or strongly agree that "technical aspects" is important for improving TX. Overall, all participants agree that this sub-factor is important.

As shown in Table 4 below, all project managers and researchers and professors agree that "technical aspects" is important for improving TX. There is also a general agreement among developers and testers, even though over 30% of developers either disagree or neither agree nor disagree. There is no consensus for test managers, as 50% disagree while 50% agree. A future work will therefore to perform another survey with more test managers to help arrive at a consensus.

Table 4: Participants' ratings (in percentage) of the "Processes" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	0	15	15	40	30	Agreement
Tester	0	8	8	75	9	Agreement
Test Manager	0	50	0	50	0	No consensus
Project Manager	0	0	0	80	20	Agreement
Researcher and Professor	0	0	0	60	40	Agreement

Table 5 below shows participants' ratings for "procedures". All professionals agree of the sub-factor being important. 25% of testers however, disagree. We can also observe that all project managers and researchers and professors agree or strongly agree.

Table 5: Participants' ratings (in percentage) of the "Procedures" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	0	45	45	Agreement
Tester	0	25	0	25	50	Agreement
Test Manager	0	0	25	25	50	Agreement
Project Manager	0	0	0	80	20	Agreement
Researcher and Professor	0	0	0	60	40	Agreement

As shown in Table 6 below, over 17% of testers disagree that "skills" is important for improving TX. 100% of test managers and researchers and professors strongly agree.

Table 6: Participants' rating (in percentage) of the "Skills" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	20	25	45	Agreement
Tester	0	17	0	50	33	Agreement
Test Manager	0	0	0	0	100	Agreement
Project Manager	0	0	0	40	60	Agreement
Researcher and Professor	0	0	0	0	100	Agreement

Overall, cognitive sub-factors presented in the tables above, present an overall agreement of their importance to improving TX. Only test managers presented no consensus for “processes”, while project managers show a tendency for agreement for the “platform” sub-factor. In addition, we can observe overall higher ratings for test managers, project managers, and researchers and professors than for developers and testers. “Platform” obtained the overall highest disagreement among all professionals. We can therefore conclude from these observations that in all, Cognition is important for improving TX, with its importance more prominent among project managers, test managers and researchers and professors.

Tables 6 to 10 below present the evaluations for the conative factor and its sub-factors (alignment, intention, motivation, commitment, goals and plans)

As shown in Table 7 below, it can be observed that, even though 60% of researchers and professors agree that “alignment” is important for improving TX, 40%, however, neither agree nor disagree. We also see some discrepancies among developers and testers. On the other hand, we observe that all project and test managers agree or strongly agree to this factor. Overall, all professionals agree to “alignment” being important for improving TX.

Table 7: Participants' rating (in percentage) of the "Alignment" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	5	45	40	Agreement
Tester	0	0	17	50	33	Agreement
Test Manager	0	0	0	75	25	Agreement
Project Manager	0	0	0	40	60	Agreement
Researcher and Professor	0	0	40	0	60	Agreement

As shown in Table 8, it is observed that there is a certain degree of all participant profiles (except test managers) neither agreeing nor disagreeing that “intention” is important for improving TX (between 15% and 20%). Despite this, more than 70% of all participants agree to “intention” being important.

Table 8: Participants' rating (in percentage) of the "Intention" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	15	50	25	Agreement
Tester	0	8	17	58	17	Agreement
Test Manager	0	0	0	50	50	Agreement
Project Manager	0	0	20	40	40	Agreement
Researcher and Professor	0	0	20	40	40	Agreement

Different from “intention”, “motivation” presents an improvement in participants' responses: all test managers, project managers and researchers and professors agree and/or strongly agree of this sub-factor being important for improving TX (100% in total). Developers and testers show a slight disagreement (Table 9). Over 75% of all professionals agree that “motivation” is important.

Table 9: Participants' rating (in percentage) of the "Motivation" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	15	25	50	Agreement
Tester	0	8	9	50	33	Agreement
Test Manager	0	0	0	50	50	Agreement
Project Manager	0	0	0	20	80	Agreement
Researcher and Professor	0	0	0	20	80	Agreement

From Table 10, we can observe some similarities in the number of developers and testers who disagree that “commitment” is important for improving TX. We also observe 100% agreement among test managers, project managers, and researchers and professors.

Table 10: Participants' rating (in percentage) of the "Commitment" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	10	35	45	Agreement
Tester	0	8	8	50	34	Agreement
Test Manager	0	0	0	25	75	Agreement
Project Manager	0	0	0	60	40	Agreement
Researcher and Professor	0	0	0	60	40	Agreement

As shown in Table 11, over 17% of testers disagree that “goals and plans” is important for improving TX. Again, as observed in “intention” and “commitment”, all test manages, project managers and researchers and professors agree or strongly agree to this sub-factor being important for improving TX. Over 75% of developers and testers agree.

Table 11: Participants' rating (in percentage) of the "Goals and Plans" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	10	25	55	Agreement
Tester	0	17	0	33	50	Agreement
Test Manager	0	0	0	25	75	Agreement
Project Manager	0	0	0	60	40	Agreement
Researcher and Professor	0	0	0	40	60	Agreement

In summary, as observed from the evaluations of the conative sub-factors, we can conclude that there is a general agreement about the sub-factors, especially among test managers, project managers, and researchers and professors. We also observe that 5% of developers strongly disagreed and 5% disagreed for all sub-factors. Notwithstanding, conation is rated as important for improving TX by all professional profiles.

Tables 12 to 14 below present the analyses of the affection factor and its sub-factors. As observed in Table 12, there is some discrepancy among developers and testers with respect to “social factors”, where we see 5% and 9% of developers and testers strongly disagreeing, and 10% and 25% disagreeing respectively. We also observe 30% of developers and 16% of testers neither agreeing nor disagreeing. However, 55% of developers and 50% of testers either agree or strongly agree that this sub-factor is important, therefore presenting a slight tendency for agreement among these professionals. As for test managers, project managers, and researchers and professors, all of them (100%) either agree and/or strongly agree.

Table 12: Participants' rating (in percentage) of the "Social Factors" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	10	30	40	15	Tendency for agreement
Tester	9	25	16	25	25	Tendency for agreement
Test Manager	0	0	0	50	50	Agreement
Project Manager	0	0	0	100	0	Agreement
Researcher and Professor	0	0	0	80	20	Agreement

With respect to “respect, attachment and belonging” (Table 13), some degree of disagreement can be seen for developers and testers. Overall, over 65% of these professionals consider this sub-factor as important for improving TX, as well as 100% of test managers, project managers, and researchers and professors.

Table 13: Participants' rating (in percentage) of the "Respect, Attachment and Belonging" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	15	45	30	Agreement
Tester	0	17	16	42	25	Agreement
Test Manager	0	0	0	50	50	Agreement
Project Manager	0	0	0	40	60	Agreement
Researcher and Professor	0	0	0	60	40	Agreement

As for team Table 14, we observe similar results to “respect, attachment and belonging”, with more than 70% of all professionals agreeing that “team” is important for improving TX.

Table 14: Participants' rating (in percentage) of the "Team" sub-factor

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	15	30	45	Agreement
Tester	0	17	0	25	58	Agreement
Test Manager	0	0	0	25	75	Agreement
Project Manager	0	0	0	0	100	Agreement
Researcher and Professor	0	0	0	80	20	Agreement

For developers who strongly disagreed about most factors being important for improving TX, he/she gave his/her reasons for the choice: “*I do not agree with your roles. For instance, test manager is a role which should be extinct from the context of testing. Such role is only one of the attributes of a tester, since in order to excel at their tasks, testers should have no barriers between them and the client. Having a test manager increases the bureaucracy and minimizes the potential of testers*.” These roles were defined by

(ISO/IEC/IEEE 29119-2:2013) and therefore we consider them valid, even if they are not applied in all software companies.

As a suggestion for improvement, a project manager suggested "*continuous learning*" as a sub-factor to be included: "*Continuous learning is an important factor*". We however, believe that this can be included under the "*skills*" sub-factor, since continuous learning that lead to an improvement in the professional's skills.

Another professional, a test manager, made the following comment about the factors: "*In my view (with an "industrial look"), the tester/test analyst in such a primary role, was left behind. We need to demystify that the tester is only looking for failures, it is necessary a definite awareness that this profile is an active member of development, therefore, there is a previous work for the contribution of the development with quality, so that there are the minimum of possible failures. The company has a role interested in this scenario, it must provide an environment that allows professionals to develop their skills. Thanks :)*" One possible reason for this comment was that, due to the division of the questionnaire into two main sections, the respondents could only respond to the section based on his/her profile (in this case test/project manager), and therefore, could not see the other section (testers/test designers/developers). Not withstanding, we could highlight three important points from this comment: tester's role in identifying failures, tester's role in participating in decision making, and testing as an environment for skills development. The tester's role in identifying failures was the focus of this work, while an environment for skills development was discussed partly as a means of continuous learning in order to improve skills.This point can also be investigated in more details in future works. The last point (tester's role in decision making) was not discussed in this research, and therefore serves as a good opportunity for future works.

With respect to the three factors, cognition, conation, and affection, we can conclude that they are important for improving TX, given that their respective sub-factors are also important for improving TX. The "social factors" sub-factor, of the "conation" sub-factor may or may not be important for improving developers' or testers' TX. Figure 10 Summarizes the ranking of the factors per professional. As observed, developers and testers ranked conation as the most important factor, followed by cognition and finally affection. This implies that these professionals consider the value of their contribution as most important for improving their TX, that is, alignment with their activities and those of the team, their intentions, motivations, commitment, and goals and plans. Meanwhile, test and project managers consider affection as the most important factor for improving their TX. This makes sense as they are incharge of building and maintaining good teams that will achieve the objectives of their tasks: the personality traits of each team member, respect, attachment and belonging of every member, and finally team spirit. Finally, researchers and professors considered cognition as the most important factor. This is true because the job of professors is to educate students on how to use tools, techniques, processes, methodologies, and therefore their primary focus is on cognitive factors.

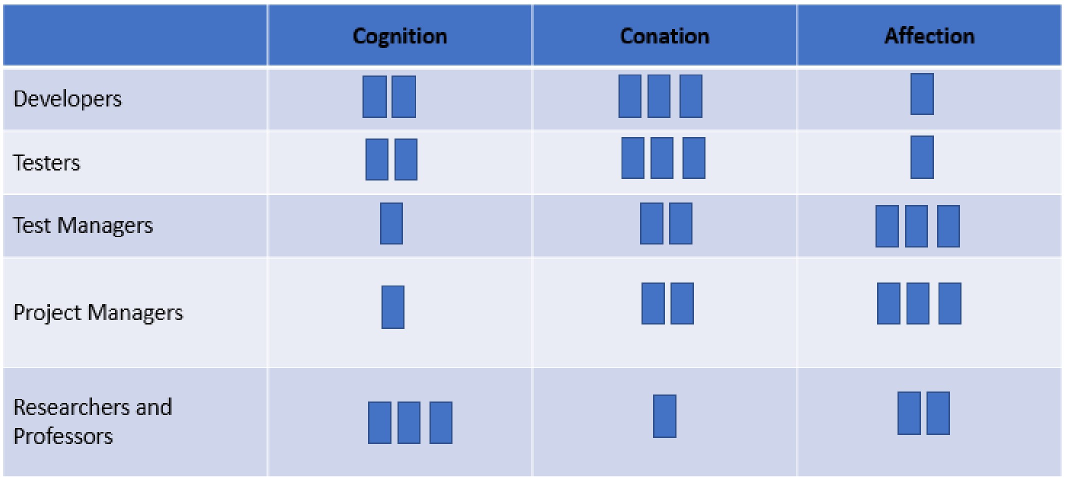

Figure 10: Ranking of factors per profile

In summary, all of the factors (except one – processes for test managers) were rated by professionals as being important for improving TX. We could also perceive all affection and most conative sub-factors obtaining 100% agreement by test managers, project managers, and researchers and professors, and over 70% for developers and testers. We believe one possible reason some professionals disagreed or neither agreed nor disagreed was due to the fact that they might not have completely understood the factors or their functioning. For this reason, we provided guidelines, which were also evaluated by them. The analysis is presented in the next sub-section.

6.4.2 RQ2: Are the Guidelines of the Corresponding TX Factors Relevant in Improving the Software Development Practitioner's Tester Experience?

Apart from the factors affecting TX, their corresponding guidelines were also evaluated in the survey. The practitioners were offered the list of guidelines and asked to rate them according to their relevance. The results obtained are described as follows.

Table 15 presents how participants evaluated the "platform" guideline. Different from the sub-factor's rating in the previous section, we observe an improvement: there is no agreement among professionals, except for developers. However, there is no significant different among those who neither agreed nor disagreed. But we find that there is an overall 80% or above agreement for test managers (100%), project managers (80%) and researchers and professors (80%). There continues to be a general agreement for developers and testers.

Table 15: Participants' rating (in percentage) of the "Platform" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	25	55	10	Agreement
Tester	0	0	33	42	25	Agreement
Test Manager	0	0	0	50	50	Agreement
Project Manager	0	0	20	80	0	Agreement
Researcher and Professor	0	0	20	20	60	Agreement

In Table 16 below, we observe 100% agreement for test managers, project managers, and researchers and professors for "technical aspects" guideline, and some discordance for developers (5%) and testers (17%). The results are not significantly different for the sub-factor's ratings in the previous section.

Table 16: Participants' rating (in percentage) of the "Technical Aspects" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	25	55	10	Agreement
Tester	0	17	8	42	33	Agreement
Test Manager	0	0	0	50	50	Agreement
Project Manager	0	0	0	60	40	Agreement
Researcher and Professor	0	0	0	20	80	Agreement

From Table 17 below, it is important to recall that there was no consensus for the "processes" sub-factor for test managers, with 50% disagreement. As observed from its guideline analysis, this is no longer the case. There is some degree of discordance among testers (17%) and test managers (25%).

Table 17: Participants' rating (in percentage) of the "Processes" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	0	10	55	30	Agreement
Tester	0	17	0	58	25	Agreement
Test Manager	0	25	0	50	25	Agreement
Project Manager	0	0	0	0	100	Agreement
Researcher and Professor	0	0	0	20	80	Agreement

From Table 18, we observe the same percentage of discordance for developers and testers, as observed in the previous guidelines. Overall, "procedures" guideline is rated as relevant by professionals for improving TX.

Table 18: Participants' rating (in percentage) of the "Procedures" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	10	45	35	Agreement
Tester	0	17	0	58	25	Agreement
Test Manager	0	0	0	60	40	Agreement
Project Manager	0	0	0	0	100	Agreement
Researcher and Professor	0	0	0	20	80	Agreement

In Table 19, we observe significantly similar results of the "skills" guidelines to "procedures" above, except for 8% of testers strongly disagreeing.

Table 19: Participants' rating (in percentage) of the "Skills" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	10	40	40	Agreement
Tester	8	0	17	33	42	Agreement
Test Manager	0	0	0	25	75	Agreement
Project Manager	0	0	0	60	40	Agreement
Researcher and Professor	0	0	0	20	80	Agreement

In Table 20, we observe similar results obtained above, with a small percentage of developers and testers disagreeing and all test managers, project managers and researchers and professors agreeing.

Table 20: Participants' rating (in percentage) of the "Alignment" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	10	40	40	Agreement
Tester	8	0	17	33	42	Agreement
Test Manager	0	0	0	25	75	Agreement
Project Manager	0	0	0	60	40	Agreement
Researcher and Professor	0	0	0	20	80	Agreement

The results for the "intention" guideline are similar to those above, as shown in Table 21, with some degree of improvement as compared to the rating of its sub-factor.

Table 21: Participants' rating (in percentage) of the "Intention" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	10	45	35	Agreement
Tester	0	8	8	59	25	Agreement
Test Manager	0	0	0	80	20	Agreement
Project Manager	0	0	0	60	40	Agreement
Researcher and Professor	0	0	0	60	40	Agreement

As shown in Table 22, the results are similar to those above, with the same 5% of developers strongly disagreeing, 5% disagreeing, and 17% of testers disagreeing. The difference between this sub-factor's ("motivation") guideline and those above is the ratings for researchers and professors, where 40% neither agree nor disagree, and therefore a drop in the overall ratings as compared to those of its sub-factor (40% agreement and 60% strong agreement).

Table 22: Participants' rating (in percentage) of the "Motivation" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	10	35	45	Agreement
Tester	0	17	8	33	42	Agreement
Test Manager	0	0	0	25	75	Agreement
Project Manager	0	0	0	20	80	Agreement
Researcher and Professor	0	0	40	0	60	Agreement

The results obtained for "commitment" as shown in Table 23 are significantly similar to the other sub-factor guidelines shown above (except for motivation).

Table 23: Participants' rating (in percentage) of the "Commitment" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	5	40	45	Agreement
Tester	0	17	0	50	33	Agreement
Test Manager	0	0	0	25	75	Agreement
Project Manager	0	0	0	0	100	Agreement
Researcher and Professor	0	0	0	60	40	Agreement

The results obtained for the “skills” guideline, as shown in Table 24 present similar results to those above, with a small percentage of developers and testers disagreeing and 100% of test managers, project managers, and researchers and professors agreeing.

Table 24: Participants' rating (in percentage) of the "Goals and Plans" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	0	30	60	Agreement
Tester	0	16	0	42	42	Agreement
Test Manager	0	0	0	25	75	Agreement
Project Manager	0	0	0	40	60	Agreement
Researcher and Professor	0	0	0	60	40	Agreement

As shown in Table 25, different from the results obtained from “sub-factor ratings for developers and testers, described in the previous section, where the overall evaluation was a tendency for agreement, the results are a bit different for the guidelines: we observe over 65% of developers and 58% of testers either agreeing or strongly agreeing that the guideline is relevant (as opposed to 55% and 50% respectively).

Table 25: Participants' rating (in percentage) of the "Social Factors" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	25	50	15	Agreement
Tester	0	25	17	33	25	Agreement
Test Manager	0	0	0	75	25	Agreement
Project Manager	0	0	0	40	60	Agreement
Researcher and Professor	0	0	0	20	80	Agreement

As shown in Table 26, there is no significant difference between the “respect, attachment and belonging” guideline evaluation and that of its sub-factor.

Table 26: Participants' rating (in percentage) of the "Respect, Attachment and Belonging" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	10	35	45	Agreement
Tester	0	17	8	33	42	Agreement
Test Manager	0	0	0	50	50	Agreement
Project Manager	0	0	0	20	80	Agreement
Researcher and Professor	0	0	0	40	60	Agreement

From Table 27, we observe similar results to other sub-factor guidelines for the “team” guideline, with over 80% of all professionals agreeing to this guideline being relevant for improving TX.

Table 27: Participants' rating (in percentage) of the "Team" guideline

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	10	30	50	Agreement
Tester	0	16	0	9	75	Agreement
Test Manager	0	0	0	50	50	Agreement
Project Manager	0	0	0	0	100	Agreement
Researcher and Professor	0	0	0	40	60	Agreement

As shown in the tables above (Table 15 to Table 27), there is a general agreement by participants that all guidelines are relevant for improving TX. However, as observed in

Profile	SD (%)	D (%)	N (%)	A (%)	SA (%)	Analysis
Developer	5	5	0	30	60	Agreement
Tester	0	16	0	42	42	Agreement
Test Manager	0	0	0	25	75	Agreement
Project Manager	0	0	0	40	60	Agreement
Researcher and Professor	0	0	0	60	40	Agreement

As shown in Table 25, different from the results obtained from “sub-factor ratings for developers and testers, described in the previous section, where the overall evaluation was a tendency for agreement, the results are a bit different for the guidelines: we observe over 65% of developers and 58% of testers either agreeing or strongly agreeing that the guideline is relevant (as opposed to 55% and 50% respectively).

Table 25, even though testers and developers agree that “social factors” is important for improving TX, there are mixed opinions or disagreements among over 30% of participants. One possible explanation is that, as an individual, the professional does not think his/her personality traits (extraversion, agreeableness, neuroticism, conscientiousness, and openness) will influence his/her way of carrying out the testing activities. On the other hand, the test or project manager considered this sub-factor as important for individual and team performance, as confirmed by the study of (Kanij et al., 2011).

With respect to the three factors: cognition, conation, and affection, we can conclude that their sub-factors’ guidelines are all relevant for improving TX, for all professionals. The guidelines were helpful to helping the participants obtain more understanding of the factors, and for this reason, there was a general increase in the ratings (although slight). Once more, we can observe in Figure 11 the guideline rankings per profile. We observe that the scenario did not change with respect to the factor rankings. The only difference is a better overall percentage agreement of the guidelines. Given this scenario, for developers and testers, conative guidelines were considered most important; for test and project managers, affection guidelines; for researchers and professors, cognitive guidelines. Based on these results, we can provide some recommendations to professionals on how to use these guidelines. We examine this, and other aspects in the following section.

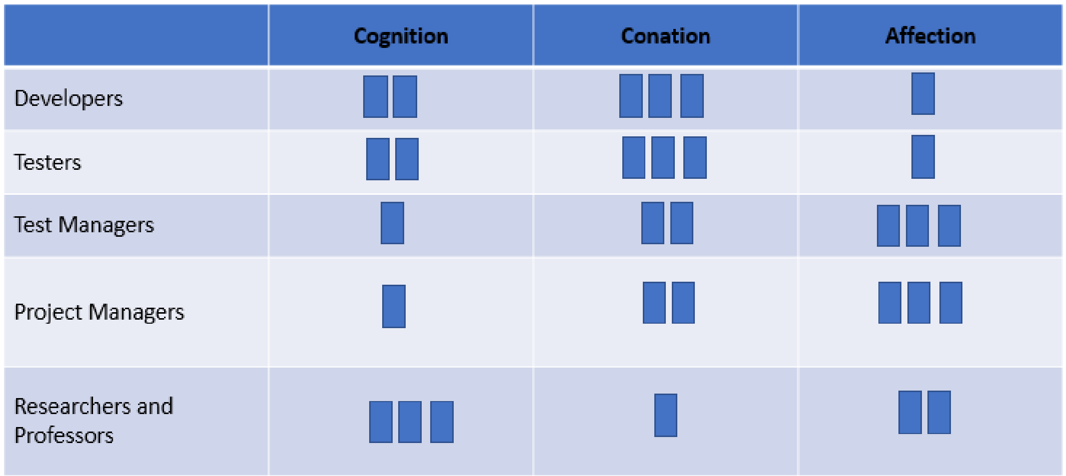

Figure 11: Ranking of Guidelines per Profile

6.5 Discussion

This section provides a description of the lessons learned from the survey and how the factors and guidelines can be recommended to software development professionals.

From the survey analysis described in the previous section, we observed a general agreement in all factors and sub-factors by professionals as being important, and their respective guidelines being relevant, for improving TX. We also observed general high levels of agreement among test managers, project managers and researchers and professors for all sub-factors and guidelines. Furthermore, we observed some discrepancy among testers and developers with respect to the “social factors” sub-factor. We also observed relatively higher ratings in the guidelines than their sub-factors due to better professionals’ understanding.

In addition, it can be observed that developers and testers showed similarities in the ratings for both sub-factors and guidelines. For both professionals, the cognitive factor and its sub-factors obtained the lowest overall rating, among which “platform” obtained the lowest rating. The next lowest rating was obtained for “affection”, even though “respect, attachment and belonging” and “skills” obtained general high evaluations. The highest rating was obtained for “conation”, where sub-factors like “commitment”, “motivation”, and “goals and plans” stood out. We therefore conclude that developers and testers believe they obtain better TX when they perceive the value of their contribution, much more than their perception of the testing infrastructure. This makes them more motivated and committed and therefore their intentions are aligned with their activities as well as their goals and plans. Also, their feelings about their work, especially the sense of respect, attachment and belonging increases their TX, and this better improves team spirit.

As for test and project managers, we observe a difference: with the exception of "processes" for test managers (25% discordance) and "motivation" for researchers and professors (40% neither agree no disagree), all sub-factor guidelines obtained 100% approval. The small sample might have played a part in this as there were just 4 test managers, 5 project managers and 5 researchers and professors who took part in the survey. A possible future work would be to increase the sample size for these professionals. Hence, based on these observations, we can recommend the factors and guidelines to the different professional profiles as follows:

a) Developers

For this profile, we recommend that the focus be on all factors and sub-factors, but should lay less emphasis on social factors. Greater emphasis should however focus on cognitive and conative factors, as well as "team" and "goals and plans" sub-factors;

b) Testers

For this profile, we recommend all factors and sub-factors. Greater emphasis should focus on sub-factors like: "team" and "goals and plans", and cognitive and conative factors;

c) Test Managers

We recommend all factors and sub-factors especially "team", "respect, attachment and belonging", "goals and plans" and "commitment" sub-factors. These professionals should set the example for testers and encourage team work and mutual respect among team members;

d) Project Managers

We recommend all factors and sub-factors especially "team", "respect, attachment and belonging", "goals and plans" and "commitment" sub-factors;

e) Researchers and Professors

We recommend all factors and sub-factors especially "team", "respect, attachment and belonging", "goals and plans" and "commitment" sub-factors. More emphasis should be place on cognitive factors. These professionals focus on educating students and younger researchers, and therefore cognitive factors such as testing tools, technical aspects and processes should be their main focus.

In the next section, we present the threats to validity of this study.

6.6 Threats to Validity

The threats to validity were organized into 4 categories: construct, internal, external, and conclusion.

Validity of conclusion: This study was carried out by the simple demonstration (or not) of the importance and relevance of the TX factors (and sub-factors) and their guidelines respectively, in improving TX. This went through a conception phase, based on the construction of a knowledge body constructed by means of a literature review.

Internal validity: in this study, the selection of participants was based on software development professionals who have ever developed a software before. This was also done randomly. In addition, the instrument used (questionnaire) went through thorough revision and a pilot study was performed for improvements.

Validity of construct: this study is characterized by the analysis of the importance of the factors (and sub-factors) affecting TX, as well as the relevance of their guidelines in improving TX.

External validity: participants were considered a sample and representation of the population because they have all developed software before.

In the next section, we present our conclusion and suggestions for future works.

6.7 Conclusion

In this chapter, we have presented the results obtained from a survey carried out with software development professionals from several countries. Overall, we obtained positive results, and therefore achieved the goals of this study. The factors, sub-factors and their guidelines were accepted by the sample of the software development community who participated. Therefore, we can conclude that the factors and guidelines that compose our TX Guideline-Based Approach have been approved.

There were some limitations to this work: most of the participants (87%) are from Brazil. Ideally, we would have love to have participants from different geographical locations and continents, apart from the countries that took part in this survey.

As future works, we intend to define metrics to measure TX. This will help to measure the TX of the tester before, during and after his/her testing activities. With the metrics defined, it will be possible to apply this approach in a real scenario, where software development professionals will use the guidelines as recommended, and their TX will be measured, so as to determine the feasibility of the approach.

CHAPTER 7 - CONCLUSION AND FUTURE WORKS

In this chapter, we present this work's conclusion, contributions, limitations and future works.

7.1 Conclusion

In this research, we have observed that there are many software testing tools, techniques, frameworks, and methodologies, and despite this, testers have little (if any) information about them, nor knowledge on how to use them. This has therefore stood as a barrier for the proper inclusion of test processes in development. In addition, some of these tools and techniques are expensive, difficult to use or even time-consuming. Despite these barriers, most research has focused their attention on tools, techniques and methodologies and neglected the humans who carry out testing, that is, testers. These professionals are the most important people in software development when it comes to guaranteeing software quality.

As a means of providing a solution to the problems, this research focused on investigating good testing practices by means of *Tester Experience*, which provides a means of improving developers' experience with testing as relates to their perception about testing infrastructure (e.g. platform, techniques, processes, skills and procedures), the value of their contribution (e.g. intention, plans, goals, motivation, commitment and alignment) and how they feel about their work (e.g. respect, team, attachment and belonging). We also provided TX-A, an approach made up of the TX factors and also guidelines on how TX should be applied by the tester in his/her daily testing activities in order to foster best testing practices in software development organizations.

After proposing the TX-A approach, we went on to evaluate it using a survey. Software development practitioners, including testers, developers, test managers, project managers, and researchers and professors participated in the survey. The results were positive, as over 95% of the practitioners agreed that the factors are important and their respective guidelines relevant for improving TX. As a results, the hypothesis of our research "*The factors and guidelines that compose TX-A are respectively important and relevant in improving the TX*" was proven to be true and therefore, the goals of this research achieved.

7.2 Contributions

Apart from the experience of defining and evaluating the Tester Experience-based Approach, which can serve as basis for other researches in the human aspects of software testing, this research provides other contributions, including:

1. The definition and conceptualization of Tester Experience. From the best of our knowledge, this is the first work of its kind;
2. The provision of guidelines as on how TX can be applied in the tester's activities;
3. An article published based on the findings and contributions of this research:
    a. *Tester Experience: Concept, Issues and Definition* – COMPSAC 2017;
    b. *Facing up the Primary Emotions in Mobile Software Ecosystems from Developer Experience* – WASHES, 2017 (co-authorship);

7.3 Limitations

This research had the following limitations:

1. The number of papers extracted and selected in the literature review were very few (seven). Perhaps a systematic mapping study would have been carried out in addition to the snowballing procedure. However, due to the novelty of the TX concept, and the adoption of relevant information from similar concepts (UX and DX), the results were not negatively affected;
2. The survey was composed of over 80% of participants from Brazil. We would have loved to have more international participants;
3. Just five project managers, five researchers and professors and four test managers participated in the research. Perhaps the results would have been more diversed as those obtained for developers (20 participants) and testers (12 participants);
4. This work does not provide metrics for measuring TX.

7.4 Future Works

Opportunities for future works include:

1. Define metrics on how to measure TX, in such a way that the TX of the software development practitioner can be measured before, during and after his/her testing activities;
2. The survey can be extended to include professionals from more countries, social and ethnic backgrounds, more test and project managers, and researchers and professors;

3. This work can be extended to provide a means of test inclusion in Software Ecosystems. This is a very promising field as there are currently no works in this area related to software testing;
4. Another opportunity will be investigating ways in TX can help establish startup and newborn companies;
5. Involvement of the tester in decision making as a means of improving his/her TX;
6. How the testing environment can serve as a platform for skills development;
7. Eventhough we were able to distinguish between DX and TX, the factors for each of these two concepts are similar. What changes are the roles of the professionals involved, that is developers for DX and testers for TX. Therefore, these two concepts can be aggregated in order to create just one main concept. This will be an interesting future work because the activities of developers and testers complement each other, and their main objective is a quality software end-product.

BIBLIOGRAPHY

Amen BM, Mahmood SM, Lu J. Mobile Application Testing Matrix. Fifth Int. Conf. Comput. Sci. Inf. Technol., 2015, p. 27–40.

AmericanMarketingAssociation. American marketing association dictionary [Online]. Available: http://www.marketingpower. com/ layouts/Dictionary.aspx?dLetter=B n.d.

Bach J. General Functionality and Stability Test Procedure for Certified for Microsoft Windows Logo Desktop Applications Edition. 2000.

Beer A, Ramler R. The Role of Experience in Software Testing Practice, 2008, p. 258–65. doi:10.1109/SEAA.2008.28.

Bertolino A, Bertolino A, Faedo IA. Software Testing Research : Achievements , Challenges , Dreams Software Testing Research : Achievements , Challenges , Dreams 2007.

Boehm B, Basili VR. Software Defect Reduction Top 10 List. J Comput Arch 2001;24:135–7.

Brakus JJ, Schmitt BH, Zarantonello L. Brand Experience: What Is It? How Is It Measures? Does It Affect Loyalty? J Mark 2009;73:52–68. doi:10.1509/jmkg.73.3.52.

Briand L, Labiche Y. Empirical studies of software testing techniques: challenges, practical strategies and future research. ACM SIGSOFT Softw Eng Notes 2004;29:1–3.

Carpretz LF. Bringing the Human Factor to Software Engineering. IEEE Softw 2014;31:104–104.

Cem Kaner. Black Box Software Testing (Professional Seminar), Risk Based Testing and RiskBased Test Management. [Online] Available at: Www.testing-Education.org 2002.

Cohen C, Birkin S, Garfield M, Webb H. Management conflict in software testing. Commun ACM 2004;47:76–81.

Cunha A, Greathead D. Does personality matter? an analysis of code-review ability 2007;50:109–12.

DeMarco T, Lister T. Programmer performance and the effects of the workplace. 8th Int. Conf. Softw. Eng. (ICSE ’85), 1985, p. 268–72.

Dhaliwal J, Gabriel C, Poston R, Paul X. Journal of Strategic Information Systems Alignment within the software development unit : Assessing structural and relational dimensions between developers and testers. J Strateg Inf Syst 2011;20:323–42. doi:10.1016/j.jsis.2011.03.001.

Donahue JF, Hall R, Roberts JA, Slaughter SA. Understanding the Motivations , Participation , and Performance of Open Source Software Developers : A Longitudinal Study of the Apache Projects 2006;52:984–99. doi:10.1287/mnsc.1060.0554.

Dummies.com. Ten Qualities of an Effective Team Player. [Online] Available: Http://www.dummies.com/business/humanresources/employee-Engagement/ten-Qualities-of-an-Effective-Team-Player/ n.d.

Ekwoge OM, Fontao A, Dias-Neto AC. Tester Experience: Concept, Issues and Definition. Proc - Int Comput Softw Appl Conf 2017;1:208–13. doi:10.1109/COMPSAC.2017.232.

Fagerholm F, Ikonen M, Kettunen P, Münch J, Roto V. Performance Alignment Work : How software developers experience the continuous adaptation of team performance in Lean and Agile environments. Inf Softw Technol 2015;65:132–47.

Fagerholm F, Munch J. Developer Experience : Concept and Definition. Int. Conf. Softw. Syst. Process (ICSSP ’12), 2012, p. 73–7.

Fontão ADL, Pereira R, Feijó Filho J, Dias-neto AC. MSECO-DEV : Application Development Process in Mobile Software Ecosystems MSECO-DEV : Application Development Process in Mobile Software Ecosystems. Conf. Softw. Eng. Knowl. Eng. (SEKE '16), 2016, p. 317–22.

Gannon. Human Factors in Software Engineering. IEEE Comput Soc 1979;12:6–7.

Gavalas D, Economou D. Development platforms for mobile applications: status and trends. IEEE Softw 2011;28:77–86.

Harty J. A Practical Guide to Testing Wireless Smartphone Applications. Synth Lect Mob Pervasive Comput 2014;4:1–99.

Hase S. Measuring Organizational Capability: Beyond Competence. Futur. Res. Res. Futur. Aust. Vocat. Educ. Train. Res. Assoc. (AVETRA '2000), 2000, p. 23–4.

Hassenzahl M. User Experience (UX): Towards an experiential perspective on product quality. 20th Conf. l'Interaction Homme-Machine (IHM '08), 2008, p. 11–5.

Hilgard ER. The trilogy of Mind. J Hist Behav Sci 1980;16:107–17.

Hyman R. Creative chaos in high performance teams: an experience report. Commun ACM 1993;36:56–61.

ISO 730-1989: IEEE Standard for Softwre Quality Assurance Plans. 1989.

ISO/IEC/IEEE29119-2:2013. Software and systems engineering - Software testing - Part 2: Test processes, [Online]. Available: http://www.iso.org/iso/catalogue_detail.htm?csnumber=56736. n.d.

ISO9241-210:2010. Ergonomics of human system interaction – Part 210: Human-centered design for interactive systems. Switzerland 2010.

ISTQB Exam Certification. Definition of Bugs. [Online]. Available: http://istqbexamcertification.com/what-is-defect-or-bugs-or-faults-in-software-testing/.

Kanij T, Grundy J, Merkel R. Performance appraisal of software testers. Inf Softw Technol 2014a;56:495–505. doi:10.1016/j.infsof.2013.11.002.

Kanij T, Merkel R, Grundy J. An Empirical Investigation of Personality Traits of Software Testers 2015.

Kanij T, Merkel R, Grundy J. An Empirical Study to Review and Revise Job Responsibilities of Software Testers 2014b:1–4.

Kanij T, Merkel R, Grundy J. A Preliminary Survey of Factors Affecting Software Testers. Aust. Softw. Eng. Conf., 2014c, p. 116–25.

Kanij T, Merkel R, Grundy J. An Empirical Study of the Effects of Personality on Software Testing 2013:239–48.

Kanij T, Merkel R, Grundy J. A Preliminary Study on Factors Affecting Software Team Performance. Int. Symp. Empir. Softw. Eng. Meas. (ESEM â€™ 11), 2011, p. 359–62. doi:10.1109/ESEM.2011.48.

Kasurinen J, Taipale O, Smolander K. Analysis of Problems in Testing Practices. 16th Asia-Pacific Softw. Eng. Conf., 2009, p. 309–14. doi:10.1109/APSEC.2009.17.

Law EL, Leicester LE, Hassenzahl M. Understanding , Scoping and Defining User eXperience : A Survey Approach. SIGCHI Conf. Hum. Factors Comput. Syst. (CHI 'O9), 2009, p. 719–28.

Liu Z, Gao X, Long X. Adaptive random testing of mobile application. ICCET 2010 - 2010 Int. Conf. Comput. Eng. Technol. Proc., vol. 2, 2010, p. 297–301. doi:10.1109/ICCET.2010.5485442.

Majchrzak TA. Improving Software Testing. 2012.

Majchrzak TA, Schulte M. Context-Dependent App Testing. Open J Web Technol 2015;2:27–39.

Merkel R, Kanji T. Does the Individual Matter in Software Testing ? 2010.

Merriam-Webster. Definition of Experience. Https://www.merriam-Webster.com/dictionary/experience n.d.

Merriam-Webster. Definition of Thorough. [Online] Available: Https://www.merriam-Webster.com/dictionary/thorough n.d.

Mockus A. Organizational volatility and its effects on software defects. Eighteenth ACM SIGSOFT Int. Symp. Found. Softw. Eng. (FSE '10), 2010, p. 117–26.

Myers GJ, Sandler C, Badgett T. The Art of Software Testing. 2011.

Pagano D. User Feedback in the AppStore : An Empirical Study. 21st IEEE Int. Requir. Eng. Conf. (RE '13), 2013, p. 125–134.

Panzl D. TEST PROCEDURES, A NEW APPROACH TO SOFTWARE VERIFICATION - 78.pdf. 2nd Int. Conf. Softw. Eng. (ICSE '76), 1976, p. 477 – 485.

Patwa S, Malvija A. Impact of Development Phases in Object Oriented Software Testing : A Survey. Int J Ethics Eng Manag Educ 2014;1:116–9.

Rodrigues A, Dias-Neto AC. Relevance and Impact of Critical Factors of Success in Software Test Automation Lifecycle : A Survey. 1st Brazilian Symp. Syst. Autom. Softw. Test. (SAST '16), 2016.

Rodriguez L. The Interpretation and Legitimization of Values in Agile's Organizing Vision. Eur. Conf. Inf. Syst. (ECIS '12), 2012, p. 10–3.

Sackman H, Erikson WJ, Grant EE. Exploratory experimental studies comparing online and offline programming performance. Commun ACM 1968;11:3–11.

Smith EK, Bird C, Zimmermann T. Beliefs , Practices , and Personalities of Software Engineers : A Survey in a Large Software Company. Microsoft Res [Online] Available http//research.microsoft.com/pubs/262594/smith-Chase-2016.pdf 2016.

Vegas S, Basili V. A characterisation schema for software testing techniques. Empir Softw Eng 2005;10:437–66.

Wohlin Bc. Guidelines for snowballing in systematic literature studies and a replication in software engineering. Proc. 18th Int. Conf. Eval. Assess. Softw. Eng. (EASE '14), 2014, p. 38.

APPENDIX 1 – Survey to Evaluate Tester Experience (Factors and Guidelines)

Phase 1 – Request for participants' consent to take part in the survey

Hello, we are Brazilian researchers investigating the concept of Tester Experience (TX). As an initial proposal, we have constructed a TX framework, which is made up of factors. In this survey, we intend to obtain feedback from the software development community about these factors, as well as the guidelines that accompany them. By obtaining your feedback, we will be able to improve the initial version of the framework and guidelines, and as such, make it available to the software development and testing community in order to improve the tester experience of the developer (or tester).

This is an academic research with no commercial interests. We will openly publish the results so everyone can benefit from them, making anonymous everything before doing so. Once the data is collected and analyzed, any personal information will be removed from the data and it will not be used at any time during the analysis or when we present the results. If at some point during this survey you want to leave, you're free to do so without any negative consequences.

Thanks a lot!

Arilo Claudio Dias Neto - Federal University of Amazonas (UFAM) - arilo@icomp.ufam.edu.br

Awdren de Lima Fontão - Federal University of Amazonas (UFAM) - awdren@icomp.ufam.edu.br

Oswald Mesumbe Ekwoge - Federal University of Amazonas (UFAM) - ome@icomp.ufam.edu.br

Phase 2 – Characterization of participant's profile

* Required

1. In which country do you currently live? *

2. Do you have any prior experience in software development? (please select as many answers that you think are relevant) * Check all that apply.

☐ I have never developed software before
☐ I have developed software in individual projects
☐ I have developed software as part of an academic team
☐ I have developed software as part of a team in an industry

3. Do you have any prior experience in software testing? (please select as many answers that you think are relevant) * Check all that apply.
   ☐ I have never tested a software before
   ☐ I have tested software in an individual project
   ☐ I have tested software as part of an academic team
   ☐ I have tested software as part of a team in an industry

4. How many years of experience in software development do you have? *Mark only one oval.
   ◯ Less than 1
   ◯ 1 - 3
   ◯ 4 - 6
   ◯ More than 6
5. How many years of experience in software testing do you have? *Mark only one oval.
   ◯ Less than 1
   ◯ 1 - 3
   ◯ 4 - 6
   ◯ More than 6
6. What is your current position in your organization? * Mark only one oval.
   ◯ Developer
   ◯ Tester
   ◯ Test Designer
   ◯ Project Manager Skip to question 17.
   ◯ Test Manager Skip to question 17.
   ◯ ____________________ Other: Skip to question 17.

Phase 3 – Evaluation of TX-A factors and guidelines

Definition

TX is defined as a means of capturing how testers think and feel about their activities within the software testing environment, with the assumption that an improvement of the tester's experience has an impact on the quality of the software. A tester in this case refers to anyone involved in testing a software product, which includes a developer when he/she needs to test his/her software product.

Factors

TX is composed of three main factors: Cognition, Conation, and Affection.

Cognition consists of factors that affect how testers perceive their testing infrastructure on an intellectual level. The testing infrastructure includes: testing tools, programming languages, platforms, frameworks, processes and methods.

Conation consists of factors that affect how testers see the value of their contribution. These include:

Alignment, intention, motivation, goals and plans, and commitment.

Affection consists of factors that influence how testers feel about their work (that is, feelings or emotions about work. These include: respect, attachment and belonging; social factors, and team.

Guidelines

Three main roles of the tester are considered for the context of this study: Test Manager, Test Designer, and Tester.

For each TX factor and sub-factor, a set of guidelines is defined for the role of the tester (Test Manager, Test Designer, Tester).

The following part of this questionnaire is to evaluate the factors that affect TX as well as the guidelines for each factor and role, in order to obtain feedback about their importance and relevance, as well as to obtain suggestions for improvement from participants.

Part 1: Cognition (Factors that affect how testers perceive their testing infrastructure)

The factors include:

1.1. Platform: Common desktop platforms include: Windows, Mac and Linux, while for mobile platforms, they are: Android, iOS and Windows Phone.

1.2. Technical Aspects: They include testing tools, techniques or frameworks, test automation techniques and tools, programming languages, training or certification on testing techniques and tools, ability to handle complex technical aspects and knowledge about specific testing techniques.

1.3. Processes: The ISO/IEC/IEEE 29119-2 model specifies test processes that can be used to govern, manage and implement software testing in an organization, project or testing activity.

1.4. Procedures: They are a fundamental specification of test cases to be applied to one or more target program modules. Facilitate software testing by allowing modules to be thoroughly tested outside the environment in which they will eventually reside.

1.5. Skills: The aim of the software tester is to encounter problems or faults in a software product, and to do this, good testing skills are required. Skills include: knowledge of testing, capacity to diagnose and solve problems, platform knowledge or application to be tested.

7. The following factors are important for improving TX * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
1.1. Platform	⬭	⬭	⬭	⬭	⬭
1.2. Technical Aspects	⬭	⬭	⬭	⬭	⬭
1.3. Processes	⬭	⬭	⬭	⬭	⬭
1.4. Procedures	⬭	⬭	⬭	⬭	⬭
1.5. Skills	⬭	⬭	⬭	⬭	⬭

8. Do you have any suggestions for improvement?

Guidelines for using "Cognition" in Test Processes include:

1.1. Platform: For the roles of Tester and Test Designer, knowledge on how to use these platforms is very important in order to perform tests. Intermediary programming skills is also a bonus.

1.2. Technical Aspects: For the roles of Tester and Test Designer, knowledge on how to use testing tools, techniques or frameworks in order to carry out a specific testing activity is necessary. It is the responsibility of he/she to develop the capacity to handle complex technical aspects when the need arises.

1.3. Processes: the Tester or Test Designer has the responsibility to know how to apply the processes described in the ISO/IEC/IEEE 29119-2 test process model in order to carry out his/her testing activities. The model is also made up of guidelines on how to use these processes.

1.4. Procedures: appropriate test case design techniques are to be used by the Tester and Test Designer; he/she should make sure the testing environment should be identical to the production environment in terms of hardware and software; he/she should verify if the metrics to be collected have been specified.

1.5. Skills: at each phase of dynamic testing, the Tester and Test Designer must show good interpersonal and communication skills. In the case of interpersonal skills, extra training may be necessary depending on the complexity of the task; working with a more experienced tester may make the task easier.

9 The following Guidelines of the TX factors are relevant for guiding the Tester or Test Designer in carrying out his/her testing activities * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
1.1. Platform					
1.2. Technical Aspects					
1.3. Processes					
1.4. Procedures					
1.5. Skills					

10. Do you have any suggestions for improvement?

Part 2: Conation (Factors that affect how testers perceive the value of their contribution)

The factors include:

2.1. Alignment: A well aligned software development team, whose sub-units (such as systems development, database management, network operations and architecture planning) are coherent, integrated, and harmonized with the testing team is essential for achieving all objectives and improving TX.

2.2. Intention: Retaining software developers and testers has been a problem in many organizations for decades. When they quit, they depart with critical knowledge of business processes. Therefore, job satisfaction is directly related to the tester's intentions. Testers' intentions are also influenced by their perceptions of usefulness, social pressure, compatibility and organizational mandate.

2.3. Motivation: Motivation arises from individual or team recognition for achievements or efforts. A tester or the testing team can be motivated when the software being developed is meaningful, challenging and valuable. Other aspects include: opportunities to innovate, flexible working hours, an infrastructure that allows employees to work from anywhere, empowerment to make decisions about their work, respect, etc.

2.4. Commitment: this is very important when selecting the right technical skill to get some work done. Reliability, constructive communication, active listening, active participation, willingness and openness to share, and attention to details are some important characteristics. Attention to details is a very important aspect of the software tester. Any detail omitted or negligence in his tests can cause severe impacts on the software product.

2.5. Goals and Plans: Testing managers and team leaders need to develop a clear and communicated purpose that is both compelling and that makes team members feel important. Testers are more likely to be committed to the purpose of the team if they are involved in creating it. An unclear mission will result in lack of focus and a low level of engagement and commitment towards achieving it.

11. The following factors are important for improving TX * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
2.1. Alignment	⬭	⬭	⬭	⬭	⬭
2.2. Intention	⬭	⬭	⬭	⬭	⬭
2.3. Motivation	⬭	⬭	⬭	⬭	⬭
2.4. Commitment	⬭	⬭	⬭	⬭	⬭
2.5. Goals and Plans	⬭	⬭	⬭	⬭	⬭

Guidelines for using "Conation" in Test Processes include:

2.1. Alignment: the relationship between the Tester or Test Designer and other team members, as well as his/her alignment with the testing project is crucial for compatibility with team members. This is especially critical in complex projects.

2.2.Intention: in order to carry out the testing activities, the Tester or Test Designer must maintain a positive attitude, and must be willing to cooperate with other team members throughout the whole process (in the case where development is not an individual effort).

2.3 Motivation: Tester and Test Designer: although most of the motivation comes from the kind of project, or from the managerial and organizational processes, the individual tester has his/her own part to play: dedication and commitment. These two factors may lead to his/her recognition or promotion, or any other aspect that can make him/her more accomplished and satisfied, hence more motivated.

2.4 Commitment: reliability, constructive communication, active listening, active participation, willingness and openness to share, cooperation, flexibility, team commitment, respectful and problem solver are key characteristics every Tester or Test Designer should have.

2.5 Goals and Plans: during all the testing activities, the Tester or Test Designer must make sure his/her goals and plans are in line with those of the development team and those of the testing goals and plans

12. The following Guidelines of the TX factors are relevant for guiding the Tester or Test Designer in carrying out his/her testing activities * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
2.1. Alignment	⬭	⬭	⬭	⬭	⬭
2.2. Intention	⬭	⬭	⬭	⬭	⬭
2.3. Motivation	⬭	⬭	⬭	⬭	⬭
2.4. Commitment	⬭	⬭	⬭	⬭	⬭
2.5. Goals and Plans	⬭	⬭	⬭	⬭	⬭

13. Do you have any suggestions for improvement?

Part 3: Affection (Factors that affect how testers feel about their work)

The factors include:

3.1. Social Factors: This is mainly related to the personality traits of the tester. These traits include: extraversion, openness to experience, neuroticism, conscientiousness and agreeableness. Each member of the software testing team falls into one of these personality factors, and therefore, will tend to affect the social interaction and work environment.

3.2. Respect, Attachment and Belonging: A high-performing software testing team has a particular identity and its members have a feeling of team spirit and pride. Social skills, intrinsic motivation to perform, and a desire for personal development are key traits of members in such teams. Self-motivation and dedication to the team's goals, within an environment of open communication and mutual respect, foster the commitment necessary for success.

3.3. Team: Working as a team is very crucial for software testing experience. The basis of a dynamic, cohesive team is trust. Each team member must implicitly trust teammates to do their part, keeping the team goals paramount in their actions.

14 The following factors are important for improving TX * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
3.1 Social Factors					
3.2 Respect, Attachment and Belonging					
3.3 Team					

Guidelines for using "Affection" in Test Processes include:

3.1. Social Factors: it is important for the tester or test designer to know which personality traits he/she possesses: extravert, introvert, open to experiences, neuroticist, conscientious or agreeable. Each of these personality traits can be decisive to the performance and alignment of the testing team.

3.2. Respect, Attachment and Belonging: desire for personal growth and development, mutual respect, feeling of belonging and attachment are expected from the tester or test designer.

3.3. Team: teamwork is crucial during software testing. The testing and development teams have to be in harmony in order to foster productivity and quality of software.

15. The following Guidelines of the TX factors are relevant for guiding the Tester and Test Designer in carrying out his/her testing activities * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
3.1. Social Factors	⬭	⬭	⬭	⬭	⬭
3.2. Respect, Attachment and Belonging	⬭	⬭	⬭	⬭	⬭
3.3. Team	⬭	⬭	⬭	⬭	⬭

16. Do you have any final remarks?

Stop filling out this form.

TX: Definition, Factors and Guidelines

Definition

TX is defined as a means of capturing how testers think and feel about their activities within the software testing environment, with the assumption that an improvement of the tester's experience has an impact on the quality of the software. A tester in this case refers to anyone involved in testing a software product, which includes a developer when he/she needs to test his/her software application.

Factors

TX is composed of three main factors: Cognition, Conation, and Affection.

Cognition consists of factors that affect how testers perceive their testing infrastructure on an intellectual level. The testing infrastructure includes: testing tools, programming languages, platforms, frameworks, processes and methods.

Conation consists of factors that affect how testers see the value of their contribution. These include:

Alignment, intention, motivation, goals and plans, and commitment.

Affection consists of factors that influence how testers feel about their work (that is, feelings or emotions about work. These include: respect, attachment and belonging; social factors, and team.

Guidelines

Three main roles of the tester are considered for the context of this study: Test Manager, Test Designer, and Tester)

For each TX factor and sub-factor, a set of guidelines are defined for each role of the tester (Test Manager, Test Designer, Tester).

The following part of this questionnaire is to evaluate the factors that affect TX as well as the guidelines for each factor and role, in order to obtain feedback about their relevance, as well as to obtain suggestions for improvement from participants.

Part 1: Cognition (Factors that affect how testers perceive their testing infrastructure)

The factors include:

1.1. Platform: Common desktop platforms include: Windows, Mac and Linux, while for mobile platforms, they are: Android, iOS and Windows Phone.

1.2. Technical Aspects: They include testing tools, techniques or frameworks, test automation techniques and tools, programming languages, training or certification on testing techniques and tools, ability to handle complex technical aspects and knowledge about specific testing techniques.

1.3. Processes: The ISO/IEC/IEEE 29119-2 model specifies test processes that can be used to govern, manage and implement software testing in an organization, project or testing activity.

1.4. Procedures: They are a fundamental specification of test cases to be applied to one or more target program modules. Facilitate software testing by allowing modules to be thoroughly tested outside the environment in which they will eventually reside.

1.5. Skills: The aim of the software tester is to encounter problems or faults in a software product, and to do this, good testing skills are required. Skills include: knowledge of testing, capacity to diagnose and solve problems, platform knowledge or application to be tested.

17. The following factors are important for improving TX * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
1.1. Platform	⬭	⬭	⬭	⬭	⬭
1.2. Technical Aspects	⬭	⬭	⬭	⬭	⬭
1.3. Processes	⬭	⬭	⬭	⬭	⬭
1.4. Procedures	⬭	⬭	⬭	⬭	⬭
1.5. Skills	⬭	⬭	⬭	⬭	⬭

18. Do you have any suggestions for improvement?

Guidelines for using "Cognition" in Test Processes include:

1.1. Platform: for the role of Test Manager, he/she should be able to determine the type of platform used during testing. Furthermore, it is mandatory for him/her to have a good mastery of the platform. In the case where the platform is novel, he/she should offer training to the testers.

1.2. Technical Aspects: for the Test Manager, he/she must know how to choose and use the testing tools, techniques or frameworks before prescribing them to testers and test designers. He/she must also decide which ones are best suitable for the project in context. He/she should also be able to handle complex technical aspects.

1.3. Processes: apart from having knowledge of how to use the ISO/IEC/IEEE 29119-2 test process model, the Test Manager should provide further training to testers and test designers about test processes if need be.

1.4. Procedures: appropriate test case design techniques are to be determined by the Test Manager; he/she should make sure the testing environment is identical to the production environment in terms of hardware and software; he/she should determine the types of metrics to be collected.

1.5. Skills: the Test Manager must show good interpersonal and communication skills with respect to testers and test designers. It is his/her function to assign skilled testers for particular testing tasks in order to obtain the expected results. In the case of unskilled testers, the test manager must make sure they receive the appropriate training needed to be become skilled.

19. The following Guidelines of the TX factors are relevant for guiding the Test Manager in carrying out his/her testing activities * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
1.1. Platform	◯	◯	◯	◯	◯
1.2. Technical Aspects	◯	◯	◯	◯	◯
1.3. Processes	◯	◯	◯	◯	◯
1.4. Procedures	◯	◯	◯	◯	◯
1.5. Skills	◯	◯	◯	◯	◯

20. Do you have any suggestions for

improvement?

Part 2: Conation (Factors that affect how testers perceive the value of their contribution)

The factors include:

2.1. Alignment: A well aligned software development team, whose sub-units (such as systems development, database management, network operations and architecture planning) are coherent, integrated, and harmonized with the testing team is essential for achieving all objectives and improving TX.

2.2. Intention: Retaining software developers and testers has been a problem in many organizations for decades. When they quit, they depart with critical knowledge of business processes. Therefore, job satisfaction is directly related to the tester's intentions. Testers' intentions are also influenced by their perceptions of usefulness, social pressure, compatibility and organizational mandate.

2.3. Motivation: Motivation arises from individual or team recognition for achievements or efforts. A tester or the testing team can be motivated when the software being developed is meaningful, challenging and valuable. Other aspects include: opportunities to innovate, flexible working hours, an infrastructure that allows employees to work from anywhere, empowerment to make decisions about their work, respect, etc.

2.4. Commitment: this is very important when selecting the right technical skill to get some work done. Reliability, constructive communication, active listening, active participation, willingness and openness to share, and attention to details are some important characteristics. Attention to details is a very important aspect of the software tester. Any detail omitted or negligence in his tests can cause severe impacts on the software product.

2.5. Goals and Plans: Testing managers and team leaders need to develop a clear and communicated purpose that is both compelling and that makes team members feel important. Testers are more likely to be committed to the purpose of the team if they are involved in creating it. An unclear mission will result in lack of focus and a low level of engagement and commitment towards achieving it.

21. The following factors are important for improving TX * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
2.1. Alignment	◯	◯	◯	◯	◯
2.2. Intention	◯	◯	◯	◯	◯
2.3. Motivation	◯	◯	◯	◯	◯
2.4. Commitment	◯	◯	◯	◯	◯
2.5. Goals and Plans	◯	◯	◯	◯	◯

22. Do you have any suggestions for improvement?

Guidelines for using "Conation" in Test Processes include:

2.1. Alignment: the Test Manager must make sure the testing team is well aligned to the testing project, since alignment is crucial for the success of testing. He/she is also in charge of making sure the testing team is aligned with the developing team, hence strengthening cooperation between the two entities.

2.2. Intention: since retaining software developers has been a problem in many organizations, the Test Manager is to make sure each tester is satisfied with his/her task, or adopt strategies to make him/her feel accomplished and comfortable when carrying out a specific task.

2.3. Motivation: the Test Manager should ensure the testing project adds more value to the team, including different challenges. Recognition for their work is also important for motivating testers. Furthermore, the testers should be given the opportunity to innovate, and perhaps flexible working hours, and offering empowerment to the testers in terms of decision making.

2.4. Commitment: it is important for the Test Manager to ensure reliability, constructive communication, active listening, active participation, willingness and openness to share, cooperation, flexibility, team commitment, respectfulness within the testing team.

2.5. Goals and Plans: the Test Manager should ensure that the testing project’s goals and those of individual testers are all in line in order to obtain the maximum participation of each tester. Clear operational goals should be defined, measurable and understandable by each tester. It is also advisable for the testers to be involved in decision making so that the goals and plans are clearly understood by everyone.

23. The following Guidelines of the TX factors are relevant for guiding the Test Manager in carrying out his/her testing activities * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
1.1. Alignment	⬭	⬭	⬭	⬭	⬭
1.2. Intention	⬭	⬭	⬭	⬭	⬭
1.3. Motivation	⬭	⬭	⬭	⬭	⬭
1.4. Commitment	⬭	⬭	⬭	⬭	⬭
1.5. Goals and Plans	⬭	⬭	⬭	⬭	⬭

24. Do you have any suggestions for improvement?

Part 3: Affection (Factors that affect how testers feel about their work)

The factors include:

3.1. Social Factors: This is mainly related to the personality traits of the tester. These traits include: extraversion, openness to experience, neuroticism, conscientiousness and agreeableness. Each member of the software testing team falls into one of these personality factors, and therefore, will tend to affect the social interaction and work environment.

3.2. Respect, Attachment and Belonging: A high-performing software testing team has a particular identity and its members have a feeling of team spirit and pride. Social skills, intrinsic motivation to perform, and a desire for personal development are key traits of members in such teams. Self-motivation and dedication to the team's goals, within an environment of open communication and mutual respect, foster the commitment necessary for success.

3.3. Team: Working as a team is very crucial for software testing experience. The basis of a dynamic, cohesive team is trust. Each team member must implicitly trust teammates to do their part, keeping the team goals paramount in their actions.

25. The following factors are important for improving TX * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
3.1. Social Factors	⬭	⬭	⬭	⬭	⬭
3.2. Respect, Attachment and Belonging	⬭	⬭	⬭	⬭	⬭
3.3. Team	⬭	⬭	⬭	⬭	⬭

26. Do you have any suggestions for improvement?

Guidelines for using "Affection" in Test Processes include:

3.1. Social Factors: it is the role of the Test Manager to know which personality traits each tester possesses: extravert, introvert, open to experiences, neuroticist, conscientious or agreeable. This is important for assigning tasks to testers based on their personality traits in order to achieve the testing goals.

3.2. Respect, Attachment and Belonging: the Test Manager should ensure that each team member has the feeling of team spirit, belonging and pride. He/she should also encourage the feeling of harmony and equality among the team members.

3.3. Team: the Test Manager should ensure the testers work as a team in order to achieve the testing goals. In addition, the team must be skilled and dedicated; there must be trust among members of the team, as well as with the developers.

27. The following Guidelines of the TX factors are relevant for guiding the Test Manager in carrying out his/her testing activities * Mark only one oval per row.

	I strongly disagree	I disagree	I neither agree nor disagree	I agree	I strongly agree
3.1. Social Factors					
3.2. Respect, Attachment and Belonging					
3.3. Team					

28. Do you have any suggestions for improvement?

29. Do you have any final remarks?