

*Radosław Hofman*

---

BEHAVIORAL  
PRODUCTS QUALITY ASSESSMENT MODEL  
ON THE SOFTWARE MARKET

---

DOCTORAL DISSERTATION

PROMOTER: PROF. DR HAB. WITOLD ABRAMOWICZ

POZNAŃ, 2011

## OUTLINE OF THE DISSERTATION

This dissertation presents the research related to software products' quality perception. The motivation for the research results from two main areas: the growth of importance of the software market, and the research results on the differences between nominal and descriptive economics agent analysis. The software market, although relatively new, is becoming one of the pillars of the modern global market. The second aspect is based on the observation that the actual processes and judgments formulated by customers on the software market do not comply with the theoretical models describing the quality of software products.

Based on empirical observation of the market, the author proposes a new explanation of the observed phenomena. The proposed model reveals the actual relations between the inherent characteristics of the product and the state of the observer's cognitive structures. The model is, however, hypothetical. Therefore, the author constructs the verification requirements based on contemporary economics research methods, and identifies independent variables hypothesized to be impacting on the process and software market-based data.

One of the most important aspects of the proposed research is related to scientific control over the experimental environment, especially the manipulation of the quality level of an evaluated software product. The author proposes an order relation between products, which allows the construction of lists of products ordered by quality level, and also the complete toolset required to perform the research.

The empirical research involves professional software evaluators, who were engaged to participate in the project on a commercial basis, in order to completely reflect real world circumstances. According to the research assumptions, the results may be related in the first place to professional activities related to software product quality evaluation. The research results support the thesis of the dissertation, pointing out that the actual processes related to software quality perception are influenced by cognitive processes characteristics.

The research results may be applied by the software industry, which is struggling with problems related to miscommunication with their customers. This dissertation explains the roots of this situation, pointing to methods which may be employed to reverse it.

The presented research is, according to the author's best knowledge, the first attempt to compare behavioral economics research results to a market of complex products, such as software products. The obtained results show the significant influence of cognitive structures on the quality assessment processes. This constitutes a motivation to continue developing the research plan proposed in this dissertation.

## SHORTCUTS AND ABBREVIATIONS

<b>ACT</b>	Adaptive Thought Control	<b>IEC</b>	International Electrical Committee
<b>AGM</b>	Alchourron-Gardenfors-Makinson model	<b>IEEE</b>	Institute of Electrical and Electronics Engineers
<b>AHP</b>	Analytic Hierarchic Process	<b>ISO</b>	International Standardization Organization
<b>ANOVA</b>	Analysis Of Variance method	<b>ISTQB</b>	International Software Testing Qualification Board
<b>APA</b>	American Psychological Association	<b>IT</b>	Information Technology
<b>API</b>	Application Programming Interface	<b>ITIL</b>	IT Infrastructure Library
<b>ATF</b>	Appraisal Tendency Framework	<b>ITSMF</b>	IT Service Management Forum
<b>BDM</b>	Behavioral Decision Making	<b>JTC1/SC7/WG6</b>	Joint Technical Committee 1, Sub-Committee 7, Work Group 6
<b>BDR</b>	Behavioral Decision Research	<b>MEG</b>	magnetoencephalograms
<b>CD</b>	Commission Draft	<b>NATO</b>	North Atlantic Treaty Organization
<b>CDT</b>	Causal Decision Theory	<b>OASIS</b>	Organization for the Advancement of Structured Information Standards
<b>COBIT</b>	Control Objectives for Information and related Technology	<b>OGSA</b>	Open Grid Services Architecture
<b>COTS</b>	Commercial Off The Shelf	<b>PET</b>	positron emission tomography
<b>D-E gap</b>	Description-Experience gap	<b>QoR</b>	Quality of Results
<b>DU</b>	Discounted Utility	<b>QoS</b>	Quality of Service
<b>EDT</b>	Evidential Decision Theory	<b>QUEST</b>	QoS Assured Composeable Service Infrastructure
<b>EEG</b>	Electroencephalography	<b>RAD</b>	Rapid Application Development
<b>EU</b>	Expected Utility and also European Union	<b>RES</b>	Reputation and Endorsement System
<b>FCM</b>	Factor-Criteria-Metrics model	<b>SERVQUAL</b>	Service Quality Model
<b>FDIS</b>	Final Draft of International Standard	<b>SEU</b>	Subjective Expected Utility
<b>fMRI</b>	functional Magnetic Resonance Imaging	<b>SOA</b>	Service Oriented Architecture
<b>GUI</b>	Graphical User Interface		
<b>HTTP</b>	Hyper Text Transfer Protocol		

**SOAP** Simple Object Access Protocol

**SQPM** Software Quality Perception  
Model

**SQuaRE** Software Quality Requirements  
and Evaluation

**SwEBoK** Software Engineering Body of  
Knowledge

**TestApp** Testable Application

**TMS** Transcranial Magnetic  
Stimulation

**TR** Technical Report

**U/EOU** Usefulness / Ease Of Use

**UAT** User Acceptance Test

**UML** Unified Modeling Language

**WSDL** Web Services Description  
Language

**XP** eXtreme Programming (also  
referring to a version of Microsoft  
Windows)

## TABLE OF CONTENTS

1	Introduction.....	7
1.1	Motivation.....	7
1.2	Goal and research questions.....	9
1.3	Research methodology.....	11
1.4	Organization.....	17
I.	Background.....	20
2	Software market.....	22
2.1	Goods on the software market.....	23
2.2	Industry of products for the software market.....	24
2.3	Software market products lifecycles models.....	31
2.4	Quality of goods on the software market.....	35
2.5	Evaluation of products on the software market.....	42
3	Customer decision theory.....	49
3.1	Decision theory overview.....	49
3.2	Quality perception for the purpose of decision making.....	57
4	Modern approaches to decision theory.....	63
5	Empirical decision research.....	69
5.1	Research methods.....	69
5.2	Customer rationality boundaries.....	71
II.	Conceptual Model.....	78
6	Identification of the variables impacting software quality perception.....	78
6.1	Empirical data from the market.....	78
6.2	Methods related to software industry.....	83
6.3	Methods based on the neoclassical economics approach.....	86
6.4	Methods based on behavioral economics results.....	87
6.5	Conclusion.....	89
7	Software Quality Perception Model.....	91
7.1	Construction of Software Quality Perception Model.....	91
7.2	Characteristics of the model.....	94
8	Software Quality Perception Model verification.....	97
8.1	Research methods overview.....	97
8.2	Requirements regarding the research design.....	99

8.3	Variables identification .....	101
8.4	Requirements regarding verification process execution .....	104
8.5	Verification summary.....	110
III.	Verification .....	111
9	Verification: tools and results .....	111
9.1	Experiment plan .....	111
9.2	Experiment variables.....	118
9.3	Experiment environment and tools .....	122
9.4	Experiment track record.....	129
9.5	Experiment results.....	131
9.6	Validity issues and discussion.....	142
9.7	Empirical research results summary .....	146
	Conclusions and outlook .....	150
10	Conclusion .....	150
10.1	Applicability of the results .....	150
10.2	Analysis of the thesis and objectives .....	153
10.3	Further research.....	158
11	List of figures.....	163
12	List of tables.....	165
13	References.....	167
	Appendix A – Raw Empirical Data.....	198

# 1 INTRODUCTION

This part introduces the scope of the dissertation, presenting the motivation for the undertaken research, its goal, research questions and methodology, and a detailed description of the dissertation structure.

## 1.1 Motivation

As the software market only emerged in the second half of the 20<sup>th</sup> century, products on this market are relatively new in comparison to the majority of human craft products. However, the importance of software products is difficult to overestimate. Software is present in phones, music players, cars, planes, steering devices and medical equipment, as well as in TV receivers and appliances. Software products support the economic activities of individuals and companies, provide entertainment, handle public safety etc. Therefore, the quality of software products has become an important aspect of these products (Kan, et al., 2010).

The concept of software quality is, however, ambiguous (Suryn, et al., 2003). Although attempts to define software quality date from the 1970s (Kobyliński, 2005), no commonly accepted model of software quality has yet been proposed (in fact Kobyliński underlines that there is no even commonly accepted software quality related vocabulary). Software engineering scientists underline (Basili, 1993) that software products are intangible and seem to be more complex in terms of quality measurement. However, at every stage of the software production lifecycle, when a software product is presented to individuals (e.g. customers, users), those individuals formulate their opinion about the product's quality in a relatively short time. This aspect lies beyond the scope of software engineering, however the process of products evaluation is an important part of market behavior analyzed by economic science.

Neoclassical economic models assume that humans follow the *homo economicus* model: they are perfectly rational, utility maximizing and in possession of complete and relevant information (Simon, 1956). Economic models based on these assumptions can be applied to the analysis of various aspects of behavior, thus providing economics with a framework to predict future decisions. However, in several recent field observations these assumptions were partially refuted due to the rationality limitations of humans. This research area is named behavioral economics and aims to provide descriptive models of economic behavior (Camerer, et al., 2003).

The individual psychology aspect has been present in economic science since its beginning (cf. the works of Adam Smith). A good example supporting this observation is the description of the Diamonds and Water Paradox. In *The Theory of Moral Sentiments* Adam Smith described a phenomenon of disproportion between appreciation and regret if the same amount of good was to be given to or taken away from a person (1759). This phenomenon was studied and supported with empirical evidence by behavioral economists in the 20<sup>th</sup> century (positive-negative asymmetry). A significant milestone in human economics decision making analysis was achieved by marginalists in the 19<sup>th</sup> century, who explained several decision making processes phenomena through the perspective of individual needs saturation (e.g. the law of diminishing marginally utility Angner, et al., 2007). Several speculations regarding the feelings and attitudes that influence economic decisions may be found in early 20<sup>th</sup> century literature (cf. the works of Irving Fisher and Vilfredo Pareto; John Maynard Keynes has made a notable contribution to psychological insights ).

In the second half of the 20<sup>th</sup> century, the works of Herbert Simon<sup>1</sup>, Gorge Kantona, Harvey Liebenstein and Gary Becker<sup>2</sup> suggested the need for relaxing the assumption about perfectly-rational decision makers, and called for psychological insights analysis in regard to economic decision making processes. Herbert Simon, in his prize-winning lecture, pointed out that economics was proclaimed to be a psychological science by Alfred Marshall in the opening words of *Principles* (Simon, 1979). The acceptance of expected utility and discounted utility models, descriptive decision under risk models, uncertainty processes, intertemporal choice etc., is regarded as the beginning of behavioral economics. The works of Daniel Kahneman<sup>3</sup> and Amos Tversky, Thaler, Loewenstein and Prelec provide ample replicable evidence revealing the anomalies of perfectly-rational decision maker models.

The models for the quality assessment of products on the software market seem to assume that humans are perfectly-rational, utility maximizing, and possess complete information about the products and their attributes (Hofman, 2011). This last assumption is not related to the type of attribute (visible or hidden), or to the ability of verification by the observer. Behavioral economics proposes several methods for expressing and evaluating software

---

<sup>1</sup> Simon was awarded the Nobel Prize in Economics in 1978 for pioneering research into the decision-making process within economic organizations

<sup>2</sup> Becker was awarded the Nobel Prize in Economics in 1992 for having extended the domain of microeconomic analysis to a wide range of human behaviors and interactions, including nonmarket behavior

<sup>3</sup> Kahneman was awarded the Nobel Prize in Economics in 2002 for having integrated insights from psychological research into economic science, especially concerning human judgment and decision-making under uncertainty



product quality. However, despite the fact that there is neither a commonly accepted software quality model nor a software evaluation process, humans are able to assess and evaluate software quality.

Understanding and describing actual software quality evaluation processes is an important challenge for software engineering and economics (Ernst, et al., 2010). Software vendors often fail to satisfy user quality requirements. Even if they provide the user with a good quality product, they may hide important information regarding its quality. This may lead to decreasing the size of the market, as predicted by George Akerlof<sup>4</sup>. If vendors could use knowledge about real quality assessment processes, then they could deliver a product which satisfies users. Consequently, the satisfied users would be a positive reference for the vendor and possibly a source of future orders.

Research in the area of customers' and users' perceptions of software products utilizing insights from the economic sciences is already being postulated by the industry (Davis, 2004). The research described in this dissertation represents, according to the author's best knowledge, the first attempt to apply behavioral economics research methods in software projects in regard to software quality assessment. Brooks has suggested that the quality of the product is affected by organizational structure and communication within the producing organization (1995). It is also possible that perceived product quality is affected by the communication between customer and producer.

## **1.2 Goal and research questions**

The general objective of this dissertation was to develop a new method for clarifying and more precisely predicting customers' decision making processes related to software quality. The development and verification of this model required also a new explanatory research method focused on the descriptive aspects of the software evaluation process.

The general objective consists, therefore, of the following research goals:

- 1) Identification of the variables impacting on the software quality assessment process during the perception process,
- 2) Development of the descriptive Software Quality Perception Model,
- 3) Elaboration of a research method for the verification of the model,
- 4) Elaboration of methods for the manipulation of the environment configuration to emulate occurrences taking place in the software market,

---

<sup>4</sup> Akerlof was awarded the Nobel Prize in Economics in 2001 for analyses of markets with asymmetric information

- 5) Elaboration of the required research environment,
- 6) Execution of the verification, and the assessment of the proposed model.

The general objective and goals constitute the thesis of this work: *The model of customer assessment processes related to software quality, taking into account phenomenon described by behavioral economics, allows for the more accurate prediction of customer decisions than commonly used normative models.*

The objective of this dissertation required the development of the Software Quality Perception Model, which was based on market observations and abstraction. Verification of the model required the elaboration of a framework of empirical research procedures, which took into account an extended validity analysis. It also required preparation of the evaluation environment, which allowed for the reconstruction of real world projects' occurrences. The empirical verification required specific tools, which were prepared for the manipulation of the environment and for the support of the research (gathering data, managing subjects etc.).

The research required the use of following methods and tools:

- The method of gathering data expressing subjective software quality assessment based on Osgood's semantic differential (1957),
- Detailed verification procedures (experiment plans) based on the behavioral economics paradigm,
- Tools for evaluated applications management and for the support of the research process,
- Dedicated applications for the purpose of evaluating the method .

The measure of the achievement of dissertation goals and thesis verification was based on the analysis of the corroboration for the proposed model in the empirical research. The predictive power of the proposed model was compared to existing models.

The results from the research presented in this dissertation include following products:

- The analysis of the software market in the context of the quality of versions based on a sample of 15 projects (presented in section 6.1);
- The Software Quality Perception Model based on secondary research results, theory, and direct observations of the market (presented in section 7.1);
- A variables list hypothesized to affect the software quality perception process, based on the literature review and market empirical observations (presented in section 8.3);

- A method for software quality manipulation for the purpose of research, based on quality level differences (presented in section 8.4);
- Experiment plans (presented in section 9.1);
- Tools for conducting the experiments (discussed above, and presented in detail in section 9.3);
- Software errors analysis categorization, based on a sample of 100 projects (presented in section 9.3.1); and
- Empirical research data and analysis (presented in section 9.5).

The products listed above constitute the genuine contribution of this dissertation, and may be divided into two types. The first type includes observation results based on the software industry data gathered by the author, and the conceptual model based on the theoretical knowledge. The second type of products is related to experimental research related to software quality perception processes, including requirements, plans, tools, and the results.

The general conclusion from the dissertation corroborates the thesis presented in this section (as outlined above).

### **1.3 Research methodology**

The selection of the research methodology was based on the careful review of the history of scientific inquiry and the contemporary epistemological perspective, which provides methods for assessing scientific progress. From the historical perspective, Plato is considered the father of scientific method (Hirschheim, 1985). However, the early methods were limited only to the inductive approach. Other historians emphasize that the first definitions of scientific methods were made by Aristotle (Kuhn, 1996), who distinguished between exact and approximate reasoning, and used an inductive approach for the abstraction of empirical observations (Nijland, 2002), arguing that knowledge comes from experience, and that a human is initially a *tabula rasa*.

#### **1.3.1 The general approach to research methodology**

The central debate in the philosophy of science is related to the problem of demarcating between scientific and non-scientific knowledge and methods (compare Lakatos, et al., 1999). Another important area of this debate is related to arguments for and against the empirical approach to the hypothesis formulation (compare abductive reasoning (Peirce, 1908) and Friedman's (1953) approach). The debate has not revealed an ultimate set of answers to

questions related to defining “science”. However, epistemologists’ thoughts may be used as guidelines or a set of best practices in the process of scientific inquiry.

The modern view on scientific inquiry is based mainly on the works of Karl Popper, Thomas Kuhn, Imre Lakatos and Paul Feyerabend (Hausman, 1989). Popper was influenced by positivists, originating from the “Vienna Circle” in the early 20<sup>th</sup> century. Positivists postulated the verificationism requirement, advocating that scientific and cognitive theories have to be verifiable (Ayer, 1936) and have to define truths as logical propositions: analytic (*a priori*) and synthetic (*a posteriori*) (Achinstein, et al., 1969). Popper proposed the separation between scientific and meaningful theories, proposing falsification (compare *modus tollens*) as the single and universal method of scientific inquiry (1959), and rejecting the inductive approach. Popper also described his view on how scientific progress is made, referring to Pierce’s view on the need for empirical evidence to be used for the testing of new theories, underlining also Pierce’s fallibilism. Popper’s approach is regarded as being normative (Harper, 1999), although his position on the hypothetico-deductive model considered the hypothesis as a “guess” (Popper, 2002). However, it is worth mentioning that Popper, following Pierce, believed that science is fallible, and that therefore the results of scientific inquiry cannot be regarded as certain or even probable (1959). In opposition to Popper’s standpoint, Duhem and Quine argued that a theory cannot be verified in isolation, because the failure of the verification may imply that the premises were incorrect. Therefore, the whole context should always be evaluated (Harding, 1976). Popper’s model regarding scientific methods (1959) was enhanced by Hempel and Oppenheim (1948), and named the deductive-nomological model.

Kuhn focused on the descriptive approach, describing how scientists actually conduct their research and how scientific progress is made (1970). He proposed the idea that scientists use a certain paradigm until the number of doubts and contradictions calls for a scientific revolution, which results in new paradigm formulation (1996). In his view, new ideas are accepted not because they have been proven to be correct, but by generational shift among the researchers. Regarding the testing of new theories, Kuhn has advocated the view that the way theory is tested depends on the theory itself. Therefore, once the theory has been accepted it is testable only with the use of tests which it has already passed (Kuhn, 1961).

Lakatos is known for his methodology of proofs and refutations (1976). Lakatos, like Kuhn, analyzes the actual processes that are employed by scientists. However, he defines his perspective as “popperian” as he was attempting to find a consensus between Popper’s and Kuhn’s viewpoints. Lakatos proposed the idea of a research program, which contains a core

and “protective belt” of auxiliary hypotheses (1980). In his opinion, the research program may be progressive (if new facts are discovered, new research techniques are being used, more accurate predictions are being made etc.) or degenerating (when there is a lack of growth). Lakatos has also proposed a demarcation criteria between science and pseudoscience. In his opinion, if the theory does not predict new, previously unknown facts, then it should be regarded as pseudoscience (Lakatos, 1974). Notably, Lakatos classified neoclassical economics as pseudoscience during his lecture in 1973 (Lakatos, et al., 1999). This has remained a challenge to his antecessors. For the same reasons, Latsis considered Friedman’s approach as non-scientific, publishing an article on this view in the *British Journal for the Philosophy of Science*, edited by Lakatos (1972). Twenty years later, Helena Coronin, using Zahar’s criterion (Steuer, 2003), was arguing that providing novel explanations of old and well known facts should also be regarded as science, while other researchers were also pointing out that the novel facts predicted by Friedman-Phelp’s model contradicted Lakatos’s statement.

Feyerabend represented the position called epistemological anarchism. He argued that there are no methodological rules that are always used by scientists (1975). He also pointed out that no theory is consistent, or deals with all relevant factors, due to the stochastic nature of many processes (Preston, et al., 2000). Feyerabend (1975) also claimed that Lakatos’s philosophy of research programs is “anarchism in disgust”. His perspective had a strong influence on contemporary science. He is well known for the phrase “whatever goes”. His influence may be seen, for example, in Richard Feynman’s words: “[The] philosophy of science is about as useful to scientists as ornithology is to birds” (Thagard, 2009).

### **1.3.2 The research area**

The adoption of the scientific method and research model should be preceded by an identification of the scientific area related to the research. Generally, for the scope defined in the previous section, two main research areas were considered: software engineering and economics. This research could have been located in the field of computing science, particularly in the field of software engineering. The general direction in this field is focused on the aspect of processes related to the development and delivery of software processes. Therefore, software engineering covers also the aspect of communication with customers and users (defined as stakeholders). The research results related to the perception of product’s quality could thus be regarded as a natural part of this field.

However, there has been little research to date on cognitive and subjective theories in relation to this area. There have emerged several models related to the cognitive aspects of developers' productivity or to quality perception (see chapter 2 for a review), although their acceptance is rather uncommon (Basili, 2007). The second important aspect from the perspective of computer science is the omission of market decisions resulting from the assessment. Dan Ariely, in his lectures, has underlined that many practitioners, with whom he talks, believe that subjective beliefs or biases disappear when the decision involves real and significant assets (compare also Han, et al., 2007). Therefore, the software engineering models which are focused on the "real" quality of the product typically leave cognitive issues aside. From this perspective the subject of this dissertation lies mainly beyond the scope of software engineering.

The natural research area of this dissertation is therefore economics. Economics is focused on the market behavior of agents in normative, descriptive and explanatory dimensions. Within the area, since the second half of the 20<sup>th</sup> century, there has been a research area related to cognitive-based decision making processes: behavioral economics (Camerer, et al., 2003). Research within the paradigm of behavioral economics typically focuses on simple decisions related to common products. However, the extension of this research to sophisticated products, manufactured in a non-typical manner, extends existing theory and may be considered as a part of research program in terms of Lakatos's approach. This dissertation is therefore considered to be located in the field of economics, however regarding the evaluation of the products described within the scope of software engineering.

The first extended reflections on economics methods appeared in the works of John Stuart Mill and Nassau Senior (Peart, et al., 2003). Mill described two kinds of inductive methods: *a priori*, where the researcher determines laws resulting from the observations made, deduces consequences and tests the conclusions; and *a posteriori*, which is a method of direct experience (Lewisohn, 1972).

Contemporary mainstream economics is referred to as neoclassical. The origins of neoclassical economics date back to the late 19<sup>th</sup> century, with economists such as Jevons stating (among other ideas) that economic agents are maximizing their happiness (1871). However, in the early 20<sup>th</sup> century economists like Pareto, Hicks and Allen stated that rationality is more about ranking and choice, and that therefore hedonic aspects should not be considered (Ross, 2007). Psychology, in the early 20<sup>th</sup> century, had limited explanatory abilities. However, researchers like Veblen (1919) have investigated psychology in relation to economics. Knight (1921) asserted that psychology is irrelevant to economics, and that

economics should establish universally valid laws. By the end of the 1920s, mainstream economics was stripped of psychology (Raaij, 1985).

In 1953, Friedman formulated the idea of positive economics (1953), following Popper's ideas. Within this stream, hypotheses were to be evaluated only under accepted premises for their predictive power, while the reality of assumptions was not analyzed. Friedman directly stated that testing central explanatory generalizations is a mistake. Therefore, research into whether or not agents maximize their utility was perceived as being unscientific. The other consequence of Friedman's model was the negation of Popper's falsification postulates, because theories based on unrealistic assumptions could not be seriously tested. However, Friedman thought that theoretical economics hypotheses should be tested by their comparison to data. His claims regarding the unreality of assumptions were not accepted by logical empiricists (Hutchison, 1956).

The boundaries of rationality assumptions were discussed by Simon (1956) and others. This new research area, taking advantage of developments in psychology, cognitive revolution, anthropology, computer science, linguistics etc. was called behavioral economics. This new paradigm was focused on empirical methods, and these led to the rise of another stream called experimental economics (Roth, et al., 1988).

### **1.3.3 Requirements resulting from selected research methodology**

Ensuring high quality research results requires the adoption of an adequate research method. Based on the review presented in the previous section, the general approach to solving the research problem was based on sophisticated falsificationism (Popper, 1959), (Lakatos, 1970). The requirements for this approach were assessed as being relevant for the scope of this research (see section 6.3). In this approach, a new theory can be accepted if it fulfills the following criteria:

- 1) It has excess empirical content, which allows the researcher to predict or explain novel facts not predicted by previous theory,
- 2) It contains the non-falsified part of previous theory, or
- 3) Some new predictions have been confirmed empirically.

Noting the aforementioned definition of scientific program (Lakatos, et al., 1980), this dissertation should be regarded as being part of the behavioral economics research program, using their methods in order to research the software market (until now behavioral economics focused mainly on simple products).

The scientific model typically used in empirical research is the hypothetico-deductive model (Brody, et al., 1993) proposed in the early 19<sup>th</sup> century by William Whewell (1858), and extended in the 20<sup>th</sup> century by Popper and Lakatos. The model proposes the algorithm consisting of:

- 1) Gathering of data (or “analysis”)
- 2) Formulating the hypothesis (or “abstraction”)
- 3) Deducing the consequences from the hypothesis
- 4) Corroborating or falsifying the hypothesis (see Godfrey-Smith, 2003)

The above model is typically adopted for scientific research within behavioral economics (compare Camerer, et al., 2003). In this dissertation, the above listed phases are described in separate parts in this dissertation, as presented in Table 1-1.

Phase	Description
Gathering of data (or “analysis”)	This phase consists of the review of relevant methods described in the areas of software engineering, decision theory and behavioral economics (part I of dissertation), and is extended in the discussion of the empirical observations from the software market, which are presented in chapter 6.
Formulation of the hypothesis (or “abstraction”)	The hypothesis formulated in this dissertation reflects the model of the actual software quality perception process, and is described in chapter 7, section 7.1.
Deducing the consequences from the hypothesis	The consequences deduced from the hypothetical model were identified and described in chapter 7, section 7.2.
Corroboration or falsification of the hypothesis	Verification of the hypothetical model is discussed in chapter 8, while its planning and execution are discussed in part III of this dissertation. Part III contains also the analysis of the empirical evidence which appeared to support the model, thus confirming the thesis of this dissertation.

Table 1-1 Mapping of the hypothetico-deductive research model stages onto parts of this dissertation  
(source: own study)

Based on the above described approaches, a set of acceptance criteria for the research method may be also defined. An acceptable research method ought to be:

- 1) Testable, since science relies on evidence to validate its theories and models, using some formal methods (e.g. falsification),
- 2) Transparent, as the researcher should keep records of scientific inquiry for further investigation (ie. there should be full disclosure),



- 3) Repeatable and robust, as the results should not be arrived at by chance,
- 4) Controllable in respect to errors, since the method should allow the researcher to control or minimize the influence of variables impacting on the results and observational errors.

Kuhn's list of criteria for plausible theories contains also the need for external conformity, an ability to unify etc. However, such criteria are difficult to apply to a theory which aims to be revolutionary, even partially (Szenberg, et al., 2004) (for example external conformity with Friedman's model; compare also Thagard, 1978). Therefore, these criteria will not be considered in this dissertation.

The research method applied in this dissertation used both empirical and secondary materials. The secondary materials were mainly the existing research results related to quality perception, goods evaluation, software quality modeling and decision making processes under risk and uncertainty.

Part of the secondary materials could not be verified (e.g. the prerequisites used for the development of software quality models), and it does not seem purposive to repeat general experiments described by behavioral economists. Consequently, it was impossible to directly compare the results of empirical evidence with gathered secondary materials (e.g. in the area of assumptions made whilst conducting experiments). This gap between primary results and secondary materials may have affected the possibility of directly evaluating the conceptual model. Therefore, empirical research was required for the purpose of evaluation.

It should be noted that an analogical gap is typical for research conducted in the field of behavioral economics, where the aim of the research is to reveal the actual processes performed by economic agents (List, 2004). Therefore, the gap has to be strictly monitored in terms of results validity.

## **1.4 Organization**

The dissertation is divided into three parts: background analysis, conceptual model and validation. These parts are related to the general research method described in the previous section. Each part organizes chapters devoted to specific subjects, as described below. The dissertation begins with the Introduction, and ends with the summary presenting further research directions.

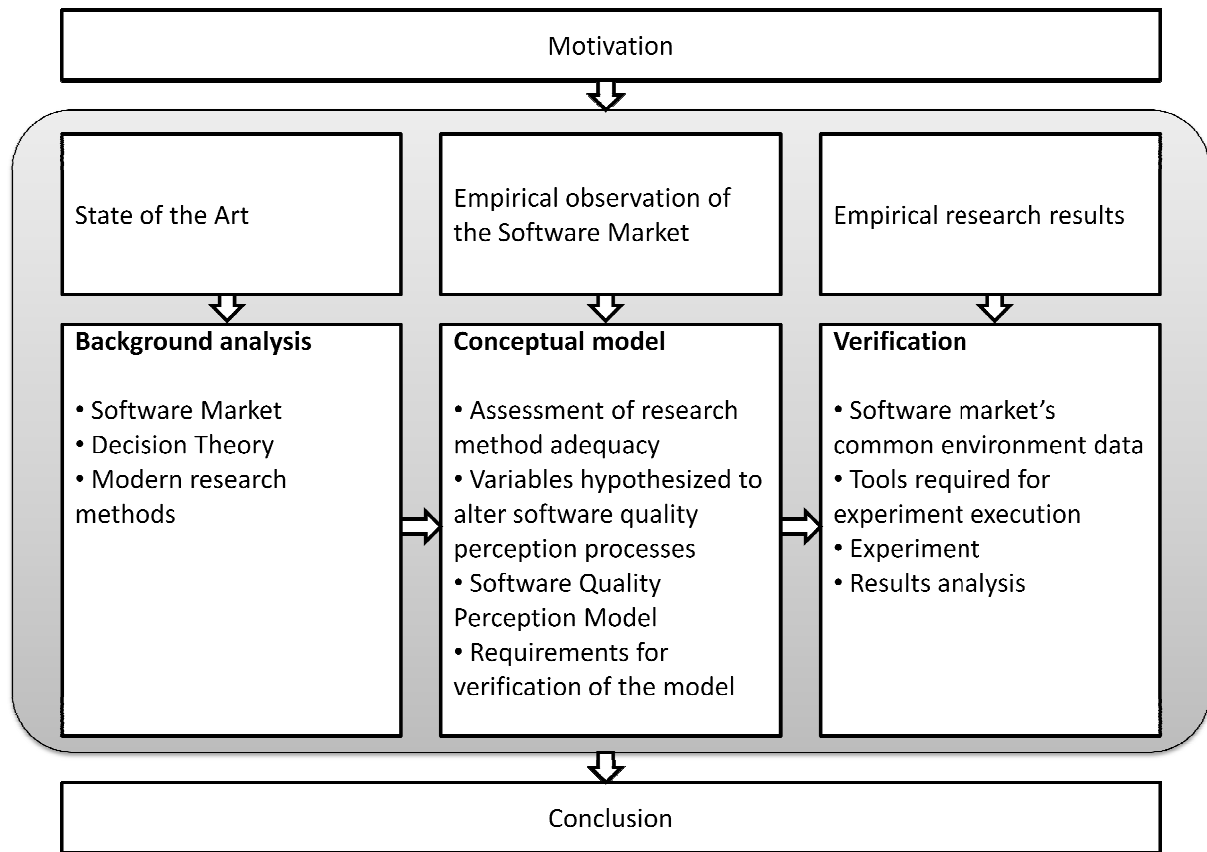


Figure 1-1 Organization of the dissertation (source: own study)

The chapters in part I summarize an extensive literature review for the areas of the software market, software engineering and software quality engineering (chapter 2), customer decision theory (chapter 3), as well as modern approaches to decision theory - including behavioral economics (chapter 4). Chapter 5 outlines the research methods used. The analysis is devoted to the role of decision makers (i.e. customers and users of a software product) in market decision making, especially regarding their assessment of a product's quality (judgment stage) and the limitations of rational behavior observed by behavioral economy scientists.

The second part outlines the conceptual model developed in the course of this research. In chapter 6, the first phase of the research is described. The observations gathered in regard to the software market, and the methods that can be used for abstraction (hypothesis formulation), are discussed. Chapter 7 introduces the Software Quality Perception Model. This model was developed as a hypothetical model based on observation and secondary sources. Chapter 8 describes the consequences resulting from this model, and the requirements for verification in terms of the acceptance criteria described in the section above.

The third part of this dissertation is devoted to verifying the proposed model. In chapter 9, detailed verification procedures and tools are described. This chapter contains also an analysis of the validity of the verification procedure. The final part of this chapter summarizes the empirical results of the research.

The concluding chapter evaluates whether or not the objectives of the research have been achieved. The final part of chapter 10 presents new research directions following on from the research presented in this dissertation (continuance of the research program). Such future research may take advantage of the tools, methods, techniques and results of this dissertation.

## I. BACKGROUND

The software market emerged in the 1950's, and has since grown continuously. From the beginning, the overlooking of decision making processes and their underpinnings by software market participants was based on neoclassical economics, which assumed the rationality of decision makers, complete information about products, and utility or profit maximization. In the following decades, economics research has shown direct violations of these assumptions, mainly in the behavioral economics research area (Camerer, et al., 2003). Economists have adopted methods for analyzing human motivation and understanding behavior, although these advances were not adopted in software market analysis.

The subjective nature of human behavior and decision making processes modeling have been the subject of study since the 19<sup>th</sup> century. This stream of thought was mainly rejected as a result of the neoclassical revolution at the beginning of the 20<sup>th</sup> century (Camerer, et al., 2003). However, there were several publications speculating on the psychological underpinnings of decisions. Akerlof (2003) states that Keynes' *General theory* (1936) is one of the greatest contributions to behavioral economics. However, most theorists at that time hoped to build up a scientific basis for the economy, rejecting psychological insights based mainly on introspection (Angner, et al., 2007).

The behavioral approach to economic science is currently perceived as being part of the mainstream. The works of Herbert A. Simon (1956), Garry S. Becker (1968), George A. Akerlof (1970), A. Michael Spence, Joseph E. Stiglitz and Daniel Kahneman (1979), and Robert Aumann (1976) won the Bank of Sweden Prize in Memory of Alfred Nobel in Economics for their authors in 1978, 1992, 2001, 2002 and 2005 respectively. The prize for Daniel Kahneman was shared with Vernon L. Smith, who was awarded the prize for his research results in experimental economy (Nobel Foundation, 2009).

Software market researchers tried to create predictive models of customers' behavior in regard to software quality assessment. However, their main focus was on normative models. In 1996, Kitchengam and Pflieger summarized five different views regarding the quality of products on the software market at that time (Kitchenham, et al., 1996):

1. Quality as an abstract, meta-physical term – an unreachable ideal, which shows the direction where products are heading to but will never attain,
2. Quality as a perspective of a user, considering attributes of software in a special context of use,

3. Quality as a perspective of a manufacturer, seen as compliance with stated requirements and following the ISO 9001:1994 view,
4. Quality as a product perspective, understood as an internal characteristic resulting from measures of product attributes, and
5. Quality as a value based perspective, differing depending on the stakeholder for whom it is defined.

The main perspective in the software quality related literature concentrates on the technical understanding of quality. However, in 2001 the ISO/IEC 9126 (ISO/IEC9126-1, 2001) standard showed that quality in use was one of three perspectives on software quality. The characteristics of the quality in use perspective were ambiguous. However, the approach to defining quality was started (Côté, et al., 2006). The recently developed ISO/IEC 25010 standard (ISO/IEC25010 FDIS, 2011) attempts to address the user's perspective on a much deeper level, although it still refers to technical rather than psychological underpinnings.

It is remarkable that there is no single and interchangeable software quality model. There is also no interchangeable description of the software evaluation process. Modern software quality models and evaluation processes do not address the problem of subjective perspective, hidden quality attributes, framing effects, and other cognitive issues related to judgment formulation processes.

The assumptions regarding a user's perfect rationality and goals, and their possession of all relevant information, are related to the role of customers (or users) in a software project, and are significant. In standard approaches, the user is expected to elicit a complete and stable set of requirements and to objectively evaluate each version of the delivered application. In agile approaches, the user is expected to take part in the project, elicit and decide on requirements, and objectively evaluate prototypes and the final product.

Finally, this literature review raises questions regarding the potential influence of judgment formulation about product quality and decision making processes on the software market. Are these processes objective and predictable? Can the normative models proposed by software engineering researchers be used? Are experienced users resistant to the framing effect, as suggested by neoclassical economists? Are the processes subjective, and are users highly influenced by cognitive biases during the assessment of a product's quality? The following chapters of this dissertation address these questions.

The following information in this chapter presents the general background and concept formation for the thesis. First, general information regarding the software market is presented, then decision theory is introduced. Further on, the modern theories and paradigms of decision

theory are described. Finally, the current research methods regarding actual decision processes are presented.

## 2 SOFTWARE MARKET

In mainstream economics, the market is defined as a structure which allows buyers and sellers to commit a transaction via exchanging money, goods, services and/or information (Simon, 1979). Market categories reflect the degree of economic freedom, regulatory institutions, geographic location, and goods that are traded etc. However, the general elements (e.g. transaction parties, goods and their value, price, decision etc.) seem to exist in every market.

The software market may be described as a horizontal market containing structures in which buyers and sellers commit transactions regarding the purchase of COTS<sup>5</sup> products, dedicated products developed according to customers' orders, and the use of electronic services (Blokdijs, 2008). Trading in software products seems to be similar to trading in other markets: COTS products are purchased in the same way as other types of repeatable products, whilst dedicated software products are ordered similarly to complex engineering products (e.g. buildings, ships etc.). The use of electronic services also seems to be comparable to any market where services are purchased (Barros, et al., 2005).

However, there are some important differences (Papazoglou, et al., 2002). Several electronics services are offered free of charge. Therefore, it is inconvenient to use the term 'buyer' when describing users of these services. At a deep analysis level the transactions related to software products are also different from other markets. The key difference regards the nature of the software product and an inability to evaluate the product without considerable expense (in most cases the complete evaluation of the software product is much more expensive than the product itself Patton, 2005).

An interesting perspective of the rapidly changing software market was described by Schumpeter (1950). According to Schumpeter, modern capitalism accepts monopolies. However, sometimes these monopolies are destroyed by emerging products, new technologies etc. Schumpeter does not address income loss, but indicates that monopolies can potentially be swept from the market by innovations made by their competitors. This vision seems to be a typical threat for large vendors on the software market (Schmalensee, 2000).

---

<sup>5</sup> COTS – Commercial Off The Shelf (ie. standard product)

The size of the software market is difficult to ascertain. Software vendors are at the same time customers, and there are no reporting mechanisms (at least in some countries) to distinguish between components bought for their inclusion into large company projects and products bought for a vendor's own use (for example, if the estimate of the software market's size is based on the sales volume of software vendors, then in the case of subcontracting, the same product is included in market size more than once). Ascertaining the size of the electronic services market seems to be difficult, because the services are partially offered free of charge for users and the cost is paid for by sponsors or advertisement issuers.

The size of the global software market was estimated to be about US\$ 303 billion in 2008, with forecast growth to US\$457 billion in 2012 (Datamonitor, 2009). The e-commerce market in Europe was estimated to be worth US\$133 billion in 2006, with an estimated growth to US\$407 billion by 2011 (eMarketer, 2006). Gartner predicts that by 2015 the Internet marketing part of the market will reach US\$250 billion, and that the number of adults able to transact online will grow to three billion (Gartner Research, 2010). Studying the annual reports of some of the largest market participants, it may be seen that in 2009 Microsoft's annual revenue was US\$59 billion (Microsoft, 2010), SAP's US\$15 billion (SAP, 2010), and Google's US\$23 billion (Google, 2010) (additionally, companies that offer software and hardware products earned in 2009: HP US\$118 billion, IBM US\$103 billion, Intel US\$37 billion, and Apple US\$32 billion CNN, 2010). Although there is no clear idea regarding the size of the software market, the above figures show that its size should be considered as large.

## **2.1 Goods on the software market**

Goods traded on the software market are different from those in other markets (Basili, 1993). The most important difference is related to informative boundaries about products. The "consumption" of professional software products often affects important areas of customer activity. However, it is impossible to foresee all consequences of using a certain product. This feature may be perceived as being common to the food market, as food products affect consumers' health but it is difficult to foresee the consequences of a specific diet without scientific research.

The second important difference is the dependency of the product's usage on external conditions. Unlike physical products, software or the web service may behave in unexpected ways even when the context of their use seems to be typical (or near typical). Software products are used in a vast number of contexts, and the customer cannot foretell if the product will perform correctly in all of them (Kan, 2002).

The above characteristics may be described as hidden product attributes (Braddon-Mitchel, et al., 1996), although the intangibility of software products (Basili, 1993) puts these attributes beyond the tools and methods of evaluation (especially from a customer's standpoint).

Moreover, the products on the software market may be perceived as having a very short lifecycle. If "the product" represents a single version of the software product, then in many situations the lifecycle length of this product may be no more than 24 hours (Lindstrom, et al., 2004). In this time, another product may be released and deployed into billions of computers around the world (e.g. via the Microsoft automatic update process). This situation reinforces the product's intangibility and reduces the sense of understanding the product.

However, repeatable products (e.g. computer games, commercial off the shelf products etc.) are, in many situations, perceived as being typical goods and are analyzed through the perspective of typical product management patterns (Dymek, 2000).

## **2.2 Industry of products for the software market**

Software engineering describes processes of software development, implementation and maintenance, and also attempts to improve these processes (Roger, 2001). The motivation of improving software development techniques results from the extension of software applicability, increasing competition in the software market, and growing quality expectations. The term "software engineering" has been used since the 1950's. However, the first remarkable use of this term dates from the late 1960's, where it was used in the title of a NATO conference in Berlin (IEEE SwEBok, 2004).

At that time the software industry had been concerned with the software crisis (Jaszkiwicz, 1997). Authors had identified the rapid growth of the computational power of computers, which allowed the broader use of these machines in business and everyday life (Dijkstra, 1972). Reports had also described the threats posed by computers to human lives, health and assets resulting from software malfunctions (e.g. the Therac 25 incident) (Hofman, 2007).

In the 1980's, the software industry slowly accepted the conclusion that there is no single solution to software quality problems (Brooks, 1986). This could be interpreted as a failure of software engineering, which aimed to solve software crisis problems. On the other hand, it may be considered as offering proof of the maturity of the discipline, and its altered focus on the various aspects of software development.



The following decades have introduced new challenges in the development of software products. The rapid growth of the Internet and new models of software usage (e.g. software as a service, web services etc.) raised the number of software services suppliers and software users (Hofman, 2007). The importance of software products and their quality is indicated by the following facts: software failures caused more than 4,000 deaths and cost billions of dollars (compare Kobyliński, 2005; McConnell, 2004).

The software engineering discipline may be defined as a set of technical knowledge regarding every stage of the software development process in every possible lifecycle model, or as the disciplined development and evolution of software systems based on processes, principles and technical methods (Basili, 1993). The definition of software engineering developed by IEEE<sup>6</sup> is: “(1) The application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software; that is, the application of engineering to software” (IEEE SwEBok, 2004). The Software Engineering Body of Knowledge has become an International Standardization Organization standard (ISO/IEC TR 19759, 2006).

Modern software engineering is adapting the experimental methods developed originally for psychological, sociological and behavioral economics purposes (Basili, 1993). Researchers develop new theories and afterwards use the experimental paradigm to evaluate them or to propose new theories, which may increase the accuracy of these theories (Hochstein, et al., 2008). However, this research focuses mainly on the observable impacts of the use of certain techniques, methods, tools etc. on software engineering processes (see Sauer, et al., 2000, Basili, 2007). The customer’s perspective is considered mainly as being that of a stakeholder’s during the software project.

Software engineering identifies several sub-disciplines associated with software lifecycle stages. These sub-disciplines and their techniques are also subject to standardization by the International Standardization Organization (ISO), as well as the International Electrical Committee’s (IEC) Joint Technical Committee 1 (JTC1) and Sub-Committee 7 (SC7). Although the literature regarding software engineering is broad, standards, which result from international cooperation aiming to express common sense of meaning, are used for the review of sub-disciplines within Software Engineering.

---

<sup>6</sup> Institute of Electrical and Electronics Engineers

### **2.2.1 Software requirements**

The software requirements of software engineering sub-disciplines describe the elicitation, analysis, specification, and validation of requirements for software (IEEE SwEBok, 2004). The goal of the software requirements gathering stage in a software project is to establish a comprehensive set of requirements describing the expected characteristics of the product. This process typically assumes that stakeholders are the source of the requirements (ISO/IEC12207 FDIS, 2007). In formal approaches the requirements are elicited and signed off by stakeholders, while in agile approaches stakeholders accept requirements based on prototype evaluation (Beck, 2000). Other approaches also place the responsibility of requirements acceptance upon stakeholders (e.g. the evolutionary approach Krzanik, 1988).

Software requirements describe the functional behavior of future software products (functional requirements) and requirements for operation (non-functional requirements) (IEEE SwEBok, 2004). The approach presented by the most recent SQuARE model (ISO/IEC25000, 2005) considers quality requirements as a super set of non-functional requirements.

During the 1990's, research showed that 20 to 60 percent of software errors lay in the requirements and analysis stage (U.S. Department of Defence, 1996), (Patton, 2005). The latest approach to the software requirements stage necessitates the performance of validity analysis upon the requirements using reviews, prototyping, formal validation or acceptance testing techniques (IEEE SwEBok, 2004).

### **2.2.2 Software design**

The design of a software product follows the software requirements. The goal of this stage is to describe the software architecture, internal decomposition to components, and their interfaces. The level of description must be detailed enough to allow software construction (Bobkowska, 2001).

The design not only breaks a software product into components, but also describes interrelations and interactions between those components. Additionally, the design should also describe the software's planned deployment (the physical location of the components), considering the functional, informational and technical aspects of the environment (Dymek, 2000). Typically, "trade-offs" are considered to optimize important characteristics of the future product (e.g. the location of information sources, computational components and information recipient requires the analysis of data volumes, network performance and queries characteristics).

Design concepts assist the designer, as they present the designer with a good set of patterns for designing. The design is expressed in semi-formal languages such as the Unified Modeling Language (UML). The design maps the requirements onto software components. The functional requirements are mapped onto dynamic components, while non-functional and quality requirements are mapped onto static design characteristics as well as whole tiers of the designed software, units or global product characteristics. Additional support is offered through the use of design patterns or design strategies (IEEE SwEBok, 2004).

Models created by Maciaszek (2009) or Bobkowska (2001) predict the quality of the designed software at this stage. The design itself is also subject to quality verification, employing reviews, static techniques, simulation and prototyping. Errors made at this stage comprise about 30% of the total number of errors in software products (U.S. Department of Defence, 1996).

### **2.2.3 Software construction**

Software construction describes a set of activities resulting in product preparation. The goal of this stage is to release a product compliant with the design, minimizing the complexity, anticipating changes, constructing a verifiable product, and using standards. It is expected that software components are tested at the component level (unit testing and integration testing) (IEEE SwEBok, 2004).

The construction is performed according to a plan based on formal or agile approaches. One non-formal construction method is the eXtreme Programming approach proposed by Beck (2000). The technology and programming language affects the construction of the software, and the ability to verify units of the product apart from the complete product.

Errors at this stage result in 25% to 40% of the total number of errors in the software product stage (U.S. Department of Defence, 1996), (Patton, 2005).

### **2.2.4 Software testing**

The goal of the testing stage is to evaluate the software product's quality and to improve it by identifying errors. Typically, a complete and implemented software product is tested, although the evaluation plan may include the evaluation of prototypes and internal releases (IEEE SwEBok, 2004).

The Software Engineering Body of Knowledge regards tests as a dynamic verification of system behavior, while other approaches assume that static verification is also a testing activity (compare ISO/IEC29119 CD, 2009). Static verification at this stage typically covers

compliance between the software product and the requirements, and the design and verification of the internal quality characteristics (ISO/IEC25000, 2005).

Dynamic testing techniques are divided into black box testing, smoke tests, and white box testing as a measure of how much the testing relies on the knowledge of internal mechanisms. From a different perspective, test classes are divided into functional, performance, stress, configuration, reliability, security, safety and usability tests depending on the measure type (ISO/IEC29119 CD, 2009). In the literature, the term regression testing is used to describe the repetition of previously performed test scenarios after a software version has been modified to ensure that the product preserves its prior functionality.

The testing discipline develops ample detailed test design techniques. In regard to test case definition, these techniques cover equivalence partitioning, boundary-values analysis, decision tables and finite-state based approaches. In some cases tests are designed according to formal specification, or randomly generated test cases are used (IEEE SwEBok, 2004).

The software testing process is often described using the V model (see section 2.3.1). The sequential phases of software construction produce the specifications. The sequential phases of a testing process verify those specifications. The concept of testing in the V model is presented in Figure 2-1.

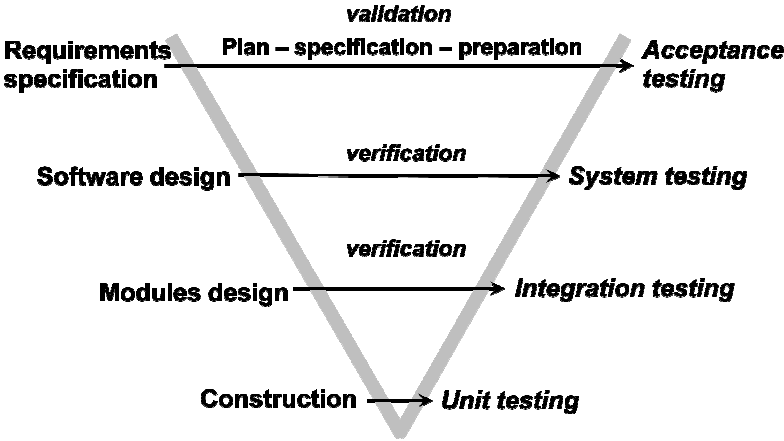


Figure 2-1 Testing in the V model (ISTQB, 2008)

The Software Engineering Body of Knowledge (IEEE SwEBok, 2004), Testing Foundation Syllabus (ISTQB, 2008) and the project of software testing international standard (ISO/IEC29119 CD, 2009) define several levels of testing. The first level is the unit testing, which is typically performed during the construction stage. When the units are integrated their interactions are the subject of integration testing performed typically by the development team. System testing is performed by a test team (by the same company or an independent test unit).

The final testing stage is the acceptance testing performed by or with the customer test team. The software engineering literature defines the goal of this stage as being to confirm that the requirements have been fulfilled. From the customer's perspective this is typically the first stage of the software product testing. This situation is typical for a sequential lifecycle model (see section 2.3.1).

### **2.2.5 Software maintenance and operation**

Software products are often intended to be used for a long period. The goal of software maintenance is to ensure a process of problem solving and developing software changes (IEEE SwEBok, 2004).

In the maintenance stage the software product is modified according to an agreed change requests. Changes are processed according to the software lifecycle model (typically: gathering and analyzing the requirements, designing the change, developing the change, testing the software product as a whole, and implementing a new version). Change in software construction or implementation may result in unexpected behavior in an unchanged area of the product. Software engineers advise project managers to perform an extensive set of regression tests irrespective of the scope of the change.

The costs involved in the maintenance stage are considerable in relation to the total software cost during its lifecycle. 67% of the total cost related to an IT project is assigned to the maintenance stage (Schach, 1992). Most of these costs are related to changes, enhancements, operation etc., and not to error correction (Pigosky, 1996), (Dymek, 2000), (Roger, 2001), (Wiederhold, 2006).

Pfleeger (2001) suggests that “maintenance has a broader scope, with more to track and control than development”. The extent of software maintenance covers processes associated with service management (IT Service Management Forum, 2007). Approaches such as ITIL (Information Technology Infrastructure Library) and COBIT (Control Objectives for Information and related Technology) (ITGI, 2007) concentrate on the value added for business use resulting from the IT (Diao, et al., 2008).

### **2.2.6 Software configuration management**

The software product quality for its users is dependent on the software characteristics and their operation processes (e.g. availability) (IT Service Management Forum, 2007). The configuration of system parameters covers both hardware and software items. This area is regarded as an important area of dynamic software quality assurance (ISO/IEC25000, 2005).

The goal of software configuration management is to support the lifecycle processes with reliable information about Configuration Items (IEEE SwEBok, 2004).

Software configuration management aims to be a lifecycle-independent process. Typical configuration items to be managed are: source code packages, compilers and libraries information, hardware configuration, system software, network configuration etc. Configuration management processes assume the need to audit and control processes to ensure reliable information handling in case of non-authorized changes in configuration, which seems to be one of the problems for quality assurance during system operation (IT Service Management Forum, 2007).

### **2.2.7 Software engineering management**

Management practices for software projects differ in their nuances from general project management theory. Therefore, the area of software engineering management represents the state-of-the-art in managing software projects. The main goal of this area of practice is to ensure that development, implementation or maintenance processes are systematic, disciplined and quantified (IEEE SwEBok, 2004).

The concepts and activities defined within this area form an extensive list covering activities related to agreement management and requirements negotiation, the planning and enactment of software, review and evaluation activities, software engineering monitoring, and project closure activities (IEEE SwEBok, 2004). There are several approaches to the scope definition of management processes (compare ISO/IEC12207 FDIS, 2007), although the core of the management's role is to plan, execute and check quality and risk management.

### **2.2.8 Software engineering process**

The software engineering process is an area of knowledge addressing the technical management of a software project. The main objective of this area is the implementation of new or improved processes in an acquiring organization (IEEE SwEBok, 2004).

The software engineering process sets up a framework for the measurement, assessment and implementation of new processes. The important part of this area is that it emphasizes tailoring processes to the needs of a project, customer or organization. The software engineering process focuses on continuous improvement, establishing the process infrastructure, improvement planning, change implementation and post-implementation reviews (IEEE SwEBok, 2004).

## **2.3 Software market products lifecycles models**

Software lifecycle modeling is an area of both software engineering (concerning the strategy of software construction) and of a software product's lifecycle. The Software Engineering Body of Knowledge places this area strictly within the construction stage (IEEE SwEBoK, 2004). However, however, an analysis of a typical product lifecycle shows that this process covers the entire scope of marketing product lifecycle management (compare Sääksvuori, et al., 2008).

The main difference in these two approaches concerns the definition of the product, as mentioned in section 2.1. From the perspective of software engineering, the product may be defined as a version of software (e.g. Microsoft Windows XP build 2600.xpclient.010817-1148) or a set of versions with a set of patches (e.g. Microsoft Windows XP with Service Pack 1). The definition used for marketing or formal purposes may be based upon non-technical decisions and strategies.

Software lifecycle models are typically divided into models assuming the sequential development of the desired product, and models assuming iterative development (which feature a greater number of cycles of delivering a new version to the customer). In a broader perspective taking the maintenance stage into account, the whole process may be perceived as a long-term evolutionary model (Lehman, et al., 1997). Agile software development is considered to be a different type of lifecycle model.

### **2.3.1 Sequential lifecycle models**

Sequential lifecycle models assume no repetitions of the project phases, with the exception of necessary feedback on preceding phases. Phase products remain constant after the phase is closed. Consequently, the requirements for the software product are stable during the project (ISO/IEC29119 CD, 2009).

An example of a sequential lifecycle model is the waterfall model presented in Figure 2-2.

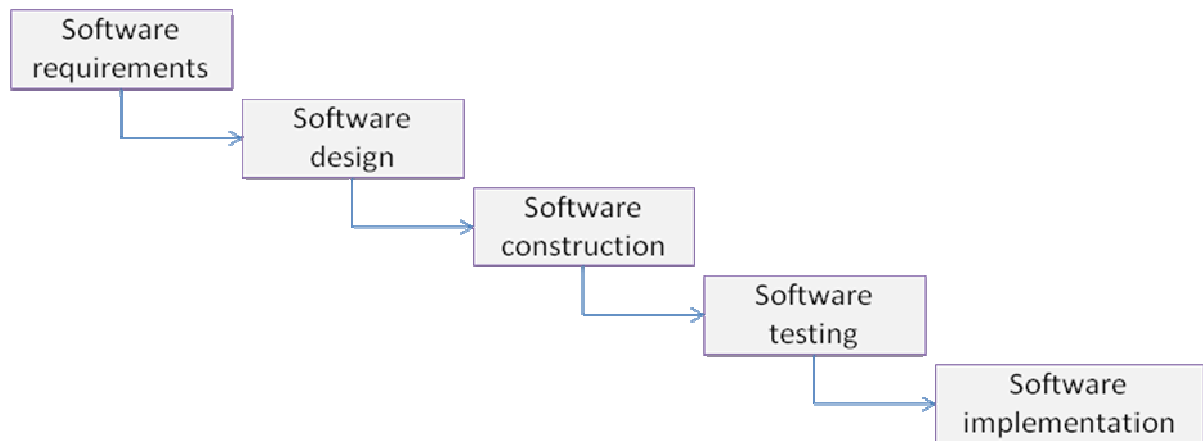


Figure 2-2 A typical waterfall lifecycle model (Rajlich, 2006)

Each phase of the sequential model should define the completion criteria (IEEE SwEBOK, 2004). Phase products are verified according to the completion criteria.

A modern sequential lifecycle model is the V model (ISO/IEC29119 CD, 2009). This model contains software requirements, design, and construction phases with corresponding testing phases. The idea of the model described by Boehm is presented in Figure 2-3. This idea is similar to the model of testing described in section 2.2.4 above, however Boehm regards unit testing as a part of software construction process therefore he did not describe them as a separate phase.

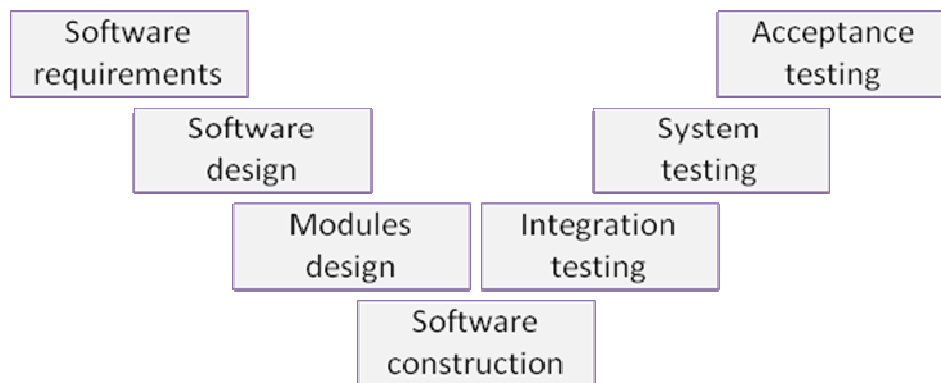


Figure 2-3 A typical V model lifecycle (Boehm, 2006)

Sequential approaches are used mainly in high risk or publicly funded projects (Dymek, 2000).

### 2.3.2 Iterative lifecycle model

Iterative lifecycle models or evolutionary models assume that the customer is not able to exactly elicit their requirements or that the requirements will continue to change. The product



is released iteratively, and therefore entails repetitions of phases. After the product is released, the customers provide feedback for the consecutive version. During the preparation of a new release, the requirement specification or software design - especially the software construction - may be changed (ISO/IEC29119 CD, 2009).

The overall approach towards the phases of the software development process is similar to that of sequential lifecycle models. Additional tasks occurring in the iterative lifecycle models are re-engineering tasks during the development stages and regression testing tasks for the evaluation stages.

One of the most popular iterative lifecycle models is the spiral model (Jaskiewicz, 1997). This model assumes sequential full development cycles where each cycle follows the waterfall model. Each cycle enhances the software scope and refines its current characteristics.

Contemporary approaches to the spiral model propose the risk driven development of each cycle (ISO/IEC29119 CD, 2009), using the waterfall model for the final product delivery (Boehm, et al., 1994).

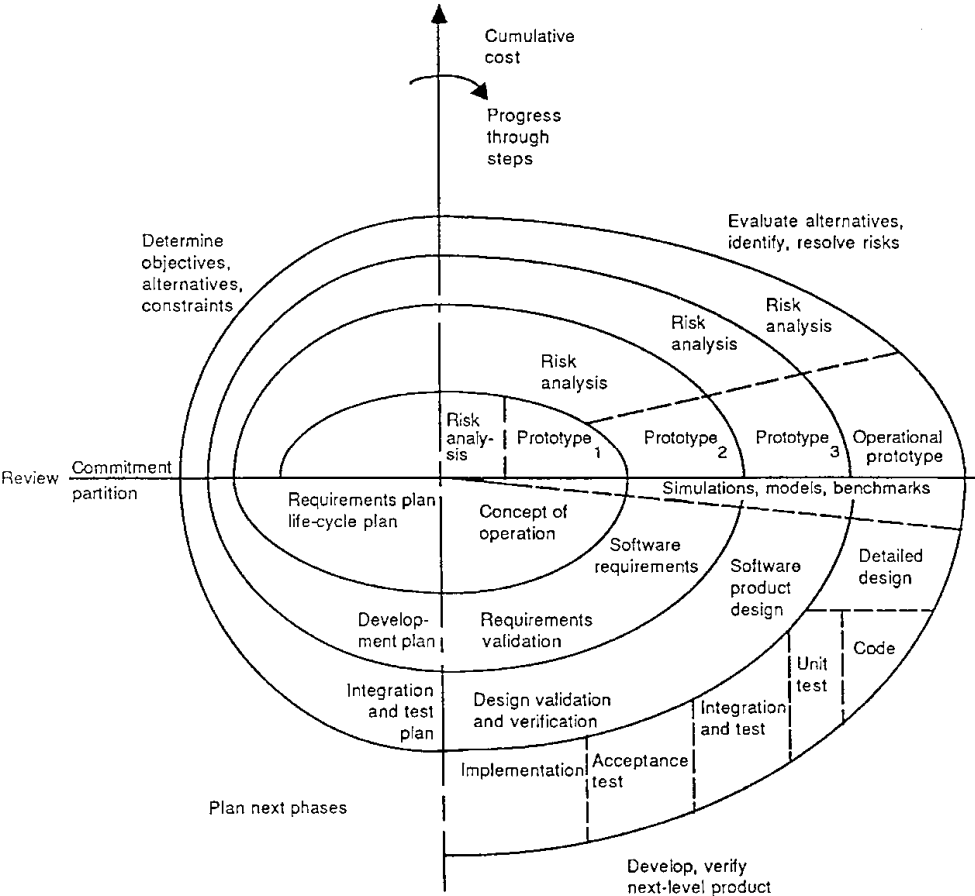


Figure 2-4 A modern spiral lifecycle model (ISO/IEC29119 CD, 2009)

Another example of the iterative lifecycle model is the RAD (Rapid Application Development). This model was proposed by James Martin (Apelbaum, 2002), and has since formed the basis for several different types of iterative lifecycle models (ISO/IEC29119 CD, 2009).

In the original concept of the RAD model there is the assumption that the general goal and requirements are known. Development is divided into a fixed number of increments (time-boxes). Each increment follows the waterfall model.

The spiral and RAD lifecycle models were designed in order to achieve general goals at the end of the project. In 1969, Lehman proposed an evolutionary view on the development and maintenance of software (1997). Lehman's research results demonstrate the process of continuous evolution and an increase of complexity over time, justifying the need for code refactoring.

### **2.3.3 Agile approach software development**

Agile software development is a modern approach to lifecycle selection for the process (Agile International, 2008). The approach relies on several general assumptions:

- Feedback is preferred over a detailed plan,
- Individuals and interactions are preferred over technical processes,
- Collaboration with the customer is preferred over contract negotiations,
- A fast response to changes is preferred over following the plan,
- A working product is preferred over detailed documentation.

One popular example of the agile development approach is eXtreme Programming (XP) (Beck, 2000). Following this approach, the software is developed by small teams (3-8 persons) building nearly autonomous parts of a software product. Users' representatives take part in the activities of these teams. Software development phases are extremely short (or continuous).

The freedom to change the internal construction of the component as long as its interface is stable requires frequent integrations. These integrations are performed in order to ensure that all parts of the system work together. Continuous integration requires the preparation of automated tests for product components before these components are actually constructed.

The documentation of the system construction following the agile approach is often prepared using refactoring techniques. Software design is the role of the designer-programmer. Users are introduced to an incomplete but functional system.

## **2.4 Quality of goods on the software market**

Products are compared on the basis of their quality characteristics (compare Aristotle definition of quality Kiliński, 1979). It is assumed that software products should be compared on the same basis. However, the set of relevant quality characteristics is the subject of ongoing debate (see below). In this part, the dominant models regarding software products quality and software products quality evaluation processes are reviewed.

### **2.4.1 Quality assessment**

The modern software engineering approach regards software and the use of electronic services as a product (compare ISO/IEC25000, 2005), although some approaches regard software as a service (Turner, et al., 2003). The interdisciplinary research area concerning product quality, quality measures, quality management etc. is still in the developmental stage (Hamrol, et al., 2006).

The quality of a general product is defined using several approaches. The ISO9001 approach states that quality is the conformance with stated and implied requirements for the product (2008). Approaches regarding a certain model or an application area of the product define quality as the ability of the product to satisfy stated and implied user needs in the desired context of use (compare ISO/IEC9126-1, 2001, ISO/IEC25000, 2005, Kiliński, 1979). Quality is also defined as a comparison of the evaluated product with an ideal product (Kitchenham, et al., 1996), or as a set of differences which allow products of the same category to be distinguished (Aristotle's definition Kiliński, 1979).

Other definitions of quality may be found in the Six Sigma model (Pande, et al., 2002), where quality is defined according to the number of defects per million opportunities, as fitness to use (Juran, 1998), or as a two-dimensional model including attractiveness (Kano, et al., 1984).

Definitions of quality in many cases do not relate to precise methods for assessing a quality level based on more than one attribute. The basic normative model for quality perception is based on the simple weighting of a product's attributes (see Wilkie, et al., 1973 for a review). Modern approaches are based mainly on Multiattribute Utility Theory or the Analytic Hierarchic Process (see Wallenius, et al., 2008 for a review). These approaches adopt a simple linear weighting of attribute values with an optional two phase protocol (the first phase verifies minimal values and the second calculates the overall grade). This approach is useful when a quality based decision is to be made by an automatic process (see (Jaeger, et al., 2006) for an example). However, in many cases this approach seems unrealistic: it does

not analyze changes in marginal increases, or the limitations of the person who is making a judgment (e.g. where the person does not possess all relevant information or has insufficient computational power to perform the procedure) (see section 5.2).

The Software Engineering Body of Knowledge provides a set of arguments supporting the statement that software products are specific, and therefore that processes associated with software are different from typical industry processes (IEEE SwEBok, 2004). Some authors suggest that there is no common and objective definition of software quality that is acceptable for both the customer and the producer (Kobyliński, 2005). However, in practice software quality is often expressed only with regard to the number of open failures (especially during acceptance of the product).

#### **2.4.2 First software quality models**

The first software quality model was developed by McCall et al. in 1977 (1977). This model presents a set of desired characteristics consisting of attributes influencing these characteristics. This model was originally designed for software evaluation purposes for the United States Department of Defense (Fenton, et al., 1997). It became the basis for the ISO/IEC 9126:1991 standard (Kobyliński, 2005).

McCall's model is considered to be difficult to use in real projects due to the imprecise nature of the characteristics definitions (Pressman, 2001).

Boehm's software quality model was published one year after McCall's. Both models use similar concepts for quality characteristics. However, Boehm's model defines general quality in terms of general utility (1978). General utility is dependent on as-is utility, maintainability and portability.

#### **2.4.3 The late 20<sup>th</sup> century software quality models**

In 1991, the first international standard regarding software quality was published by the International Standardization Organization and International Electrical Committee. The quality model within this standard defines software quality using six characteristics: functionality, reliability, usability, efficiency, maintainability and portability. The IT industry was looking forward to the establishment of this standard (Bazzana, et al., 1993) due to commonly occurring problems with understanding the software quality concept.

The standard did not meet the IT industry's expectations (Pfleeger, 2001). The main problems with its application were: the limitation of the perspective on software quality to the

producer's perspective, the imprecise measurement definitions, and the lack of a general quality assessment method.

Dromey's software quality model, published in 1994, was one of the most popular software quality models in the 1990's (Kitchenham, et al., 1996). Dromey had proposed a quality perspective independent of the construction method. In his model, each component's quality affected the quality of an upper level in the model of the product's decomposition.

Quality in Dromey's software quality model was considered as an extension of ISO/IEC 9126:1991, distinguishing between the internal and external characteristics. This model analyzes three stakeholder groups: customers, users and administrators (Dromey, 1994). The model describes relations between quality characteristics and stakeholders' focus areas.

#### 2.4.3.1 ISO/IEC 9126:2001

A new version of the international software quality standard was published in 2001. This publication presents a new approach to software quality definition, presenting three perspectives on software quality: internal quality, external quality and quality in use. This model clearly describes the difference between the production process, its quality and the product's quality. The relations between the process's quality and quality perspectives defined in the standard are presented in Figure 2-5.

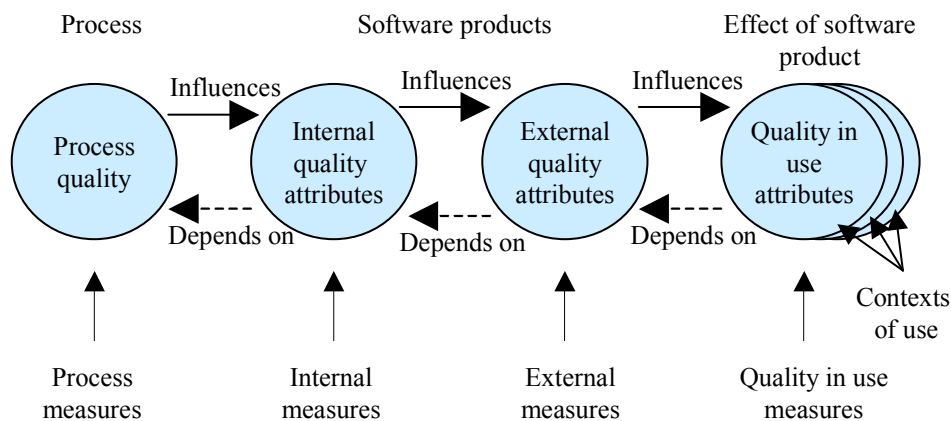


Figure 2-5 Relations between process quality and product quality perspectives (ISO/IEC9126-1, 2001)

Internal and external quality is defined using the same characteristics as those of ISO/IEC 9126:1991: functionality, reliability, usability, efficiency, maintainability and portability. In the new version of this standard, these characteristics (ISO/IEC9126-1, 2001) were expanded to sub-characteristics and a set of measure definitions were provided.

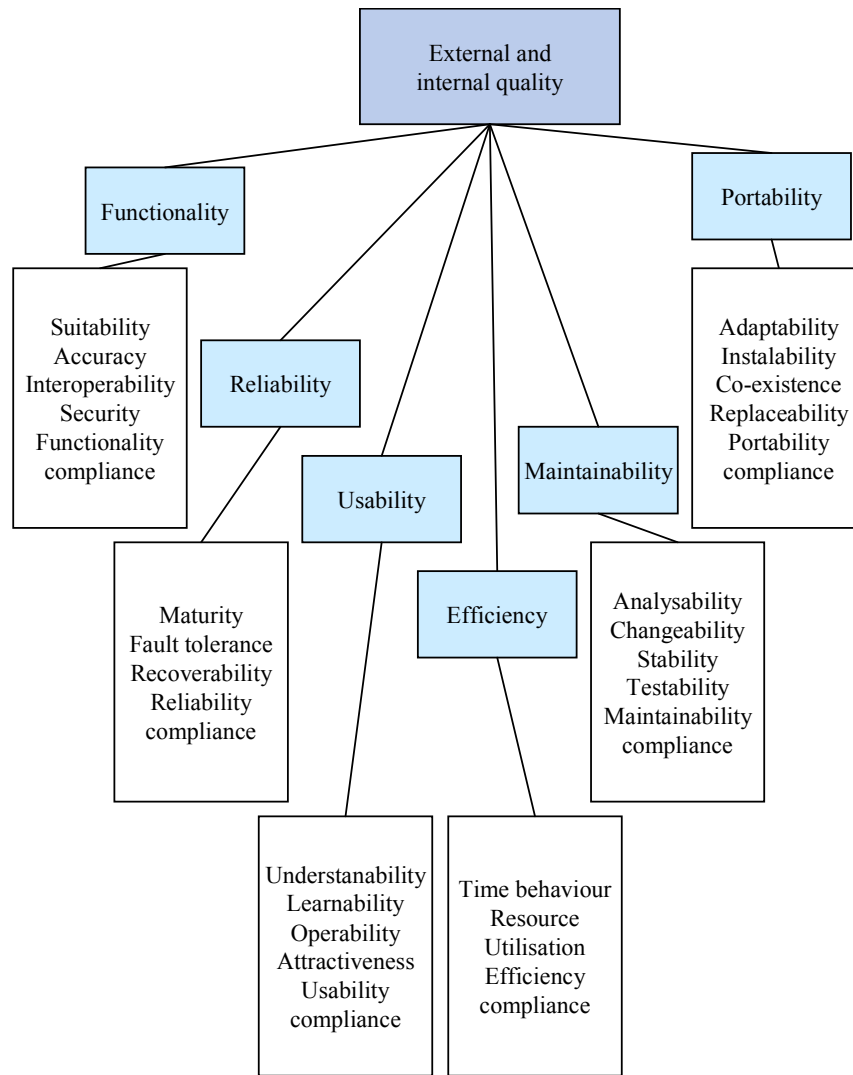


Figure 2-6 Internal and external quality characteristics with sub-characteristics (ISO/IEC9126-1, 2001)

The new perspective on quality representing the user's point of view (quality in use) consists of effectiveness, productivity, safety and satisfaction.

ISO/IEC 9126:2001 is a standard compliant with IEEE1061:1998 (Côté, et al., 2006). The IEEE1061 standard was an answer to the problems of a large number of software quality models presented in literature of the 1990's. This standard defines a meta-model for software quality standards, imposing two requirements: top-down analysis (the software quality model has to allow the decomposition of quality requirements gathered in the early stages of the project), and bottom-up measurements (the software quality model has to allow the measurement of product quality based on the low level measures). On the contrary, ISO/IEC 9126:1991 is an example of a quality model not compliant with IEEE1061 (Pfleeger, 2001).

### 2.4.3.2 ISO/IEC 25000

The ISO/IEC 25000 standards series contain the Software product Quality Requirements and Evaluation (SQuaRE) model. This new approach (Suryan, et al., 2003) extends the ISO/IEC 9126:2001 model, defining the quality model itself, quality requirements, quality measurement, the evaluation process and quality management.

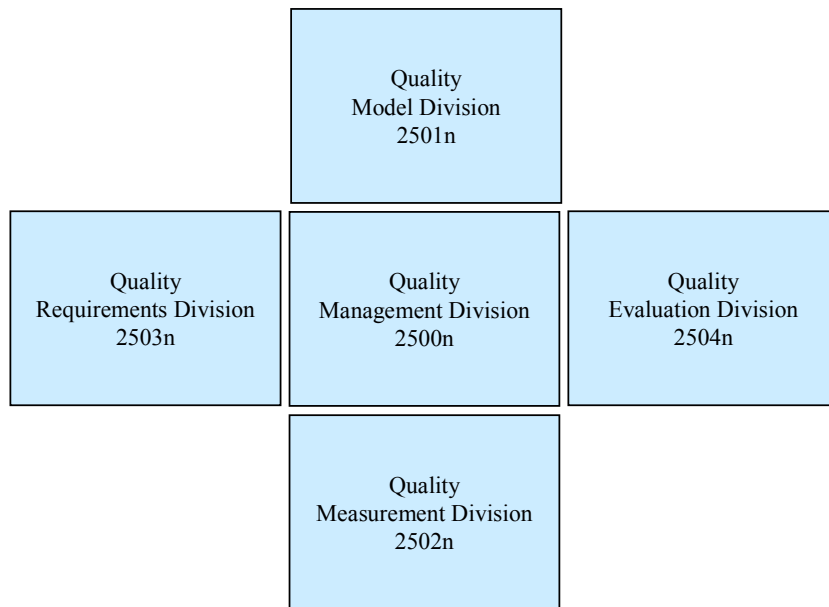


Figure 2-7 SQuaRE model organization (ISO/IEC25000, 2005)

This model is still in the developmental stage. However, in 2011 the software quality model standard is to be published (ISO/IEC 25010 achieved FDIS state in March 2011). The model continues the ISO/IEC 9126:2001 standard achievements in that it proposes three perspectives on software quality: the internal software quality, external software quality and software quality in use. However, the authors have underlined the distinction between software quality and system quality (the distinction was not mentioned in the previous standard). System quality is dependent on the software quality but also depends on hardware, other software products etc. The relation between software and system quality is presented in Figure 2-8.

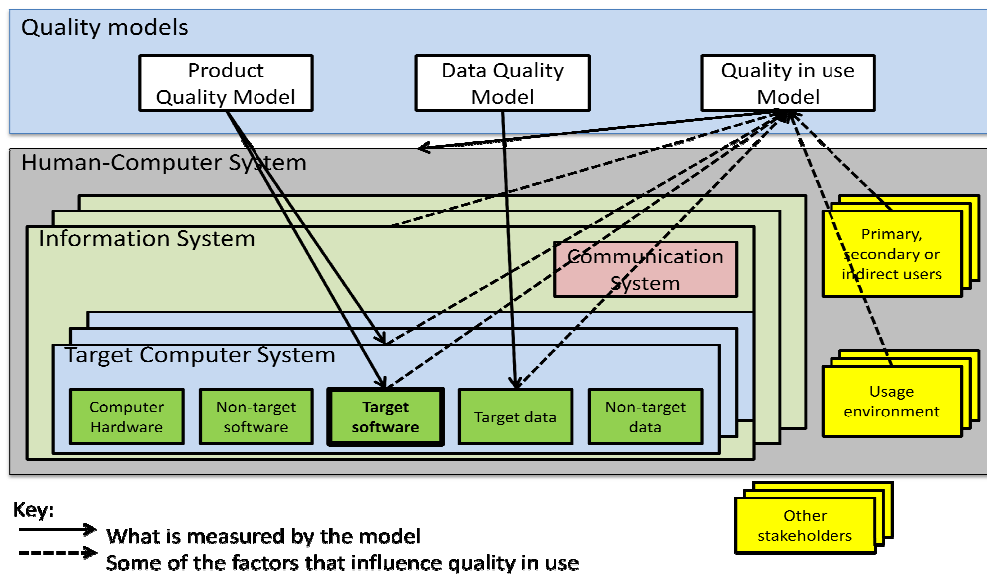


Figure 2-8 Relation between software quality and system quality (ISO/IEC25010 FDIS, 2011)

In the final version of the model, the number of perspectives was reduced from three to two: quality in use and product quality model. The product quality model consists of eight main characteristics: functional suitability, performance efficiency, compatibility, usability, reliability, security, maintainability and portability. This model is presented in Figure 2-9 (see section 3.2 for details of the quality in use perspective).

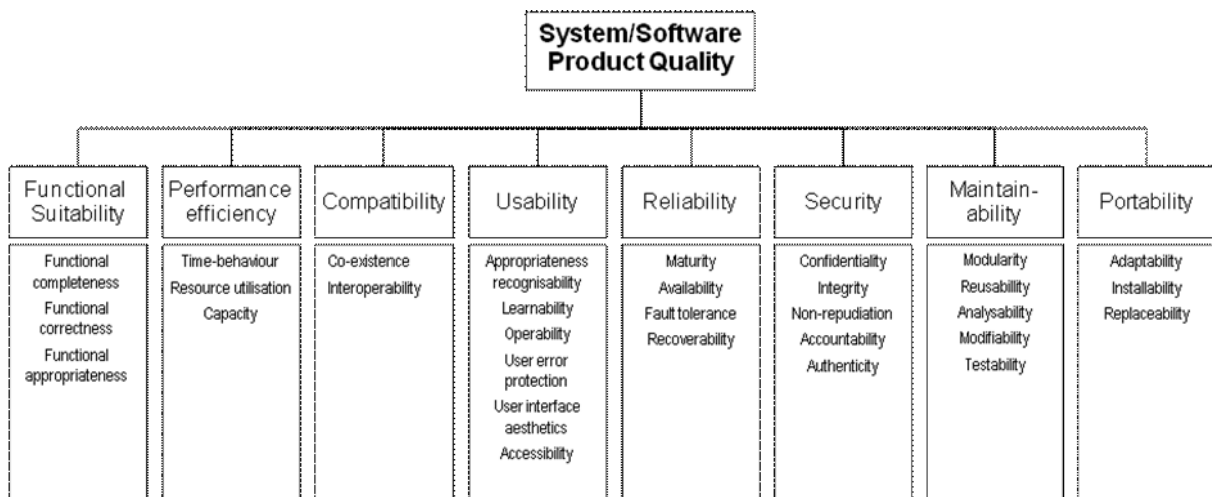


Figure 2-9 Product's quality model (ISO/IEC25010 FDIS, 2011)

The SQuaRE model is compliant with IEEE1061:1998 (IEEE 1061, 1998). This model defines the software quality lifecycle and the relation between general quality, quality characteristics, sub-characteristics and measures. These relations may be used for the



decomposition of software quality requirements in both top-down analysis and bottom-up assessment. The quality lifecycle and the decomposition of software characteristics is presented in Figure 2-10.

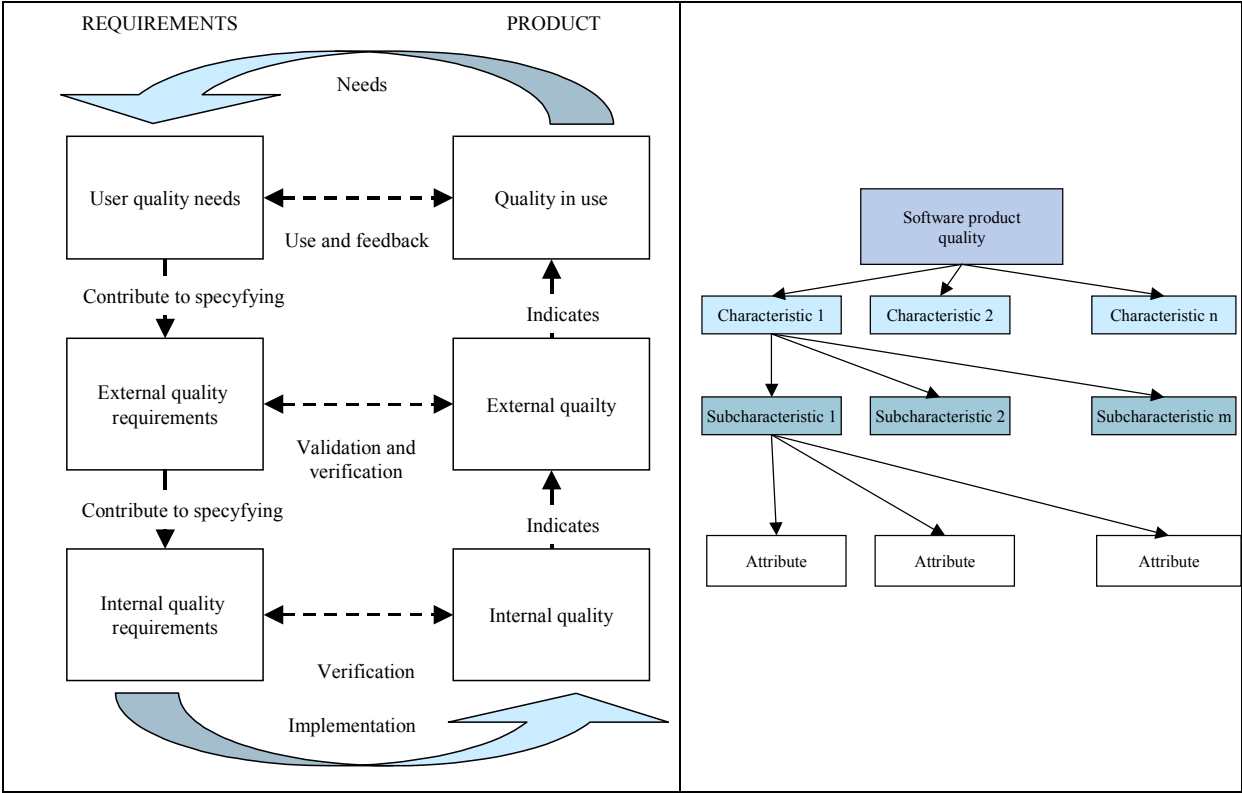


Figure 2-10 a) The software quality lifecycle; b) The decomposition of software quality characteristics (ISO/IEC25000, 2005)

**2.4.4 Electronic services**

Electronic services considered as being products on the software market are commonly named web services and are closely related to the Service Oriented Architecture (SOA) paradigm (Vitvar, et al., 2007) for software product construction and usage. The services provided in the SOA paradigm are said to be the next generation of software products (Iverson, 2004). In this architecture the software product is not owned or maintained by the customer, but is used as a service when needed. From the user’s perspective there is a choice of which analogical Web Services to choose based on predefined (e.g. quality or economical) attributes (Papaioannou, et al., 2006).

Several different quality models for web services may be found in the literature (see Abramowicz, et al., 2009 for a review). Web service quality is not only dependent on the software used for service delivery, but also on hardware, network, service provision

procedures etc. From the user's perspective, the service composition quality is an additional layer impacting upon the quality in use.

Discussion on web services quality highlights two areas related to quality: temporal quality characteristic, and quality of the service provider (Abramowicz, et al., 2008). For the web service user, there are quality characteristics which have to be known regarding a call that will be made. An example of such an attribute is response time, which may be different for certain invokes due to network traffic. Some authors propose algorithms for stressing the impact of the most recent data. However, the problem is in general difficult to solve (Abramowicz, et al., 2008). The need to assess service provider quality is related to result reliability and the security of the data being sent to a web service, although it should be noted that characteristics of the provider also reflect the assessment of past performance, which in general may be out of date.

However, there are several approaches which propose methods for calculating the quality level estimate for the purpose of comparison among different providers (see Abramowicz, et al., 2009 for a review). Proposed approaches are based mainly on the linear weighting function, which is similar to the general view on product quality (see Wallenius, et al., 2008 for a review). This method assumes that there are defined minimal acceptable values of attributes, and that when the minimum is not violated then the decision is based on the weighted sum of attribute values (see Jaeger, et al., 2006, Vu, et al., 2005) assuming that historical values offer the best means of best prediction.

## **2.5 Evaluation of products on the software market**

In this part, attention is focused on the quality assessment process. Beginning from the general perspective on product valuation, a short review of economics concepts related to this area is presented. The second part of this section is devoted to the evaluation processes related to software products.

### **2.5.1 Classic approach value of goods**

The value of goods is one of the basic ideas in human perception related processes. Sociological, psychological, economical and ethical value models have been developed since ancient times. In a typical approach, the value of an object is often described as being equivalent to the object's price (dependent on supply and demand in a competitive or non-competitive market) (Grossman, et al., 1976).

In classical economics, the value of goods is considered as being equivalent to some other goods used for production or substitution. For example the labor equivalent is the discomfort of a worker spent to produce a good (Case, et al., 1999). In this approach, the value of an object is not equal to its price because it is not dependent on the situation in the market (“natural price” in (Smith, 1776) or “prices of production” in works by Karl Marx described by Rubel, 1975).

Value has also been perceived as the usability value measured in terms of the benefits provided to an object’s owner. Ludwig von Mises (Paul, 1984) describes value using an association with a utility following the consumption of a good or use of a service.

One of the classic examples of the subjectiveness of value is the diamonds and water paradox described in the works of Adam Smith (1776). The question in this paradox uses the observation that although water is crucial to human survival while diamonds are useless from a biological point of view, diamonds are more expensive than water. In the 19<sup>th</sup> century, Herman Gossen described the law of diminishing marginal utility (1854). This law defines the relation between the subjective value of an object and the level of adequate need saturation. Friderch von Wieser (1889) suggests that satisfied needs are of less importance than unsatisfied ones.

In the 20<sup>th</sup> century, Ayn Rand formulated the objectivist theory (Rasmussen, 1990). This theory asserts that reality is independent from human perception. Reality is an objective term and it has unchangeable attributes, which may be learned and known by humans (these properties are said to be intrinsic to reality). Immanuel Kant argued that humans adopt *a priori* concepts and knowledge. After the mind is set, the observer perceives their state of mind instead of the real attributes of the observed reality (Haden, et al., 1981). Kant’s theory is supported by the results of modern experiments (Nęcka, et al., 2008). David Hume has analyzed the sources of concepts in human minds, and formed the thesis that people tend to reject observations that stand outside of other observations or their beliefs (Stroud, 1977). Hume’s observations are confirmed in the cognitive dissonance theory (Festinger, 1957).

Value can be analyzed in terms of utility. The first remarkable definitions of value come from utilitarians such as Jeremy Bentham and John Stuart Mill. However, their contribution is perceived as having strong psychological underpinnings (Anand, 2002). Neoclassical economists typically do not use cardinal utility models, which capture an artificial value of utility allowing for the comparison of magnitude. Instead, economists use the ordinal utility definition, which defines only the ranking relation associated with agent preferences over a choice set (Basu, et al., 1992). These preferences are assumed to be constant and may be used

for the purpose of constructing indifference curves, which are said to represent equal utility values for consumers (List, 2004). The typical assumptions regarding preferences are: a greater amount of a good is preferred to a smaller amount, increasing the consumption of one good results in the declining consumption of another good, and indifference curves do not intersect (Holzman, 1958). The overall value of a choice set is typically regarded as the sum of utilities associated with components of this set. Multi-attributed products are considered as they consist of a set of independent utility functions related to attributes (compare Keeney, 1977).

Modern approaches to the valuation of goods regarding perception limitations are described in sections 4 and 5.2.

### **2.5.2 Software market products evaluation process**

The evaluation of products on the software market is a process performed to assess the characteristics of the product and assign them a value. The software engineering approach suggests that a mature evaluation process should be: repeatable, reproducible, impartial and objective (ISO/IEC14598-5, 1998).

The evaluation of a software product is typically performed with the employment of software testing, and serves mainly the purposes of software construction, software acquisition and independent evaluation (ISO/IEC14598, 1999). The first stage of the evaluation process is the establishment of evaluation requirements. For the purposes of software development and independent evaluation the source of requirements is the product documentation. From the acquirer's perspective there are several sources of evaluation requirements.

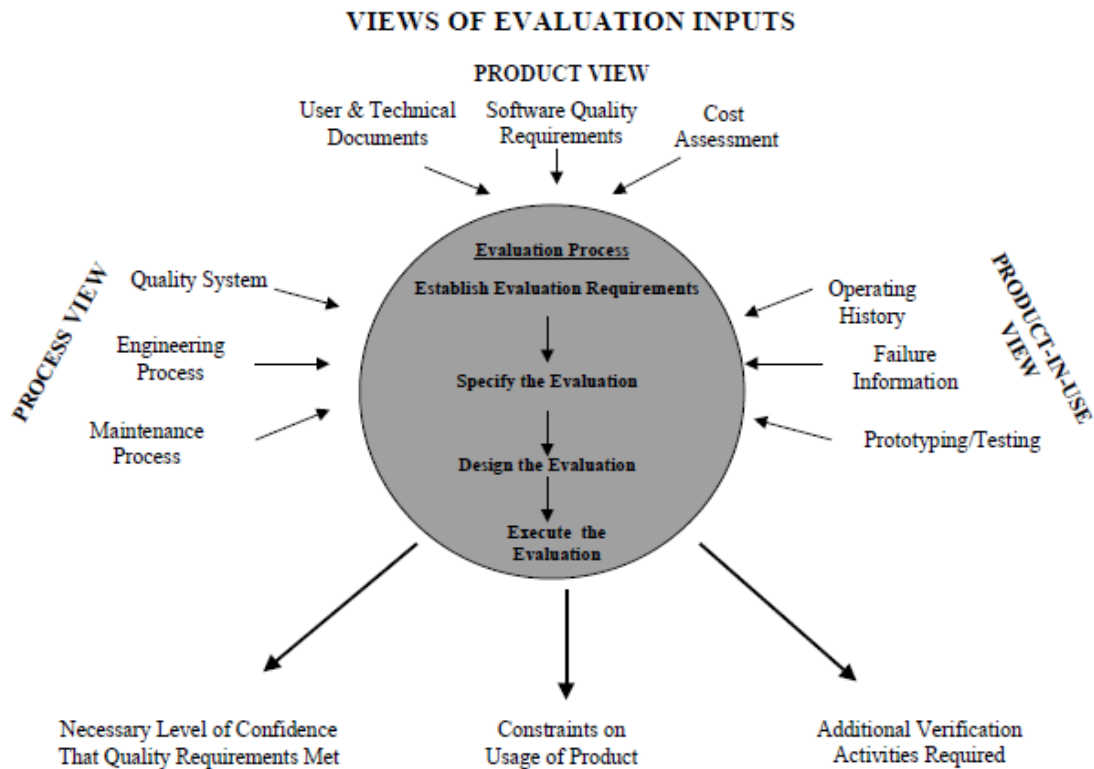


Figure 2-11 Software product evaluation process overview from acquirer’s perspective (ISO/IEC14598-4, 1999)

The stages of the evaluation process are: the specification of the evaluation, the design of the evaluation and the execution of the evaluation. After the execution of the evaluation, the conclusion stage occurs, which is different for discussed perspectives (a producer’s perspective or a customer’s perspective).

Those involved in the evaluation process should be supplied with a description of the expected levels of chosen software quality characteristics (ISO/IEC14598-4, 1999). The authors of the standard state that the set of measures is to be unambiguous to ensure the objective result of the evaluation process. The standard itself does not contain further techniques or methods to ensure the objectivity of the evaluation process.

Elicited evaluation requirements and software quality requirements are used for the design of the evaluation (ISO/IEC14598, 1999). The evaluation is performed mainly with the use of static or dynamic techniques. According to the definition of verification and validation, the evaluation tasks are expected to confirm and provide objective evidence so that a requirement or a user need is satisfied.

An objective approach is not the only approach described in the literature. Szajna (1994) describes the evaluation of software in terms of subjective results. The results and the

assessed software usability are expressed in terms of their perceived usefulness and ease of use (U/EOU). This research is based on the Technical Acceptance Model (Davis, 1989). The author analyzes and assesses users' intentions based on subjectively perceived software attributes.

The evaluation of web services enhances the difficulty of evaluating software products. Web services have invariant quality attributes (typically Quality of Response, transactional parameters, financial parameters etc.) and temporal quality attributes (availability, response time characteristics etc.) (Abramowicz, et al., 2009).

A complete definition of web services quality and an evaluation model based on the SQuaRE model was proposed by Abramowicz et al. (2008). The authors present multilayered relations between responsibilities associated with the provision of a web service (presented in Figure 2-12). The authors also present an extensive literature review regarding the quality model concept for web services.

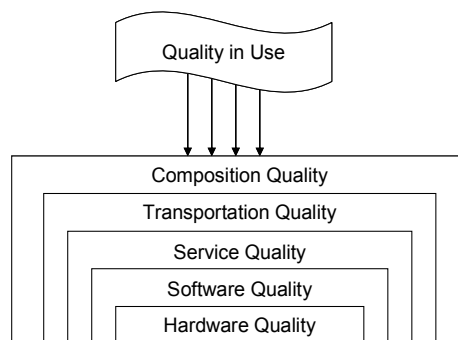


Figure 2-12 Quality and responsibility layers for a web service (Abramowicz, et al., 2008)

The quality levels for a web service emphasize the problem of evaluating quality according to a use perspective. The user of a web service assesses the risks associated with usage typically as a *caveat emptor* (roman trade rule: let the buyer beware). Users are unlikely to gather objective knowledge regarding the web service's quality despite the problem with temporal characteristics.

In the literature, concepts mitigating the above problem have been described:

- Adopting an *a priori* approach (Ran, 2003) – one assumes that quality declarations are trustworthy
- Using a certifier (Ran, 2003) or RES agency (Maximilien, et al., 2001) – one assumes that a certification service is trustworthy
- Using a service broker (Kalepu, et al., 2004) (Tian, et al., 2004) – one assumes that the brokering service is trustworthy,

- Adopting an *a posteriori* approach, based on user feedback (Ran, 2003) – one assumes that other users' opinions are trustworthy

Other approaches are: Q-components (Menasce, 2004), QUEST (Gu, et al., 2003), agent-based (Kokash, et al., 2006), OGSA (Sheth, et al., 2002), OASIS (2005) etc. All of the above approaches assume the trust of a party which is based on collected evidence (Wang, et al., 2007). The debate regarding these approaches remains open (Abramowicz, et al., 2008).

### 2.5.3 Participants in the software market

Typically, market participants are divided into sellers and buyers. Some approaches identify other participants, like regulators, organizations representing the interests of a larger number of participants etc. (Sawyer, 2001). Participants in the software market may also be divided into these categories.

Software sellers are typically divided into three sub-categories: producers, integrators and vendors (Davies, et al., 2007). However, considering electronic services as part of the software market requires the establishment of another category: providers. The ultimate goal of these four groups is in most cases similar: to maximize their profit from the market.

A special market situation occurs when the software producer or provider and the customer belong to the same organization. In this case, it is difficult to assume that the goal of the seller is to maximize their profit because in such situations it is unlikely that there will be internal profit transfers. Additionally, internal IT departments are assumed to be only the source of costs (IT Service Management Forum, 2007).

However, this does not mean that the product should be delivered on time, within the agreed budget and with high quality (some analyses show that the profits related to the maintenance of products with low quality are a source of significant revenue Cusumano, 2008). This problem seems to be a good example of a lemon market (Akerlof, 1970). In 1970, Akerlof showed that the consequence of information asymmetry in the market leads sellers to deliberately reduce product quality along with increasing the prices. Considering the problem of hidden attributes of software products (discussed in section 2.1), it may be concluded that the problem of the deliberate approach described by Akerlof exists in the software market.

Software buyers are typically divided into individual customers and corporate customers. Another distinction is made to separate customers (who make decisions about purchase) from users of the product (IT Service Management Forum, 2007). A distinction may be also made on the basis of the purpose for which the product is purchased: business customers acquire products for business purposes, typically satisfying the need of some company; however, a

large part of the market consists of entertainment products, typically satisfying individual needs (Björk, et al., 2002).

The above distinction is associated with the dominant model of electronic services provision. Most electronic services are free of charge (Anderson, 2007). However, users decide which service to use, and the provider benefits if their service is chosen (for example, for the purpose of advertisement income). Later in this dissertation the term user will be used to denote the person who decides whether or not to use the software product regardless of the fact of its purchase.

The ultimate goal of customers and users is to satisfy a set of needs (ISO/IEC25000, 2005). This goal seems to be in opposition to the seller's perspective of the market, especially for large contracts where the seller does not expect to obtain another order from the same customer.



### 3 CUSTOMER DECISION THEORY

In this chapter, customer decision theories are reviewed. In the first part, the general approach to decision theory and customer decision theory is described, and in the second part the aspects related to product quality assessment for the purpose of decision is reviewed.

#### 3.1 Decision theory overview

Decision making is one of the most common activities among all living organisms, including humans. Processes of selecting a path, choosing a prey, or selecting a financial plan for retirement all involve decision making. More formally, decision making is a process of identifying values and uncertainties, options and boundaries, and selecting an optimal decision (Keeney, et al., 1993). An alternative definition regards decision making as a cognitive process of selection among a set of alternative actions (Doyle, et al., 1999).

The modern approach to decision theory originated in the 1940's through research in several disciplines (economics, psychology, sociology etc.) (Hansson, 2005). One of the most important approaches within the decision theory discipline is the distinction between normative and descriptive approaches (Stanovich, et al., 1999). The normative approach analyzes optimal decisions to be made, and the descriptive approach analyzes decisions that are actually taken.

The earliest decision theory dates from the 18<sup>th</sup> century and is attributed to Condorcet, who divided the process into three stages (Hansson, 2005). The first stage is the identification of alternatives, in the second stage options are reduced to a smaller set of more general ones, and in the third step the actual decision is made. Modern decision theories originate from John Dewey, who published his theory for individual decision making processes ([1910] 1978), and Herbert Simon (1960), who modified Dewey's theory to suit the context of an organization. Another significant contribution to decision theories was made by Brim in 1962, who included personality and social context in the decision making process (Hansson, 2005).

The aforementioned decision theories assume that the decision making process involves a sequential list of stages. In opposition, non-sequential theories were developed in 1976 by Mintzberg, Raisinghani and Théorêt (1976). The main idea of these non-sequential models is the parallelism of the decision process phases. However, the main objectives remain the same.

Decision theories mark out the phases of information gathering, and the preparation of the final decision or opinion. The valuation of alternatives in normative theories follows

assumptions of the completeness and transitivity of valuations. By employing neoclassical assumptions regarding the maximization objective, the highest valued alternative should be selected (Hansson, 2005).

The analysis of optimal choices requires assumptions regarding what optimal means. The classic approach to economic decision making assumes that the agent is rational, utility (or profit) maximizing, and in possession of complete knowledge about the consequences of each decision (Simon, 1979). These assumptions follow the classic idea of *homo economicus* (Smith, 1776).

The review of concepts regarding Bentham's utility concept was described in section 2.5.1. Regarding agents' preferences, Paul Samuelson has proposed the Revealed Preference Theory (1937). According to this theory, agents' preferences are manifested by transactions performed. However, it is assumed that preferences are constant and transitive. Moreover, Stigler and Becker (1977) argue that preferences are not only stable, but also identical for all people, providing a model of "habit formation" (compare Duesenberry, 1952). Using theories of stable preferences, economists may draw indifference curves said to represent equal preference choices. The preferences should remain independent from current consumer entitlements (List, 2004). Ordinalism has also had methodological implications. Post World's War II neoclassical economists assumed that the most reliable method for collecting information about preference was the study of actual transactions or other observable choices (Angner, et al., 2007), therefore omitting the study of immeasurable feelings (Camerer, et al., 2005). The ordinalist approach was criticized by institutional economists (Lewin, 1996). However, mainstream economists responded by arguing that economics is independent from psychological assumptions, and that described behavior may always be rationalized by some preferences regardless of psychological underpinnings.

An alternative perspective was presented by Robbins, who pointed out that preferences cannot be identified with actual choices, but are closely linked (1932). The link between choices and preferences is said to have the same nature, and the choice of data may be used to infer preference orderings (Angner, et al., 2007).

Another distinction within decision theory is related to the risk and uncertainty associated with the consequences of action (Knight, 1921). Some researchers suggest that because each decision is associated with some risk or uncertainty, mainstream economics should be associated with decision making under risk (Rick, et al., 2008). In Knight's description, the difference lies in the knowledge of probability. Decisions made under risk and uncertainty are typically viewed as choices between prospects or gambles (Camerer, et al., 2003). In the

lottery game described by Nicolas Bernoulli, the expected value is infinite. However, it is unlikely that a real person will enter the game if the price exceeds some value. The solution to Bernoulli's paradox (known as the St. Petersburg paradox Bernoulli, [1738] 1954) is considered to be the beginning of Expected Utility (EU) theory (Kahneman, 2003).

The main assumptions of EU theory use a linear weighting approach to the utilities of each outcome and their probabilities. Von Neumann and Morgenstern (1944) have proposed a set of intuitive axioms regarding completeness, transitivity, continuity and the independence of preferences. Their contribution was considered as indicating the acceptance of EU as part of normative theory. However, it was also considered as the descriptive model of behavior, as decision makers were said to base their choices on EU rather than the expected value (Schoemaker, 1982), (Rick, et al., 2008). The decision processes described by Von Neumann and Morgenstern are based on the assumption of rational choices (represented by axioms) and complete information (Hastie, et al., 2001). The learning process is often regarded as a Bayesian updating of probabilities (especially regarding uncertainty conditions) with consistent beliefs and preferences (Gilboa, et al., 1995). The space of rational preferences over uncertain outcomes is isomorphic to quantitative representation in terms of the EU model according to Savage's representation theorem (1972).

Empirical research has shown, however, that important violations of the EU model exist (see Starmer, 2004 for a review). Considering a gamble where there is a 50% probability of winning \$100 and 50% probability of losing \$100, EU assumes that the likelihood of accepting such a gamble is based on final states. This means that if a decision maker owns \$1,000,000 then their analysis of the gamble is based on the following consideration: a 50% chance of having \$999,900 or a 50% chance of having \$1,000,100. Other examples include the famous problem of the sixth egg added to an omelet (Savage, 1954) or the Allais paradox (Allais, 1953).

EU theory was modified in response to empirical evidence: the assumption regarding the analysis of final states was abandoned (Markowitz, 1952), and the model was enhanced by introducing models of counterfactual emotions influencing decision making (see Mellers, et al., 1997), or via the observation that probabilities are weighted non-linearly but in relation to potential consequences (e.g. Edwards, 1953).

A rational-based model of decision making is also extended in some approaches with reference to attitudes, intentions, preferences, subjective norms etc. An example of such a theory is the Theory of Reasoned Actions (Hale, et al., 2003).

Savage (1954) has proposed a model of Subjective Expected Utility (SEU) where the decision maker analyzes their beliefs regarding the probabilities of choosing the action which maximizes pleasure and avoids pain. SEU addressed the problem of unknown probabilities and decisions under uncertainty, and was said to be a normative model of behavior which could also be regarded as descriptive model. The subjective probability concept was later formalized by Anscombe and Aumann (1963). However, criticism of SEU appeared shortly afterwards. Ellsberg (1961) pointed out that subjective probabilities are used not only for the estimation of occurrence likelihood, but also for decision weights associated with utilities. Consequently, probabilities cannot be unambiguously estimated, and some options may be overweighed. Ellsberg has demonstrated this phenomenon by formulating a famous paradox: people prefer betting on a lottery with a 50% probability of winning to betting on a lottery with unknown probabilities. Several theories attempted to remove unambiguity from SEU (see Camerer, et al., 1992 for a review).

In the 1950's, Milton Friedman stated that the realism of theory should not be regarded as more important than the validity of predictions about future states (1953). Three years later, Herbert Simon published his influential research results on bounded rationality (1956). Simon's model was based on a purely descriptive approach. His conclusions stated that people are rational, but because of the complexity of decision choices analysis, they are able to use only a simplified method of evaluation. The strategy of selecting the choice which fits preferences, even if it may be not optimal, was called the satisfier strategy. Another challenge for the neoclassical approach to economic human was the theory of selective rationality proposed by Leibenstein (1966). The author rejects the strict calculative concept, suggesting that selecting an action results from the appropriate combination of awareness of boundaries and internal and external pressure. Leibenstein refers also to the duality of human nature, formulating an explanation based on Freud's concepts of id and the superego (1954). The concept was developed as a result of psychological research (see Evans, 2008 for a review), and has been accepted by economists since the work of Thaler and Shefrin (1981), who proposed the idea of duality in terms of personality split: between myopic doers and farsighted planners.

In the 1970's, Daniel Kahneman and Amos Tversky published their prospect theory (1979), which is a variant of SEU. Prospect theory makes no normative assumptions, and distinguishes between two general phases of the decision making process. In the first phase (editing phase), a decider reduces alternatives to the gains and losses associated with them in relation to a subjective reference point. The reference point is often considered as being the

current state, but may be a result of expectations or an analysis of alternatives (compare Köszegi, et al., 2006). In the second phase (evaluation phase), a decision maker assigns subjective values to alternatives using gains and losses assessment. Their research results show that people tend to overweigh small probabilities and underestimate large ones, and tend to treat losses as being more unlikely than comparable gains. Their empirical observations follow Adam Smith’s findings, regarding positive-negative asymmetry, expressed in *The theory of moral sentiment* (1759). A similar observation was made by Quiggin (1982), who reported that people overreact to the best and worst possible outcome.

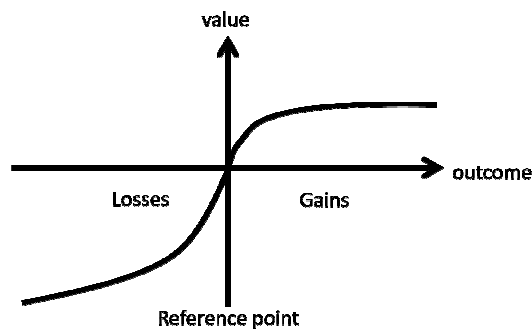


Figure 3-1 A typical gains and losses valuation function (Kahneman, et al., 1979)

The works of Simon, Kahneman and Tversky refute the assumption that preferences are stable regardless of the current entitlements of the decision maker (1990), and are considered as the beginning of a new discipline in economics (Camerer, et al., 2003). However, the term “behavioral economics” has been used since 1958 (Angner, et al., 2007). Research regarding the behavioral aspect of decision agents, as well as heuristics for the evaluation of decision options and systematic biases, has become important to the analysis of actual human behavior and to understanding the actual patterns of decision making. Preference changes and preference reversal have been studied by several researchers (see Rakow, et al., 2010 for a review). In this approach, decision making is regarded as a cognitive process resulting in an outcome that leads to the selection of a certain prospect among alternatives (Rakow, et al., 2010).

An example of the difference between normative and descriptive theories, which is important in terms of decisions with postponed results, is related to the idea of discounted utility (DU) (Samuelson, 1937). Similarly to EU and SEU, discounted utility theory proposes a normative approach and an intuitive set of axioms (Koopmans, 1960). DU assumes that utility discounting is a function of time, and results in the same discount rate for each time period. This means that the expectation of an expected gain of \$1,000,000 is the same if one

considers gaining it in one or two years or in 101 or 102 years. Senior, Jevons and Bohm-Baverek were amongst the first to publish their concept of intertemporal choices. These authors assumed that choices over time follow exactly the same rules as choices made for current decisions (compare Loewenstein, 1992).

A lot of empirical evidence questioned the predictive accuracy of the DU model (see Frederick, et al., 2002 for a review). Strotz has pointed out that a decision maker tends to make far-sighted decisions when the consequences are delayed in time and short-sighted ones when the consequences are immediate (1955). Several experiments and observations involving real money have supported the idea of declining discount rates (Horowitz, 1992). Consequently, several researchers proposed a hyperbolic time discounting utility model (e.g. Loewenstein and Prelec 1992) and quasi-hyperbolic models (e.g. Laibson, 1997). An interesting anomaly was reported by Loewenstein, who observed that in certain situations people prefer to obtain the negative outcome as soon as possible while they tend to postpone pleasure for the future (1987). Similarly, Read et al. have shown that differences result from the way the delay is described (2005).

Psychological insights into economics decisions were also criticized. Perhaps the most famous attempt is that of Grether and Plott (1979). The authors intended to discredit thirteen chosen theories based on the psychological approach by showing their irrelevance for economic decision making. They failed in their endeavor, concluding that empirical results do not fit normative models. In following years, researchers have shown situations where observed effects have been diminished (refer to List, 2004) or reversed (refer to Lerner, et al., 2004).

The models discussed above consider the situation where the decision making process does not affect the options (in contrast to situations where other participants in the market may influence the future state of the world, which is subject to game theory, Aumann, 2006). The problem with this assumption is shown in Newcomb's problem (Nozick, 1969). The problem underlines two important aspects of difference between normative and actual behavior: the first is the assumption about utility maximization, and the second is the problem of decider intentions affecting choice options. Similar decision problems, such as those described in *Death in Damascus* published by Gibbard and Harper ([1978] 1981), show the limitations of typical decision making approaches. Gibbard and Harper also published Stalnaker's resolution to Newcomb's problem, and distinguished causal decision theory (CDT) from evidential decision theory (EDT), presenting two concepts on how beliefs and objectives may be used in the decision making process.

Evidential decision theory (EDT) uses conditional probabilities regarding the state of the world resulting from a certain decision. It is relatively easy to construct an example of a decision problem to which EDT can be applied to suggest incorrect decision making (e.g. assuming that there is a cause C which may imply A or B, there is high probability of a co-occurrence of A and B; if an agent prefers A but dislikes B and is to make decision about A, then the agents' preferences are:  $A \wedge \neg B$  is preferred to  $\neg A \wedge \neg B$  and  $A \wedge B$  is preferred to  $\neg A \wedge B$ ; while it seems natural to choose A, EDT suggests that the agent will decide  $\neg A$  because of the high probability of A's co-occurrence of B, although B does not depend on A).

In contrast, causal decision theory (CDT) uses subjective unconditional agent beliefs in dependencies. CDT would suggest that the agent would decide to choose A in the above example. However, an example of a problem where CDT fails may also be shown (Egan, 2007). Egan modifies Newcomb's original problem to Newcomb's firebomb problem, where the agent is to decide between getting \$1,000,000 for sure, or taking a risk of getting \$1,001,000 if the predictor predicted that the agent would take the first option or \$0 if the predictor predicted that the agent would take the second option (the assumption is that the predictor is accurate).

Both EDT and CDT may be modified by ratificationist theory (Egan, 2007), which states that only ratifiable options may be chosen. This helps to solve some decision problems where these theories fail to indicate the optimal decision. However, in Newcomb's firebomb problem both options are unratable, thus CDT is unable to suggest the optimal choice.

Another decision theory area is known as consumer decision making. Consumer behavior theory covers actions directly related to acquiring, consuming and disposing of goods (Engel, et al., 1995). An important part of this area is related to consumer decision making processes, which include precedence, selection and following up on decisions related to the acquisition of goods satisfying certain needs, as well as changes of feelings and attitudes toward products and recognition of the moods of consumers (Schiffman, et al., 2000). The best known consumer decision making models were developed in the 1960's and 1970's by Howard (1963), Andreasson (1965), Nicosia (1966), Engel, Kollat and Blackwell (1968), Markin (1968), Howard and Sheth (1969) and Hansen (1972). These classical models describe the decision making process in terms of a logical problem solving approach (Cherian, et al., 1990), and divide the process into a five step classification: problem recognition, information acquiring, evaluation of alternative choices, choice and evaluation of an outcome (Schiffman, et al., 2000). Sometimes these steps are enhanced with additional steps. However, these five are in the central interest area for this dissertation (Engel, et al., 1995).

The above approaches incorporate the rational approach to decision making and outcome value maximization by customers (Solomon, 2006). An outcome may be measured in terms of price, quality, functional adequacy etc. (Schiffman, et al., 2000). Logical positivism, which dominated in this area at the beginning, resulted in a more rigorous approach to discover and generalize laws of consumer behavior (Engel, et al., 1995). However, revisions of the “grand-models” (242) were (Kassarjian, 1982) based on the cognitive approach (eg. Engel, Kollat and Blackwell’s model 1982).

Consumer decision making models have been criticized since their inception. The main concerns are related to the assumption regarding the rationality of consumers, observations that consumer behavior is in many cases non-conscious, the lack of emotional perspective and regard to social context, the sequential character of the process etc. (Erasmus, et al., 2010). Bettman (1993) pointed out the need to concentrate on the meaning of products for customers; Olshavsky and Granbois have pointed out that information previously gained plays a key role for the customer (1979); while others, such as Cox et al., have called for an improvement of the descriptive power of the models (1983). The theory therefore failed to cover all types of customer decision processes. Even the authors of the models admitted that their models reflect buyer rather than customer decision making processes (Firat, 1985).

Thaler’s contribution to behavioral economics began with his positive theory of consumer choice (1980). Thaler described sunk cost effect, considerations of regret and other empirically observable biases. In his later work, he proposed a new model of consumer choice combining cognitive psychology and microeconomics (1985).

Bettman (1998) argues that consumers do not analyze holistic quality as an overall value, although they perceive attributes which are combined with the use of the conjunctive or disjunctive model. In the context of software product related processes, it should be assumed that their attributes are combined with the use of the conjunctive model (Sagan, 2004). An important note should be made in regard to a situation when the decision maker lacks important information. According to Burke (1995), the information gap forces the decision maker to compare the product with others based on their experience and technical knowledge. These approaches are more likely to be used when the decision maker faces a product with which they have less experience (Solomon, 2006).

Bettman suggests also that the decision maker optimizes not only the decision itself but also the cognitive effort related to the decision making process (1993). This observation reflects Simon’s theory related to the satisfier’s strategy.



Decision theory is thus concerned mainly with individual decision making processes (Maccheroni, et al., 2008). However, most economic activity is performed by employees, who make decisions in favor of the employer (Simon, 1956). Decisions in organizations were studied by Simon (1979). The decision making process within organizations is influenced by the individual decision making process. However, it is also influenced by group dynamics (Aronson, et al., 1994) (Akerlof, et al., 2005), friends and enemies (Camerer, et al., 2007) or political bargains between participants (Thompson, 1995). The idea that the welfare and consumption comparison with others influences an agent's overall utility is attributed to Veblen (1899). This idea was enhanced by Festinger (1954), who took this theory of self-evaluation (mainly in the area of opinions) and later extended it to other areas, such as the evaluation of happiness (Strack, et al., 1990) or income (Brandstätter, 2000).

Topics within decision theory, representing theoretical foundations in this area as perceived through the perspective of contemporary research results, are discussed below in section 4.

### **3.2 Quality perception for the purpose of decision making**

The perception of software quality may be considered from the perspective of the seller (producer) or buyer. Quality assessment serves different purposes for these two participant types: sellers are willing to use the model for the purpose of deliberate software quality management, while buyers use quality assessment in deciding whether or not to purchase or use software (see discussion in section 2.5.3). This dissertation focuses on the buyer's perspective, as the economics of software production is beyond its scope.

The SQuaRE quality model (ISO/IEC25010 FDIS, 2011) describes the customer's perspective through the definition of quality in use. It assumes that the user's and customer's perspective on a product's quality are equal (for simplification, this assumption will be used henceforth). Software quality in use and system quality in use are defined as the extent to which the product satisfied stated and implied user needs when used in a certain context (ISO/IEC25000, 2005). Quality is defined through a set of characteristics which are decomposed to sub-characteristics. The current version of these decompositions is presented in Figure 3-2.

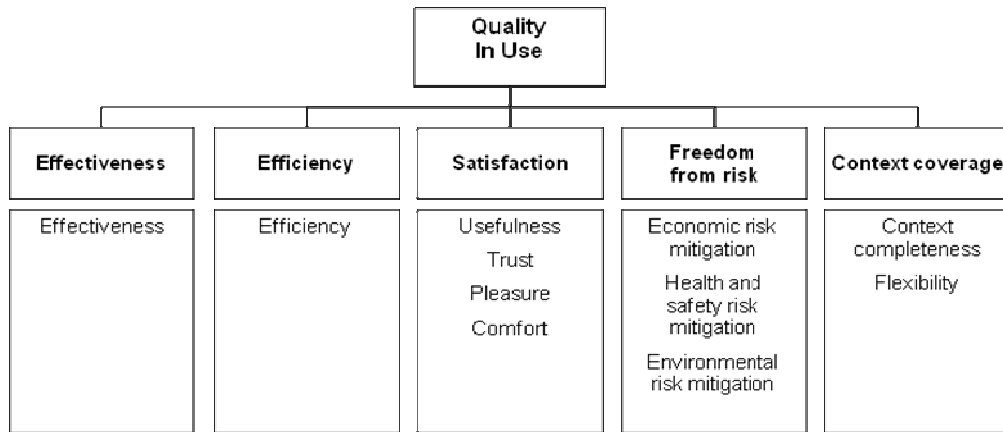


Figure 3-2 The system quality in use perspective (ISO/IEC25010 FDIS, 2011)

According to this model, the sub-characteristics will have the attributes and quality measures defined. With the use of ISO/IEC 9126:2001, (ISO/IEC9126-1, 2001) the sub-characteristics may be described by objectively measurable and subjective characteristics.

Satisfaction is the most subjective characteristic. However, among the remaining characteristics there are also subjectively measurable ones (e.g. related to risks, trust, efficiency, or usability). In (ISO/IEC9241-11, 1998), usability is defined as comprising effectiveness, efficiency and satisfaction. Nielsen extends this definition to include learnability, memorability and the number of user errors (2003). Therefore, the complete perspective of system quality in use may be perceived as being usability oriented. Krawczyk-Bryłka (2000) underlines that previous experience, as well as personal and sociological factors, influence the perceived quality, which is not included in dominant models of the area.

The usage of a new system is, from a user's perspective, associated with the process of adopting a new tool. In the Technology Acceptance Model (Davis, 1989), the author defines perceived usefulness (U) and the ease of use (EOU) for this process. Davis's research is based on the sociological model regarding the theory of reasoned action proposed by Fishbein and Ajzen (1980). An example of a similar approach to software usefulness is presented by Szajna (1994).

Another approach, commonly used at present, was proposed in 1984 by Grönroos. In this approach, the author perceives quality as being a function of expectations. Grönroos divides perception into three dimensions: functional, technical and image (perception of the brand) (1984). These dimensions form the basis for the SERVQUAL model (Parasuraman, et al., 1985). This model, and its successors, are widely used quality perception models (Kang, et al., 2002), not only for software products but also for airline services, fast-food,

telecommunications, banking, physiotherapy, web sites, healthcare and many others (Babulak, et al., 2002), (Miguel, et al., 2007).

An approach based on belief revision theory (Peppas, et al., 1995) was proposed as another model of user perception of software quality. This method adopts the AGM paradigm (Alchourron, et al., 1985) or an alternative definition Grove's system of spheres approach (1988). It proposes an epistemological approach to define beliefs and their revision processes, following the assumption that the observer is a rational, deductive agent using the principle of minimal change.

A perception model based on the above approach was proposed by Xenos et al. (1995). This model takes into account users' qualifications in commonly understood computer skills. It follows the dimensions proposed by Pressman (1992). This model assumes that users have their initial opinions about the software product when they are first introduced to it. Users then continuously discover features of the software product, gaining new information and reviewing their beliefs. The authors conclude that users finally come to an objective opinion about the software product's quality. In 1997, the authors presented a revised model, which can be used in conjunction with any software product quality model (McCall's, Boehm's, FCM, ISO/IEC 9126:1991 etc.). A summary of this model is presented in the most recent publication by these authors. The authors emphasize that user perception changes over time but arrives at a level of consensus, meaning that all users' final opinions are very similar and therefore relate to the real software quality (Stavrinoudis, et al., 2005). The results of their research are presented in Figure 3-3.

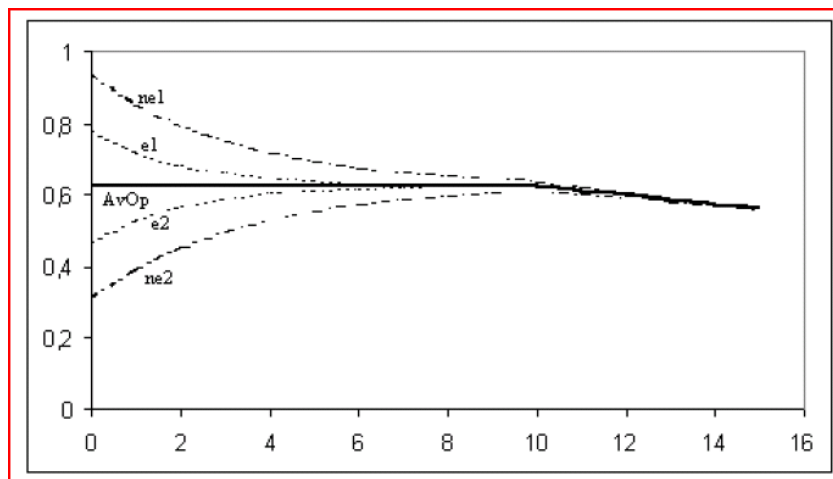


Figure 3-3 Belief revision regarding software quality (Stavrinoudis, et al., 2005)

The above approach is based on the assumption that users are rational agents using deductive reasoning and that beliefs may be represented in a formal system. The authors do

not analyze the context of the user (the context of purpose) or the user's personal aspects (tiredness, attitude, treating the evaluation seriously etc). The authors continue to measure technical quality factors, as defined in ISO/IEC 9126:1991, although usage of these is commonly regarded as being too abstract to express the user's perspective (Suryan, et al., 2003). The most important problem with their results is the problem of repetitive observations within the same group of users. In this case, it is likely that the experiment influenced users' opinions, in which case their tendency for changing their beliefs to a similar level could have been the effect of a group thinking phenomenon or could have been influenced by large amounts of external information not related to the software product being evaluated. The phenomenon observed by the authors may also be explained as a regression to mean effect (Shaughnessy, et al., 2005) (compare Basili, 2007).

Bobkowska provides a further distinction between perspectives dependent on the background of the evaluator (2001). IT personnel consider the technical quality of the application, while information management specialists consider the credibility of the author of the information, their adjustment to reader needs, their objectivism, as well as the actuality and coverage ratio of the subject. Each of these characteristics is divided into sub-characteristics. Ergonomics specialists perceive quality as a consequence of effectiveness, efficiency and satisfaction (compare ISO/IEC9241-11, 1998). Media specialists emphasize aesthetics, the usage of multimedia tools, the adjustment of presentation style to information category, and reader preferences. Dymek (2000) distinguishes between the technical and marketing quality of the software product. An analogical distinction is made by Kobylński (2005), who considers marketing quality as a challenge for Commercial Off The Shelf (COTS) software products, however suggesting that a proper marketing approach could also improve quality assessment of dedicated software products.

Comparing perceptions of a software product's quality to perceptions of food quality (following the discussion from section 2.1), it may be seen that the subjective character of quality perception has been noted. One such attempt, in terms of cognitive processes, was made by Steenkamp in his 1986 dissertation (1986, 1989), which was revised by Oprel in 1989. This model is presented in Figure 3-4. Steenkamp's model inspired several succeeding models of quality perception of food, plants etc. as well as in research regarding the area of the influence of social background on food quality perception (Sijtsema, 2003).

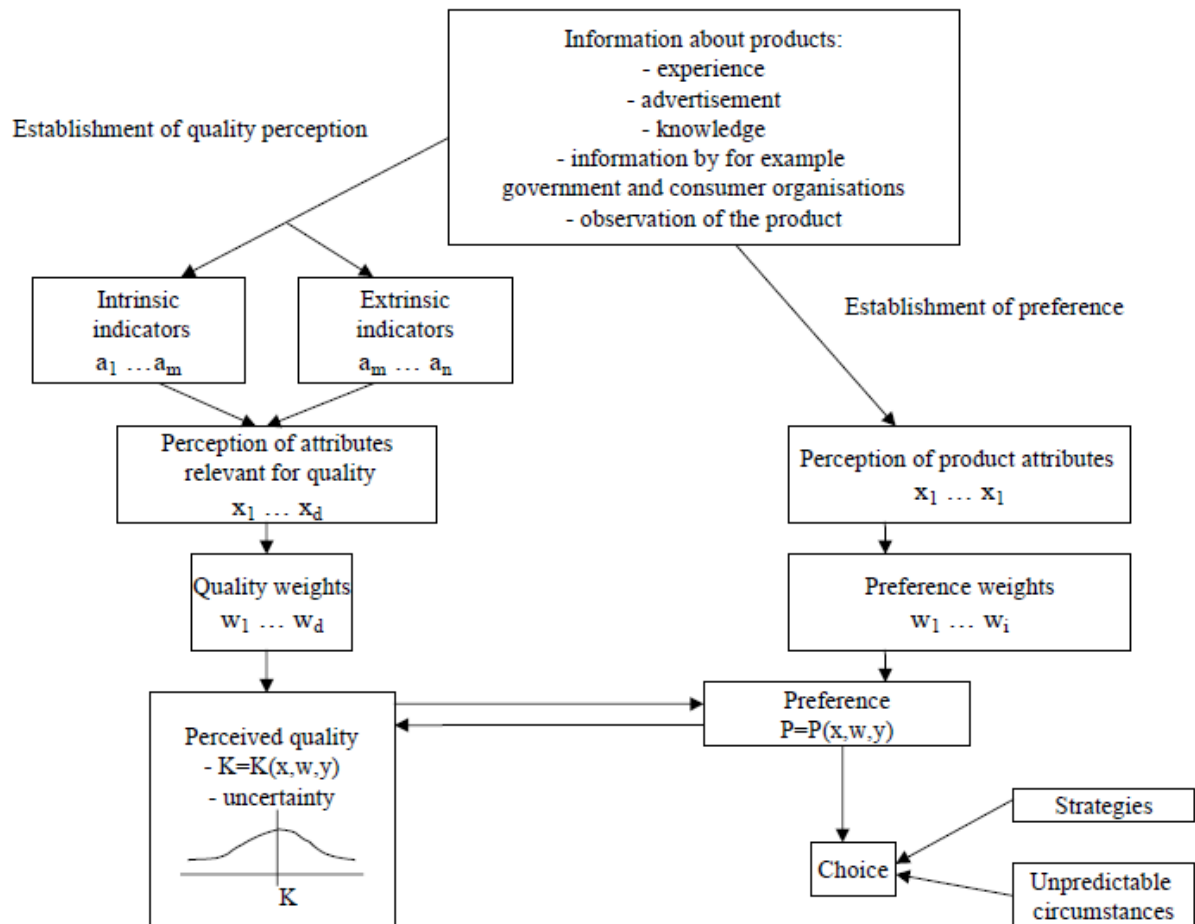


Figure 3-4 Food quality perception model (Steenkamp, 1989)

In Steenkamp's model the product attributes are divided into those that are extrinsic (visible to the observer), and those that are intrinsic (not visible). A similar distinction may be found in the works of Kramer et al. (1983) and Braddon-Mitchel et al. (1996). The proposed model does not analyze the role of the person making a judgment (compare roles in software quality assessment processes Kitchenham, et al., 1996), or acknowledge that a change of opinion may occur over time (consumption in the case of food products is typically a one-time act). This model is intended to be applied in the case of an individual judgment process, and does not include group or organization decision making processes. Therefore, although the model provides interesting insights about the actual quality perception process, it seems to be irrelevant for the purpose of software quality perception.

In summary, this section underlines basic facts regarding software quality models and decisions based on quality assessment. Several software quality models, including the latest, reveal the lack of a commonly accepted software quality model and even commonly accepted software quality related vocabulary (Kobyliński, 2005). However, the most of the models share the same approach: they tend to suggest how the software's quality should be assessed.

In these terms, the models are regarded as normative models. The discussed attempts to analyze the subjective perception of software quality were performed without regard to behavioral economics research methods. As a result, the research conclusions reflect a well known research bias: regression to mean. According to the author's best knowledge, no intensive research in this area has been performed to date. Analogous research was performed in regard to other types of products. However, the results of such research may not be simply transposed to software products, due to the complexity of the latter.

## 4 MODERN APPROACHES TO DECISION THEORY

The contemporary approach to decision theory considers actual patterns of human behavior, and attempts to understand and explain them (Angner, et al., 2007). A new direction, which emerged in psychology in the 1970's, was called "behavioral decision making" (BDM) or "behavioral decision research" (BDR). Two important factors resulting from the cognitive revolution were the ability to express human judgment and decision making processes with the use of computational models, and the observation that cognitive processes play a major role in judgment and decision making processes (Hastie & Dawes 2001). One of the most significant contributions to BDM was the prospect theory developed by Kahneman and Tversky (1979). They attributed the impact of cognitive processes on decision making to the limited capabilities of computational power and temporal memory structures.

Behavioral economics emerged in opposition to behaviorism and similar doctrines in psychology and neoclassical economics, including positivism and verificationism (Angner, et al., 2007). Some authors suggest that the name is imprecise and should be rather "cognitive economics" (Lambert (2006) quoting Eric Wanner, President of the Russell Sage Foundation).

Customer decision making models began to rely on theories related to cognitive processes (Bettman, 1993). Researchers have noticed that when decision makers face making complex and risky decisions (these attributes are often associated with software product related decisions) in a short period, their choice to apply the classical approach was assumed to be irrelevant (Ozanne, 1988).

BDM research was considered in the 20<sup>th</sup> century as being different from mainstream economics (Fischhoff, 1988). However, currently the descriptive approach is considered as a natural supplement to the normative approach (List, 2004). Currently, this research area is developing mainly in directions related to external influence on decision making processes, time discounting, and the role of experience and neuroscience, which allows better understanding of the processes within the human brain (see Camerer, et al., 2003 for a review). The ongoing discussion considers the dualism in judgment formulation and decision making. This dualism was one of the main topics of Daniel Kahneman's prize lecture (2003). Kahneman explains that the idea of a distinction between automatic and reasoning systems guided his research when prospect theory was being formulated. Although the idea is currently generally accepted (see Evans 2008 for review) there are empirical results which provide counter examples to the dual-processes model (for example De Martino et al.'s 2006).

Results regarding the reflection effect are interpreted supporting multiple systems by Kahneman and Frederick (2007), while Tom et al. (2007) interpret loss aversion as a counter example to dual-systems processing.

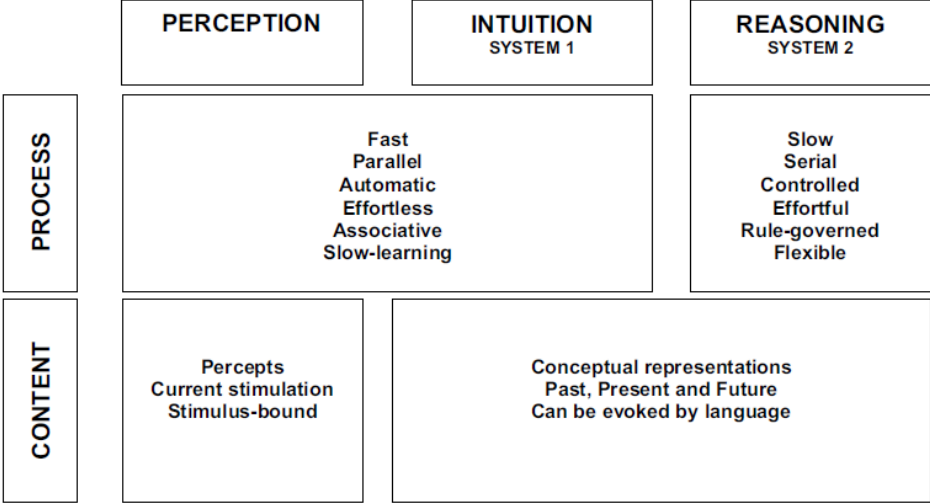


Figure 4-1 Dual-system model (Kahneman, 2003)

Recently, the duality of processes has been analyzed in the context of deliberation and self-control (Brocas, et al., 2008, Fudenberg, et al., 2006, Benhabib, et al., 2005, Loewenstein, et al., 2004, Bernheim, et al., 2004). The “first system” within the cognitive model is fast, effortless and driven by temporal state (e.g. emotions), while the “second system” is slow, controlled and deliberate (Kahneman, 2003). The inclusion of emotions is the consequence of a revival of interest in their influence on the economic decision making process (Elster, 1998). Two types of emotions and moods are distinguished: immediate and anticipated (Han, et al., 2007). The Appraisal Tendency Framework (ATF) proposed by Lerner and Keltner (2000), defines integral emotions as those associated with the anticipated state (e.g. fear) (Loewenstein, et al., 2003), while incidental emotions are defined as those that are surprising to the decision maker (Lerner, et al., 2004). Empirical evidence shows strong influence from both kinds of emotions. However, most decision makers deny the fact that they are influenced by emotions (Han, et al., 2007).

Unconscious processes play an important role in the ability of decision makers to automatically analyze cues and connect them into patterns (Polič, 2009). Decision makers are able to transform problems to some predefined canonical form, although Simon and Hayes suggest that they will perform the transformation which is the most straightforward (1976).



Although different representations of the same problem may be equivalent from a mathematical perspective, they are not from a cognitive point of view (Hau, et al., 2008).

When the recognition process is performed unconsciously, it is so fast and effortless that decision makers are not even aware that such a process is taking place. However, if this process is misled, then they are processing inadequate associations (Polič, 2009) (this may be regarded as one explanation for the framing effect). Kahneman and Frederick define this process as an attribute substitution process (2002), concluding that due to substitution a decision maker is able to give an answer to a question they were not asked (Kahneman, 2003). However, according to Zukier and Pepitone, decision makers are more likely to base their decisions on non-biased information when they are directly instructed to think as statisticians (1984). The opposite effect was observed by Shiv et al., who observed that when people were involved in a task “consuming” their cognitive resources, their self-control mechanisms were weaker (1999).

Immediate emotions are difficult to anticipate, and their influence is difficult to exclude (Han, et al., 2007). The anticipation of emotions reveals a systematic misjudgment about their strength and influence. For example, people overestimate the pain of loss or underestimate embarrassment (Van Boven, et al., 2005). Another misjudgment is related to the emotional gap between the “cold” and “hot” states of the brain (Bernheim, et al., 2004). The decision maker is able to make deliberative, long term decisions in a cold mode and react shortsightedly in a hot mode. However, when anticipating decision making processes in different states of the brain, decision makers misjudge the influence of the state change.

The impact of emotion on the decision making process adds a new dimension to the comparison of options. For example, sadness has been reported (Lerner, et al., 2004) as a feeling which may reverse the endowment effect (Thaler, 1980), happiness has been reported as a modifier of behavior while playing the ultimatum game (Andrade, et al., forthcoming), while anger was reported to modify the assessment of a negative situation into one that was more positive (Hemenover, et al., 2004).

The decision maker may also experience a feeling of potential regret if a particular decision should have been taken that was not. This perspective is represented in regret theory (Mellers, et al., 1997). The regret theory proposed by Meller et al. incorporates the impact of non-expected effect, which strengthens emotions (Loewenstein, et al., 2003). Meller et al.’s theory therefore also incorporates disappointment theories (Bell, 1982), (Gul, 1991), (Loomes, et al., 1986) which concentrate on the feeling of the decision maker when the

outcome was different than expected (for example, bronze medalists on average are more positively affected than silver medalists Medvec, et al., 1995).

The anticipation of emotions for the purpose of decision making is associated with the knowledge and experience of the decision maker (List, 2004). In recent years, the experience gathering process was analyzed in terms of future decision making. One of the most significant observations, named the description-experience gap, was first reported by Barron and Erev (2003) and replicated by several authors since (Hau, et al., 2010). This gap is said to show the reversal of behavior predicted by prospect theory, when people gain their knowledge by experience (in repeated conditions) or via description (Rakow, et al., 2010). Organizational interactions are perceived as repeated games (Camerer, et al., 2007), and this may have a significant impact on decisions made in organizations. This has led some researchers to call for separate theories to explain choices in these two situations (Hertwig, et al., 2004) (Weber, et al., 2004). However, during experiments in the 1970's, Kahneman and Tversky showed that under one set of circumstances decisions were made based entirely on new evidence, while under a second set of circumstances they were based on opinions previously formed (1972).

Anticipation of the outcome is also considered to be a challenge for prospect theory, as it may influence the reference point from which gains and losses are considered (Kőszegi, et al., 2006). Loewenstein and Lerner suggest that decision makers consider their anticipations about future states and also react to surprises (2003). In predicting the evaluation of a process, empirical evidence suggests that the most influential factors are the emotions at the peak and at the end of the process (Redelmeier, et al., 2003).

Other modern decision theories consider various aspects of decision making and the decision maker's background. For example, case-based decision theory postulates a model where the decision maker compares the average of outcomes of the same action in previous cases based on their experience (Gilboa, et al., 1995), or compares their result to the results of others (see Haisley, et al., 2008 for a review of literature related to social comparison effects).

One of the most recently developed research areas within economics is neuro-economics (Camerer, et al., 2005). The employment of neuroscience tools and methods in economics research is expected to provide neurobiological and computational insights in value-based decision making processes (Rangel, et al., 2008). It follows behavioral economics in terms of attempting to understand the processes of judgment formulation and decision making (Loewenstein, et al., 2007). This approach has already begun to show the influence of the brain's reward system on decision making, the role of affective factors in economic decisions and competitive games etc. (Sanfey, 2007). Neuro-economics thus seems to be able to go

beyond the barriers mentioned by Jevons, who doubted if economists would ever be able to measure feelings (Camerer, et al., 2005).

One of the most significant contributions of neuro-economics is related to understanding the endowment effect. The effect of the higher valuation of goods in actual possession of a person, first reported by Thaler (1980), is one of best documented biases (Camerer, 2001). According to neoclassical economics, possession should not affect preferences (Coase, 1960). However, the effect has been observed in both the laboratory and the field (Knutson, et al., 2008). The first attempts to explain this effect were based on the observation that people value losses higher than gains (Kahneman, et al., 1990). Recent research using neural monitoring tools has shown that right insular activation is correlated with endowment (Knutson, et al., 2008). This area has been reported to be correlated with loss experience (Paulus, et al., 2006). The anticipated loss may be typically overestimated, which may explain the endowment effect (Kermer, et al., 2006). It has also been observed in studies related to the anticipation of happiness level after significant health deterioration (Loewenstein, et al., 2008).

Neuro-economics research has also contributed to understanding of the just world perspective. The ultimatum game, first studied by Güth et al. (1982), was studied in terms of the activation of brain areas. The evidence shows that activation in the anterior insula predicts the rejection of an offer (Sanfey, et al., 2003). This area is regarded as a predictor of feeling pain (Knutson, et al., 2008), and its activation is significantly greater when unfair offers are received from human players than from computer random decisions (this supports Blount's findings 1995).

Knoch et al. (2006) have also studied the influence of distractions on the endowment effect. They have shown that when a distraction Transcranial Magnetic Stimulation (TMS) signal was applied to the right dorsolateral prefrontal cortex, subjects were willing to accept unfair offers.

The dual-system model is also the central topic of debate regarding intertemporal choice (Loewenstein, et al., 2007). An important contribution to this debate was made by McClure et al. (2004), who studied brain activation in terms of the immediacy of result. The results show that limbic and paralimbic structures, which are rich in dopamine innervations, were activated when the reward was immediate; however, for delayed reward the activation was stronger in the fronto-parietal regions, which are associated with higher cognitive functions. However, recent results by Glimcher et al. (2007) show that these structures are activated despite the reward delay length.

The dual-system approach may be used to explain experience based or descriptive based decision making. The description-experience (D-E) gap mentioned above (in this section) is the subject of several researchers' current research (Rakow, et al., 2010). One of the main problems with these results is the problem of payoff amount, which is typically low. It is doubtful that such rewards really employ emotions (compare Parco, et al., 2002). However, experience has an incontrovertible influence on judgment formulation (Necka, et al., 2008) and inter-personal behavior (compare Lind, et al., 1998).

The above findings suggest that preferences and choices based on preferences are not a set of pre-defined indifference curves, as presented in the classical approach (Camerer, et al., 2003).

Behavioral economics and neuro-economics have the potential to accurately anticipate consumer behavior. This has been noted by influential bodies in the USA government and the EU Commission (Oehler, et al., 2008), therefore it is expected that these areas will continue to develop.

## 5 EMPIRICAL DECISION RESEARCH

In this chapter, the research methods regarding empirical economics research are discussed, focusing on the contemporary methods relevant to this dissertation. In the second part of this chapter, selected behavioral research results, which may question the normative approach to software quality definition, are also presented.

### 5.1 Research methods

Modern research regarding the actual decisions taken by economic agents employs a variety of methods (Camerer, et al., 2003). Methods from neoclassical economics, behavioral economics, experimental economics, psychology (Stangor, 2007), neuroscience (Loewenstein, et al., 2007) and cognitive sciences (Nęcka, et al., 2008) are used to trace cause-effect relations and to explain the behavior of decision makers. An important method for this type of research is experimentation (Camerer, et al., 2003). The scientific power of experiments was realized during the Renaissance (Levitt, et al., 2008). The first laboratory for conducting experiments was founded in the 19<sup>th</sup> century by Wilhelm Wundt (Boring, 1950).

Regarding the conduction of experiments, behavioral economics uses similar techniques to experimental economics and cognitive psychology (Camerer, et al., 2003). The difference between these areas lies in the purpose and application of the experiment's results.

Experimentation is concerned with tracing cause-effect relations. The three pillars of experimental research are: description of the phenomenon, the foreseeing of behavior, and the explanation of time dependencies (Shaughnessy, et al., 2005).

Experiments may be divided into laboratory experiments, field experiments and quasi-experiments (Shaughnessy, et al., 2005). Laboratories offer the researcher a higher level of scientific control than is possible in the field, while field experiments are characterized by a higher level of external validity. Quasi-experiments analyze effects caused by a natural occurrence (Stangor, 2007). In recent years, called the third distinct period of experiment history, experimenters have transferred strict control methods from the laboratory to the field and employed methods using brain scanning etc. (Camerer, et al., 2003). Randomized field experiments are regarded nowadays as a typical experimentation method (Harrison, et al., 2004).

Modern economics field experiments are used to verify new economic theories. Typically, they use natural groups for experiment sampling, and the experiment is performed with

subjects who are not fully aware of experiment's goal (Levitt, et al., 2008). The unawareness of the subjects is used to avoid randomization bias and the Hawthorne effect. This type of experiment is named natural field experiment. An important issue related to this type of experiment is a set of ethical requirements codified in the *Code of Ethics* published by the American Psychological Association (APA, 2003). In common practice, subjects are asked to commit to usage of their participation record after the experiment with the possibility of decline (Nęcka, et al., 2008).

Experiments are usually conducted using a pre-defined experiment plan (e.g. independent groups plan). The experiment plan should address validity threads and mitigation methods (Shaughnessy, et al., 2005). The internal validity should focus on issues which may result in confounding (including the sampling method, the homogeneity of groups, the analysis of temporal precedence, co-variation and the non spuriousness of an observed effect etc., Levine, et al., 1994). External validity focuses on the probability that the observed effect will be replicated. It may be decreased if the experiment is conducted in pre-set and constant conditions (such experiments are more sensitive), and increased when researchers use balancing methods to control variations of variables (Shaughnessy, et al., 2005). However, for experiments regarding effects resulting from psychological theories, the analysis of statistical significance is less important than the analysis of effect strength, which is used for external validity analysis (Mook, 1983).

Measures are intended to be objective to avoid the experimenter effect. Typical scales used for the purpose of experiments are nominative, ordinal, interval and ratio. The ratio scale is based on scales proposed by Stevens (1951) (e.g. Duckworth et al. (2002) uses a Likert-type scale consisting of 11 levels). Depending on the scale, appropriate statistics are permissible. Statistical analysis is based mainly on Fisher's (1935) and Gossett's (Student, 1923) work. The methods and importance of sample randomization were described by Jerzy Neyman in 1934 (Fienberg, et al., 1996).

Early papers on behavioral economics established a four step pattern for research (Camerer, et al., 2003). In the first step, the normative model is identified, for which in the second step the clear violations of this model are shown. In the third step, the model is extended to a general one taking empirical data into account. A new model is constructed in the fourth step if the new model is a result of the research (Camerer, et al., 2003).

In the neuro-economics approach, the research is supported by a set of physiology monitoring techniques, especially in the area of brain activation (Camerer, et al., 2005). These techniques use different tools for monitoring the brain: electroencephalography (EEG),

functional magnetic resonance imaging (fMRI), magnetoencephalograms (MEG), positron emission tomography (PET) etc. (Camerer, et al., 2005). The application of brain and physiological reaction models are used mainly to analyze and understand the roots of decisions (Zweig, 2007). Therefore, in many cases researchers replicate well known behavioral economics experiments in order to identify activation areas in the brain (Camerer, et al., 2005).

## 5.2 Customer rationality boundaries

Neoclassical economics assume the perfect rationality of the decision maker (Camerer, et al., 2003). The history of decision theory is presented in chapter 3. In this section, examples of documented violations of rationality assumptions are presented. These results are analyzed in section 6.4 in context of the possibility to anticipate the solution of the research problem basing on these results.

The concept of bounded rationality was introduced into decision theories by Herbert Simon (1956). He argued that, being uncertain of the outcomes and of information acquiring costs, agents are unable to follow normative decision processes. Instead, agents decide to select alternatives which are satisfying enough. Simon has also pointed out that in economics rationality is understood in terms of the choices it produces, while in other social sciences it is viewed in terms of the processes it employs (1976). In the 1970's, Daniel Kahneman and Amos Tversky introduced prospect theory (1979), which described a set of heuristics and systematic biases related to the evaluation of options.

In subsequent years, researchers developed several descriptive models of behavior. The systematic irrational distractors of decision making processes are commonly known as cognitive biases (Tversky, et al., 1974). An exemplary list of biases and other violations of rationality assumptions relevant to the product evaluation process is presented in Table 5-1.

Common name	Description
Loss aversion	A bias described in the works of 18 <sup>th</sup> century researchers (Bernoulli, Smith), as investigated by Kahneman and Tversky (1979). The bias uses the observation that people typically prefer to avoid losses and to acquire gains.
Endowment effect	The endowment effect is a cognitive bias observation that people tend to place a higher value on the goods they actually possess than goods

Common name	Description
	they do not possess (Zeiler, et al., 2004). This bias may be explained in conjunction with the loss aversion bias (Kahneman, et al., 1990).
Anchoring bias, Anchoring heuristic	The anchoring bias and heuristic is a tendency to use only a piece of information which is suggested to the decision maker (Tversky, et al., 1974). The suggestion may be related to the context of the judgment being made or bear no relation to it (Hoeffler, et al., 2006). When processing acquired information, the subject does not distinguish observed information from the information which results from cognitive processes (memories, associations etc.) (Barnett, et al., 2005).
Time discounting inconsistency	The time discounted utility in the neoclassical approach is an exponential function. Assuming a hyperbolic function to represent a time pattern, supported by several empirical research results (see Ainslie, 1975), it seems to produce inconsistent valuation over time. People tend to foresee their future willingness to invest as being much stronger than their current level of willingness to invest (Camerer, et al., 2003).
Group effect	There are several observed effects which take place when a decision is to be made by a group (Baron, 2005). Groups tend to formulate polarized opinions. In one of the first experiments evaluating group effect, it was also shown that a group influences the declared opinion (Asch, 1951).
Recent Information bias, representativeness heuristic	The recent information bias and representativeness heuristic are effects observed when subjects receive a continuous stream of information. The most recent information is assumed to be more important and dominates the information gathering phase in the decision making process (Nęcka, et al., 2008).
Confirmation bias	The tendency to look for or interpret information to support one's beliefs is named the confirmation bias (Sternberg, 2007). The effect may be related to the anchoring bias, as when the subject is anchored to an opinion then they look for confirmation of the anchor.
Media bias	Media bias is the common name for the tendency to pay more



Common name	Description
	attention to information which is more frequently noticed (e.g. in media). This phenomenon provides the basis for several observed effects: exaggerating the meaning of the minority view, sensationalism etc. (Nelson, 2003).
Impact bias	The impact bias is the effect of overestimating the consequence of certain facts. Similar biases with a common root cause are the durability bias and the overestimation of the probability of exceptional occurrences (ascription of causality to exceptional conditions) (Gilbert, et al., 1998).
Repetitive bias	This bias is an effect of the phenomenon recognized since ancient times known as rhetorical argumentation by repetition ( <i>argumentum ad nauseam</i> ). Information assumed to be more important because of exposure to it from multiple sources or situations has also been investigated in exposure effect research (the earliest known research reports are from 1876 by Gustav Fechner) (Bornstein, 1989).
Faulty generalizations bias	In many cases, people ignore the rules of probability (Baron, 2000). One of the biases resulting from this phenomenon is the faulty generalizations bias. This tendency is the estimation of certain attributes of the general population based on a small sample of one's own experience.
Contrast effect bias	The contrast effect is the tendency to alter one's opinion about an object when it is compared to another object, or when exposure of another object influences the perception process. The described change may be a positive or negative change of the assessment grade (Nęcka, et al., 2008).
Choice-supportive bias	When a decision or judgment is made and new information appears, the tendency of sticking to a judgment already made is known as the choice-supportive bias. An explanation of this phenomenon could lie in the confirmation bias (Mather, et al., 2000).
Conjunction fallacy bias	The conjunction fallacy bias is the tendency to treat the probability of a specific occurrence as being higher than a general one. One of the first publications regarding this effect was the research on

Common name	Description
	associations with a certain description of a woman (Linda) described during the experiment performed by Tversky and Kahneman (1982).
Hawthorne effect	The Hawthorne effect is the observation that people behave differently when they are aware of being observed. The change is driven by a feeling of how they should behave, based on knowledge or cultural background (Henslin, 2008).
Just world bias	One of the assumptions regarding <i>homo economicus</i> is the maximization of utility. This assumption seems to be violated when people solve decision making problems based on feeling just. In the Ultimatum game (Güth, et al., 1982) subjects decide to decrease their potential payoff from the game to punish an unjust person.
Status-quo bias	Following the status quo or default option is reported to be one of the most robust and powerful forces in consumer decision theory (Han, et al., 2007). In many cases vital decisions are made following the way the question was asked by selecting the default option (Samuelson, et al., 1988).
D-E Gap	The description experience gap has been studied in recent years, and thus represents a comparatively new direction in behavioral research (Rakow, et al., 2010). The empirical results show that experience based decisions are based on the underestimation of rare events, which is contrary to prospect theory.
Emotional Gap	The emotional gap is associated with the inaccurate anticipation of feelings by underestimating their adoption abilities. This gap was studied in the context of happiness (Loewenstein, et al., 2008) and economic decisions (i.e. endowment effect Mellers, et al., 2009).
Hot-Cold gap	The Hot-Cold emotional gap is related to the misjudgment of cognitive processes when the subject is in a certain state of mind (Bernheim, et al., 2004). The decision maker makes deliberative, long term decisions when in the “cold” mode and reacts shortsightedly when in the “hot” mode. However, when anticipating decision making processes in different states of the brain, decision makers misjudge the influence of the state change.

Common name	Description
Peak time	The impact of circadian rhythms on decision making processes was studied in 2004 (Gonzalez, et al., 2004) with the use of the ultimatum game (Güth, et al., 1982). The conclusion of the study showed that people were less co-operative when they made decisions at off peak times.

Table 5-1 Selected cognitive biases (source: own study)

The above presented heuristics and biases were described from the perspective of potential influence on software product quality perception. The list is not exhaustive. However, assuming the possibility of influence from these biases, the result of evaluation may be significantly changed as a result of them.

Their potential influence begins with the observation that the analysis of a large number of decision options requires an adequate amount of time and resources. Thus, in a real decision making situation the decision maker uses heuristics and associations which are prone to cognitive biases, and relies on intuition, beliefs etc. (Kahneman, 2003). The first description of cognitive heuristics was given by Herbert Simon, and the concept was further investigated by Kahneman and Tversky (Nęcka, et al., 2008).

Tversky and Kahneman described the anchoring heuristic, availability heuristic and representativeness heuristic (see Table 5-1) with several systematic biases (1974). In recent years, researchers have identified and described subsequent heuristics and biases, such as the escalation of commitment (Barry, 1976), sunk cost effect (Arkes, et al., 1985), entrapment in investment (Rubin, et al., 1975) (these effects are based on a cognitive bias limiting the decision area to the consequences of prior investments), and naive diversification (this heuristic is based on the fallacy when choices made in decisions regarding multiple selections are made more various, Simonson, 1990). The concept of heuristics was revised by Kahneman and Frederick (2002). Their approach introduced the concept of attribute substitution, extended the concept of heuristic beyond judgment, and described the conditions under which intuitive judgments will be overridden by deliberate reasoning (Kahneman, 2003).

Judgments and opinions are understood as cognitive structures arising as a result of learning processes which also influence current and future processes. From a constructivist standpoint, these processes influence the formation of the whole mind (Nęcka, et al., 2008). Cognitive structures are said to represent, among other things: concepts, symbols, their

relations etc. The representation of concepts entails the encoding process, which converts sensory information into mental representation.

The encoding process is explored mainly in the context of memory related processes (Nęcka, et al., 2008). In an experiment concerning double encoding, subjects were given sequentially the same information in different modal representations (Paivio, et al., 1973). The authors have shown that the repetition of a message influenced the formation of permanent cognitive structures, which resulted in a higher level of information retrieval. Bransford (1972) has shown that the meaning of information is memorized better than the form in which it was presented.

In the literature, there are different concepts regarding the rise of mind representations, distinguishing the representations of abstract concepts (e.g. the set of natural numbers) from representations of natural concepts (e.g. a table), and from the representation origination (see Nęcka, et al., 2008 for a review). Wetherick (1968) has proven the speed and ease with which mind representations appear. Representations themselves depend on education and cultural issues (Rosch, et al., 1976), as well as the individual characteristics of the person (Murphy, et al., 1985). Reed (1972) argues that concept representation is formed when the object is seen for the first time (example theory). This is in opposition to the generalization process, which is based on the similarity of observations (Nęcka, et al., 2008).

Concept representations are stored in the cognitive system in a layout of interconnections between them. The Adaptive Thought Control (ACT) model proposed by Anderson (1976) is the one of most commonly accepted ones. The ACT model describes relations between concepts as links, with their strength dependent on the probability of joint activation. In this model, when one concept is activated it causes the activation of related concepts, starting from the strongest relations. Most representation theories use a similar concept in the area of relations storing (Nęcka, et al., 2008).

The activation concept has several consequences for cognitive processes. The first consequence is the propagation of the activation mechanism. Having a larger number of interconnected structures means that the concepts activate these structures more slowly. If one interconnection is much stronger than the others, then this activation dominates the relation and hampers the activation of weaker interconnections (Anderson, 1976), (Reder, et al., 1983). The second consequence is the sustaining of the activation for a period of time. Sustained activation interferes with the succeeding processes, and hence affects the result of the process (Wylie, et al., 2000).

In summary, the main reasons for rationality boundaries are associated with the possession of incomplete information by decision makers, cognitive limitations associated with the way the brain operates, learning and experience limitations, a limited amount of time to make decision, the emotions and frames in which the problem is described etc.

However, the behavioral approach has, in several cases, been criticized (List, 2004). Several scientists regard irrational behavior as a result of a lack of experience in the area where the decision is to be made (Brookshire, et al., 1987). For example, analysis of the endowment effect has shown circumstances where no effect was observed (Shogren, et al., 1994), or where the effect changed the perceived value by twelve times (Carmon, et al., 2000).

Modern research on bounded rationality and neuro-economics insights are described in chapter 4.

## II. CONCEPTUAL MODEL

In this part, the conceptual model is described. The software industry's perspective, described in section 2.5, contains positivists' assumptions about the evaluation processes that are performed by the customers in regard to software products. However, the behavioral economics perspective provides ample examples to show that actual processes differ significantly from utility-maximization models. The results apply to simple decision making processes, and therefore the secondary research results do not solve the problem regarding the accuracy of these two approaches.

In this part, selected empirical evidence from the software market, related to quality assessment, is presented with the analysis of the applicability of the research methods for the purpose of verification. In chapter 7, the Software Quality Perception Model is introduced as the hypothesis resulting from empirical observation and literature analysis. Finally, the requirements regarding verification of the model are formulated.

The Software Quality Perception Model is the hypothetical part of this dissertation in regard to the hypothetico-deductive research model described in section 1.3.

## 6 IDENTIFICATION OF THE VARIABLES IMPACTING SOFTWARE QUALITY PERCEPTION

In this chapter, the empirical observations regarding the software market are presented. The observations contradict the normative software quality models, therefore the analysis of the methods that dispel these contradictions are identified.

### 6.1 Empirical data from the market

The models of products' quality assessments on the software market typically represent a prescriptive approach (see section 2.4). Since the first models of software product quality assessment, authors have tried to propose definitions for the attributes of software relevant to quality (see Hofman, 2007 for a review).

The most widely used models of software products' quality take the producer's perspective as the dominant perspective (compare Kitchenham, et al., 1996). The user's perspective is included in the ISO/IEC 9126:2001 and SQuaRE models (ISO/IEC25000, 2005). However, even in these models, the main objective of the standard is to define objective and measurable characteristics of software quality (compare A.4 *Metrics used for*

*comparison* section in ISO/IEC 9126-1:2001 and the definition of validation in ISO/IEC 25000).

The objectivism of the measure seems to be a natural assumption regarding the model of a product's quality. This assumption implicitly adopts other assumptions:

1. The evaluator possesses all relevant information,
2. The evaluator is making rational judgments, and
3. There are no external influences on the evaluation process.

The main models of software product quality do not provide an exact algorithm for the combination of the attribute measures into a single, comparable value. Such algorithms are implemented for the purpose of the automatic, quality-based selection of web services (see Abramowicz, et al., 2009 for a review). However, the proposed approaches are based on the linear weighting function, which features a discriminating values set (Jaeger, et al., 2006), (Vu, et al., 2005). Such an approach is typically used for multiple criteria decision making processes (Hwang, et al., 1981). In some approaches, the authors propose a function representing, for example, usefulness (Zeng, et al., 2003), (Liu, et al., 2004).

The algorithms, provided explicitly or implicitly for the process of composition of the attribute values into a comparable measure, emphasize the assumption regarding the rationality of the evaluator, the requirement to possess all relevant information, and the rejection of external influence. The comparison process itself brings implicitly to the forefront another assumption regarding the maximization of a user's profit or utility. This assumption is not required for the situation of evaluating software products, as evaluators do not have any goals related to the evaluation results.

The assumptions, which are implicitly taken as the basis for the software product quality assessment process, strictly reflect the normative assumptions regarding human behavior in neoclassical economics. Therefore, the models based on these assumptions are regarded as normative models.

Observations on the software market pose a contradiction with the approach described above. The first observation described below in this section is based on a real, large project conducted in Poland 2006-2010<sup>7</sup>. The project was conducted by one of the largest software producers in Poland, who were employed by one of the central government offices. The merit of the project was highly dubious. However, it had large budget, allowing for the engagement

---

<sup>7</sup> The author was one of stakeholders in the project. However, the details are not publicly available

of appropriate resources. Therefore, the vendor hired additional specialists, who had domain knowledge and required experience in the area.

The project team (project manager, the customer and other participants) were optimistic about the project. The scope was clear, resources were secured, and the know-how was in the possession of the vendor. After the first phase, the project seemed to be progressing on schedule. Therefore, the project manager agreed with the customer that the project could be optimized by allowing the testing of the application to be performed by the customer.

The first problem appeared when the first version of the product was delivered to the customer for testing, after 20 months of development and 4 months before the project's planned termination. Although customer representatives remained enthusiastic, they were not able to test any of the business processes, because the system crashed on almost any action. The vendor was asked to deliver a correct version of the product in two weeks. However, there were no resources able to verify its correctness on the vendor's side due to a previous decision regarding the placement of testing at the customer's side.

The successive version was delivered. However, the quality of this version remained low. Customer representatives discovered several serious errors during first day of tests.

This was the turning point of the project. The customer warned the project manager that they expected a high quality product or the agreement was to be terminated. The customer withdrew from the agreement regarding performing the system tests.

The vendor was forced to finish the project irrespective of cost. The testing team was put together. However, it soon became clear that the observed problems were rooted in the system's design. The correction was planned for the next year (exceeding the original schedule by 38% of time). The schedule was extended several times after that event. Finally, the project lasted for 250% of the time of the primary schedule.

An unusual situation was observed in the final 6 months of this project. There were no serious problems with the system (there were about 5 unsolved, non-critical problems left). However, the customer still regarded the system as being unacceptable. According to the normative software quality model, the latest version should have been assessed with regard to the defined quality requirements. If it had, then the quality should have been assessed as satisfying. However, the customer's opinion was different. This observation suggests that the customers representatives were assessing not only the latest version itself, but were expressing the feelings and opinions they had developed throughout the project's duration. Based on this observation, the normative approach was shown to be inaccurate (Hofman, 2011). This is



similar to the way in which neoclassical economic models were shown to be inaccurate (for example, by Kahneman and Tversky, 1979).

A similar instance of “perception anchoring” was observed after serious quality problems were discovered in the Microsoft Windows Vista operating system. The system was improved and became more stable. However, public opinion about this system remained unchanged. Another occurrence contradicting the normative models was observed among customers who did not use Microsoft Windows Vista, yet shared public opinion about its low quality. Their opinion could not have been based on their experience, or on their observation of the system. Therefore, it could only have been based on the opinion of others. In this case, the opinion of other people is beyond the scope of normative models. Therefore, if the influence of memories on quality assessment process is shown, the inaccuracy of the normative models is also shown.

The above examples were presented only in terms of real world observations which contradict normative models. However, at this stage the nature of this contradiction could not be assessed, if the variance from the normative models was sound and important for the understanding of the actual processes taking place in the market.

Further observations were based on the potential area, where the variances resulting from subjective perception could have affected the assessment processes. The analysis of the aforementioned importance of subjectivity of the process was based on a statistical analysis of software releases in a probe of 15 projects with a total budget exceeding US\$250 million. This analysis showed that for a single planned release of a version, the customer received five versions – four of which were rejected due to errors. This remarkable figure shows that internal quality control has prevented the release of only one erroneous version. The details of this analysis are presented in Table 6-1.

Characteristic	Number	Comment
Number of planned releases	4,202	The number of versions to be delivered to the customer according to the production plans
Number of versions released only for internal evaluation purpose	4,752	These versions were intended to be evaluated in respect to the reporting of bugs
Number of versions declared by the development team to be	23,035	Includes versions declared to be ready for release to the customer which were passed to the internal quality control or directly to the

Characteristic	Number	Comment
ready for release		customer
Number of versions completely rejected by the internal quality controllers	2,584	The number of versions for which the result of the internal quality control pointed out critical issues despite the subsequent decision to send the versions to customers
Number of versions delivered to the customer with a known list of not corrected bugs	9,194	For those versions the known list of bugs could result in the version's rejection by the customer
Number of versions rejected by the customer	16,310	Rejection is defined as a demand for change in the software before its release to the live environment

Table 6-1 Summary of characteristics for 15 projects with a budget of over US\$ 250 million (source: own study)

The figures presented in Table 6-1 show that in the analyzed sample the problem of low quality delivery is significant. Therefore, if customers assess the quality of succeeding versions on the basis of previous ones (as in the examples shown in the beginning of this section), then the project should expect to encounter serious problems. In contrast, Stavrinoudis et al. argue that users' opinions evolve to some objective value (2005). Summarizing a decade (1994-2005) of research conducted in Greece at the turn of the 21<sup>th</sup> century, the authors present evidence to show that users finally reached a consensus view on product quality despite different levels of assessed quality initially (discussed in section 3.2).

Revealing the descriptive inadequacy of the most widely accepted software quality assessment models, based on the empirical evidence described above, seems to be natural. For example, one group of users may argue that one operating system is much better than another, while another group may oppose this view (see Casadesus-Masanell, et al., 2006 for an example). If normative models were adequate, then such a situation would not arise.

However natural, formal proof of this is difficult to acquire. Modern approaches to software quality models underline that product quality is to be assessed in the specific context of its use (compare ISO/IEC25000, 2005). As the software product's context of use is not unambiguously defined, the supporters of current quality models could attribute differences of opinion to differences in the context of use (the software is typically regarded as complex and

difficult to compare Hochstein, et al., 2008). Therefore, an experimental approach is required. This type of research is regarded as the key future direction for the software engineering discipline (Basili, 2007). This direction requires a reliable research method, and the ability to set up and control the environment for research purposes (Hochstein, et al., 2008).

The cognitive revolution in the second half of the 20<sup>th</sup> century has provided concepts and tools for emerging branches of economics research related to understanding decision makers' behavior (Angner, et al., 2007). Several descriptive models were proposed (see section 4 for a review), and as a result researchers were able to construct descriptive models which could be used for prediction purposes.

One example of such a model is prospect theory (Kahneman, et al., 1979). Kahneman and Tversky conducted several experiments, the results of which directly support positive-negative asymmetry (see Figure 3-1). Classical economics representatives had speculated about the existence of such phenomenon (compare Smith, 1759). These findings established a new perspective on understanding the process of software quality assessment: that it is possible that quality attributes are assessed in relation to some reference point.

This research problem, stated on the basis of the above considerations and empirical data, reflects the construction of the Software Quality Perception Model. The research method requires also the verification of the proposed model. Therefore, the problem may be expressed via the following questions:

1. Is it possible to construct a descriptive model of users' quality perception processes in regard to software quality?
2. Is it possible to prepare a relevant method for the purpose of the empirical verification of this model?
3. Is it possible to prepare a relevant method for the purpose of setting up and manipulating the research environment?

This dissertation assumes a positive answer for all of the above stated questions.

## **6.2 Methods related to software industry**

The models for software product quality assessment are intended to present a normative and descriptive view of evaluation processes. For example, the most recent SQuaRE model (ISO/IEC25000, 2005) contains definitions of basic measures, definitions of derived measures, and the relation of these measures to software quality characteristics, which are finally related to the overall assessment value (ISO/IEC25010 FDIS, 2011). This model summarizes the mainstream models developed by software engineering researchers, and

reflects the framework mandatory for software quality models as defined by the IEEE 1061 standard (1998). Therefore, it is regarded as the software engineering approach to software quality assessment for the purposes of this section (compare Suryn, et al., 2003).

The solution to the research problem described in the previous section, which is based on the software engineering approach, would assume that the normative model is the descriptive model (similar to the assumptions regarding the EU and SEU models described in section 3.1). The normative approach is based mainly on a set of preferences that are elicited (but nevertheless stated or implied), and that are stable during the evaluation process. This approach is commonly accepted in the literature (see section 2.4), and may be regarded as the present state-of-the-art in the area of software quality assessment.

The solution based on the software engineering approach rejects all influences from cognitive processes limitations, the personal and temporal cognitive predispositions of evaluators, or the influence of the information environment related to the product (i.e. the sequence of the presentation of information, information not related to the product but potentially associated with the product, the change of preferences during the evaluation process etc.).

The present state-of-the-art in research regarding the actual quality assessment of software products does not allow researchers to either empirically support, or neglect, the hypothesis related to the influence from cognitive process on software quality assessment. On the other hand, current research methods in this area do not allow researchers to refute the hypothesis that actual judgment processes are significantly different from the normative models presented in the literature. In terms of Popper's and Lakatos's postulates regarding the verification of theory (Lakatos, 1970), it may be noted that the normative approach is empirically unverifiable.

A closely related solution may be based on the SERVQUAL approach (Parasuraman, et al., 1985). This approach allows the respondent to formulate their subjective view of quality in the context of decomposition to characteristics, and then allows them to compare the assessed object's quality against an ideal entity. In this sense, SERVQUAL adopts Plato's perception of quality (quality as a degree of perfectness) (Kiliński, 1979). Adopting the SQuaRE model's set of definitions for the purpose of quality assessment (compare Abramowicz, et al., 2008), the approach based on the SERVQUAL method would contain a set of questions related to the relevant software attributes.

SERVQUAL, however, is a method for measuring the current state of users' subjective judgments. It does not allow the discovery of casual relations between the environment and

the assessment results, and therefore cannot be applied to explain the reasons for customers' attitudes and their origins. This research direction could be considered as non scientific (Lakatos, 1974).

A solution based on belief revision theory (Peppas, et al., 1995) could be employed to solve the research problem. However, the theory itself assumes that the evaluator will move toward objective information about the assessed system. Therefore, it should be regarded as a model of the judgment process in the context of the normative model. This approach rejects the emotional, biased and irrational behavior of evaluators. It is based on the AGM paradigm (Alchourron, et al., 1985) and Grove's system of spheres approach (1988) as applied by Xenos et al. (1995).

Xenos et al. (1995) and Stavrinoudis et al. (2005) employed belief revision theory. However, their research cannot be regarded as evidence that their theory is valid. The authors observe the change of opinion during a longitudinal experiment. However, their results support the regression to mean effect rather than belief revision theory (Shaughnessy, et al., 2005). In such research, the crucial aspect of internal validity is related to confounding: the research description suggests that the regression to mean effect resulted from information flow beyond the laboratory (the same seminar group was taking part in the experiment for half a year) and group effects. Sjøberg et al. (2002) identified flaws and a non-realistic approach in such experiments.

In summarizing this section, it is important to note that there is no single software quality model commonly accepted by the industry. The models proposed throughout last 40 years focus on the prescriptive approach, suggesting how the quality of software products should be evaluated, but with no respect to how the process is actually performed by the evaluators. The perception related research presented above reflects the same approach, as the authors conduct longitudinal research with no control mechanisms over group dynamics or external influence on the results. Their results could be obtained due to the regression to mean effect, therefore they do not reveal any useful or meaningful insights about the actual process of software quality assessment. Other attempts to explain the subjective perspective of evaluators were limited to the use of SERVQUAL or similar tools. These approaches may be used to identify an evaluator's subjective view, but generate no insights into the causal relations between the evaluator and the available information about the product. Therefore, after conducting an extensive literature review, to the author's best knowledge there are no examples of research devoted to understanding the actual processes related to software quality perception.

### **6.3 Methods based on the neoclassical economics approach**

The problem may be potentially solved by using neoclassical economics normative models. These models offer a similar set of assumptions regarding rationality, full information, stability of preferences etc. In fact, the software engineering approach is based on neoclassical economics models, as the beginning of the discipline dates back to the 1950's. However, according to revealed preferences theory (Robbins, 1932), also regarded as part of mainstream economics, only observed decisions or actions taken may be studied as the input for the analysis of the evaluator's preferences and attitude. This assumption contains an implicit suggestion that the research should use empirical evidence (i.e. descriptive modeling) instead of a prescriptive approach.

The above approach may be considered in the context of Friedman's positive economics (1953). The approach offers a set of assumptions regarding the rationality of the decision maker, and on this basis the normative models offer predictive ability. However, this approach does not offer any explanation for the empirical data described in the previous section. Notably, this type of approach to scientific inquiry was deemed pseudoscience by Lakatos (1999).

Software quality assessment and decisions related to this assessment in many cases influence the final quality of the software. If the product is rejected, then the customer may expect significant correction of the quality. However, any delay in implementation usually results in some economic loss (financial loss, loss of market share etc.). Conversely, the acceptance of a software product with low quality may result in future failures resulting in significant losses. Causal and Evidential Decision Theories (Joyce, 1999) describe decision models regarding the situation where the decision influences the options, and may be perceived as a game between vendor and customer (compare Aumann, 2006, Camerer, et al., 2007). However, according to Egan's counterexamples (2007) both models fail to solve certain decision problems. Therefore, one cannot expect an universal solution based on either of them. In both models, the normative approach to the decision making process leaves aside rationality boundaries (see section 5.2).

Some authors point out that users' preferences may not be compared in terms of the satisfaction of more than one need by the product (Lutz, et al., 1979). Standard preferences-based theories may not, therefore, be applicable to software products which aim to satisfy a wide range of needs.

## 6.4 Methods based on behavioral economics results

Following the hypothesis regarding the differences between actual and normative based judgment models of software quality, the solution to the research problem could be based on the research results of behavioral and experimental economics.

Models of consumer behavior and judgment processes provided by behavioral economists cover a long list of cognitive processes insights. However, these models are strongly limited by the frame and boundaries of the experimental data gathered, and therefore the speculations regarding the application of these models to software quality assessment processes is not obvious.

An example of such a difficulty may be presented by using the endowment effect (Thaler, 1980). This effect is one of the best documented effects (Knutson, et al., 2008). However, researchers have discovered a reversal of this effect in certain circumstances (for example, when subjects are sad or disgusted Lerner, et al., 2004). The implementation of new software typically results in a set of changes for the business users (e.g. the replacement of the old software system or a change in business processes). However, this information does not clearly suggest that users will value their current system higher, or that they will value it lower due to endowment effect reversal (compare also the contrast effect Hsee, et al., 1999). Another example is related to one of key findings from the work of Kahneman and Tversky: the overweighting of small probabilities (1979). Recent research has shown that when judgment is based on experience, people tend to underweight rare events (Hertwig, et al., 2004). Considering a situation where employees are involved in the evaluation of a product while managers base their evaluation on a report written by the evaluators, the research results do not provide reliable predictions regarding their opinions regarding a software product in a real situation. However, the D-E gap is typically observed in regard to small rewards. According to Parco et al., behavior may be influenced by the magnitude of the stakes involved (2002). Therefore, there is no satisfying answer to the question of the actual judgments of evaluators and managers.

Prospect theory itself introduces another source of potential doubts. The prospects are considered as gains or losses in comparison to a reference point (Kahneman, et al., 1979). However, it is unclear if the reference point is associated with the current state of the evaluator or their expectations (Montague, et al., 1996) (Klein, 2002), and thus it is unclear how the new product will be evaluated.

The above stated example is common to most behavioral economics models. Erev et al. have called this a “1-800” problem (2010), because in their opinion the models are cursory,

and if one wants to use them then they should contact the author of the model for support. As with the cognitive bias reported in prospect theory (Kahneman, et al., 1979), inaccurate reactions to small probabilities are reported to be reversed when the decision maker observes repetitions of the same decision situation (Barron, et al., 2003). In terms of organizations acquiring software, this reversal may be an important issue as organizational behavior is assumed to be equivalent to repetitive decision making (Camerer, et al., 2007).

The fact that most people accept employment as a form of economic activity, and therefore represent the opinion of their employer rather than their own, has broader consequences in terms of the research problem (compare Simon, 1995). Decisions in organizations may be led by political bargain etc. (compare Thompson, 1995) or group dynamics (compare Baron, 2005).

Analogous application problems may be shown for hyperbolic time discounting, representativeness heuristics, anchoring heuristics etc. (see section 5.2). Generally, the application problem may be regarded as a problem with the non-holistic character of behavioral models and the identified problem of preference comparison for complex products (compare Keeney, 1977, Hochstein, et al., 2008). Simon describes another problem with behavioral models relating to the professional decision makers themselves (1987). In his opinion, biased decision making is limited when the decision maker makes a decision on behalf of somebody else (Ariely, et al., 2003). This issue is typical for professional software evaluators, who are employed as independent evaluators. The same issue is reported by Simon in regard to emotions, which seem to be significantly less influential when the decision does not affect the decision maker (1987). However, the just world bias (Sanfey, et al., 2003) may still occur, as the evaluators may react adversely if they assume that the quality level is unjust and harmful for the users (people tend to assess as if they were meant to be harmed Andreoni, et al., 2002, Charness, et al., 2002).

Behavioral economic approaches as well as other economic theories are being extended via the use of neuroscientific tools and methods (Loewenstein, et al., 2007). Some of the results of this contribute to a better understanding of cognitive process (e.g. the link between the endowment effect and the anticipation of future pain Knutson, et al., 2008), or shed light on areas which could not be investigated otherwise. For example, McClure et al. have studied neural reactions to the consumption of preferred soft drink (2004) or to unpleasant smells (de Araujo, et al., 2005). Their methods could not explain the cognitive background of the observed reactions. However, if users have analogous preferences to certain types (e.g. style,



brand etc.) of software products, than it is expected that their reactions are supported by strong chemical reactions within the brain.

Another example is related to the distractions of brain processing with the use of Transcranial Magnetic Stimulation (Knoch, et al., 2006). In this research, an electromagnetic signal applied to the Right Dorsolateral Prefrontal Cortex has significantly changed decision maker behavior. Therefore, it may be possible to unify reactions to the product with the use of neural stimuli. However, it is difficult to imagine that such methods would be accepted by the industry or customers, although such research was conducted in relation to the selection of movies to watch (Read, et al., 1999).

Although neuro-economics is perceived as a promising research direction (Loewenstein, et al., 2007), it currently focuses on research into chosen types of decision processes or chosen parts of the brain. The most significant limitation is related to the methods of brain scanning, which require laboratories, specialist equipment etc. The study itself is also invasive, and may cause change in the behavior. Therefore, these methods cannot be used for solving this dissertation's research problem, which aims to trace actual behavior in real situations.

## **6.5 Conclusion**

The above analysis has considered currently available methods which could be used to explain empirical observations of the software market. Among normative models (software engineering, neoclassical economics) or descriptive models (behavioral economics, experimental economics, neuro-economics etc.) there are no means of analyzing and understanding the actual processes of software quality assessment. The main problems with the application of current methods lie in the large number of needs which are to be satisfied by a software product, and the inadequacy of normative models to explain observable gaps between rationally optimal decisions and actual decisions. Descriptive models do not provide a holistic picture of judgment formulation processes, and provide results that are highly dependent on the research circumstances, which may not be transferred to the area of software evaluation. The complexity of the research problem is also related to the fact that the product is typically evaluated by an organization and not by individual evaluators. Therefore, it would be difficult to transfer the individual model to an organizational one, even if an individual model existed.

Barron and other authors call for the empirical analysis of the decisions being taken (2003), while Kahneman points out that decision processes undergo unavoidable cognitive

limitations, and are therefore inevitably biased (2003). This suggestion, and the lack of a reliable solution based on current state-of-the-art theory, results in the conclusion that empirical research is necessary to solve this dissertation's research problem. Therefore, in the next chapter a hypothetical model is proposed. Further on, this model is empirically verified, according to a hypothetico-deductive research method.

# 7 SOFTWARE QUALITY PERCEPTION MODEL

## 7.1 Construction of Software Quality Perception Model

This section describes the construction process of the descriptive model regarding the judgment process of a product’s quality on the software market. The model is based on the research results in the area of behavioral economics and empirical observations of the software market. Therefore, the model establishes the hypothesis outlined in the selected research method (Popper, 2002). The model’s verification results are described in chapter 9.

The basic normative model for quality perception is based on the simple weighting of a product’s attributes (see Wilkie, et al., 1973 for a review). The weights depend on the evaluator’s preferences for the specific context of its use. The overall quality grade is expressed by a formula  $K = \sum_{i=1}^n a_i w_i$  and the model is presented in Figure 7-1.

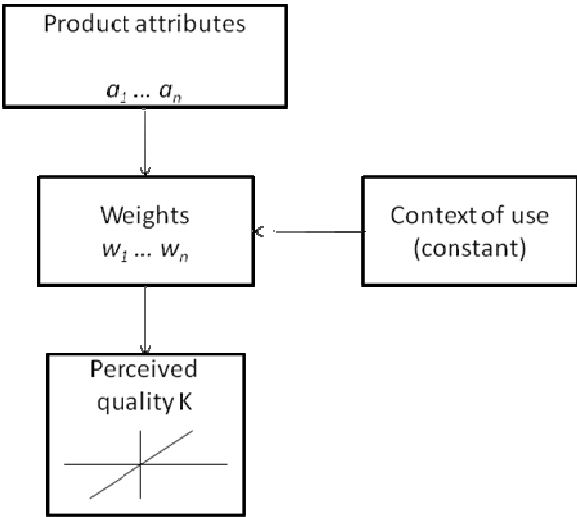


Figure 7-1 Normative-based model of software quality perception (source: own study)

The normative based model was enhanced according to cognitive research results. Ariely et al. have observed that people focus their attention on the attributes they assume are relevant for certain situations (2003), and other researchers have shown that information not related directly to a product influences its perception (see Camerer, et al., 2005 for a review). These observations may be modeled as two sets of attributes (one related to the product, and the other not related to the product; or in other words, intrinsic and extrinsic (compare Braddon-Mitchel, et al., 1996). The attributes are filtered by an attention filter  $F(a_i)$ , which is also dependent upon the context of use. The definition of the filter is expressed by the formula:

$$F(a_i) = \begin{cases} a_i & \text{if } a_i \text{ is observed} \\ 0 & \text{otherwise} \end{cases}$$
 (filtered attributes are denoted  $x_i = F(a_i)$ ). The overall quality grade is expressed by the formula  $K = \sum_{i=1}^m F(a_i)w_i = \sum_{i=1}^m x_iw_i$  and the model is presented in Figure 7-2.

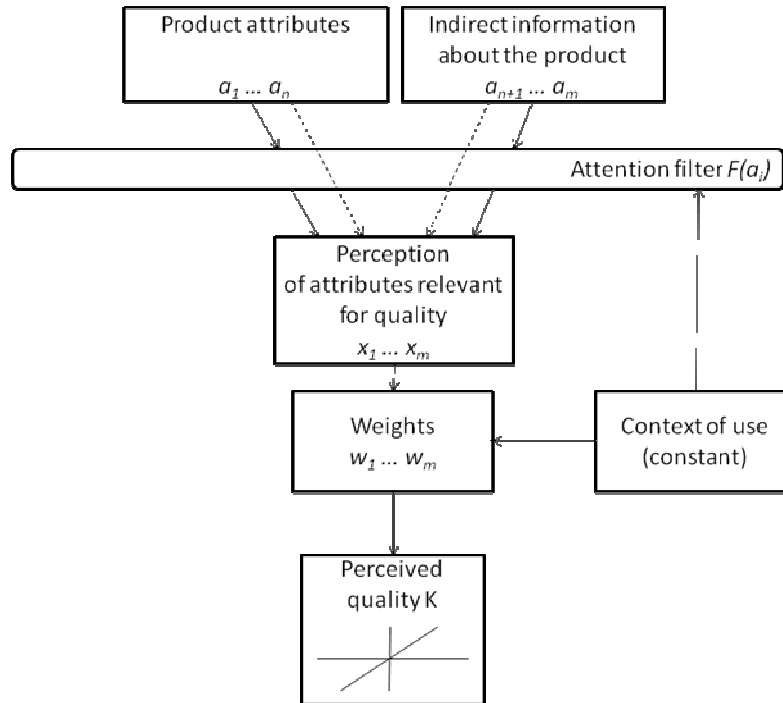


Figure 7-2 Normative-based model of software quality perception with perception filter (source: own study)

In the next step, the analysis of weights were applied to the model. According to mainstream behavioral economics research, the judgment process is dual: it is driven by immediate emotions, or by rational behavior (compare Kahneman, 2003). Additionally, Lewin et al. define aspiration levels, which are not static but rely on changing experience (following Simon, 1979). Gonzales and Loewenstein suggest that temporal state or mood also influences judgment and decision processes (2004). In this context, the source of weights were replaced by representation of knowledge (rational processing) and mental state (emotional processing). Both areas provide feedback on the attention filter, and are supplied with final judgments. The formulas of the model remain unchanged. The enhanced model is presented in Figure 7-3.

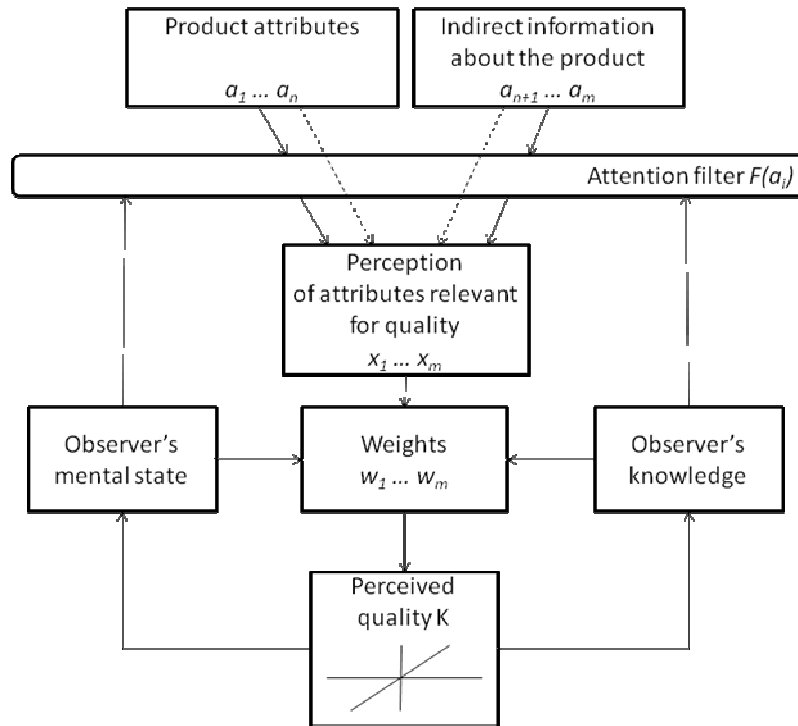


Figure 7-3 Normative-based model of software quality perception based on knowledge and mental state of the observer (source: own study)

In the final step, the model was modified in the area of the final computation function of perceived quality. Many empirical research results, even with the use of neuro-scientific tools and methods, have proven the existence of positive-negative asymmetry (compare Loewenstein, et al., 2007). Therefore, it should be expected that the final assessment of the quality level is closer to the logistic function (denoted  $L$ ) than to linear scaling. Moreover, to express the relation of the mental state to the level of needs saturation and emotions, the weights were divided into two types: knowledge dependant, and needs dependant. Additionally, the influence of mental state and knowledge on the perception of attributes, and a loop-back influence of perceived attributes on an observer's knowledge (a similar influence can be identified from observed attributes to mental state, although it is assumed that the overall grade influences the mental state rather than single observations) were identified. The overall quality is expressed by the formula:  $K = \sum_{i=1}^m L(x_i, w_i, s_i)$ , and the model is presented in Figure 7-4.

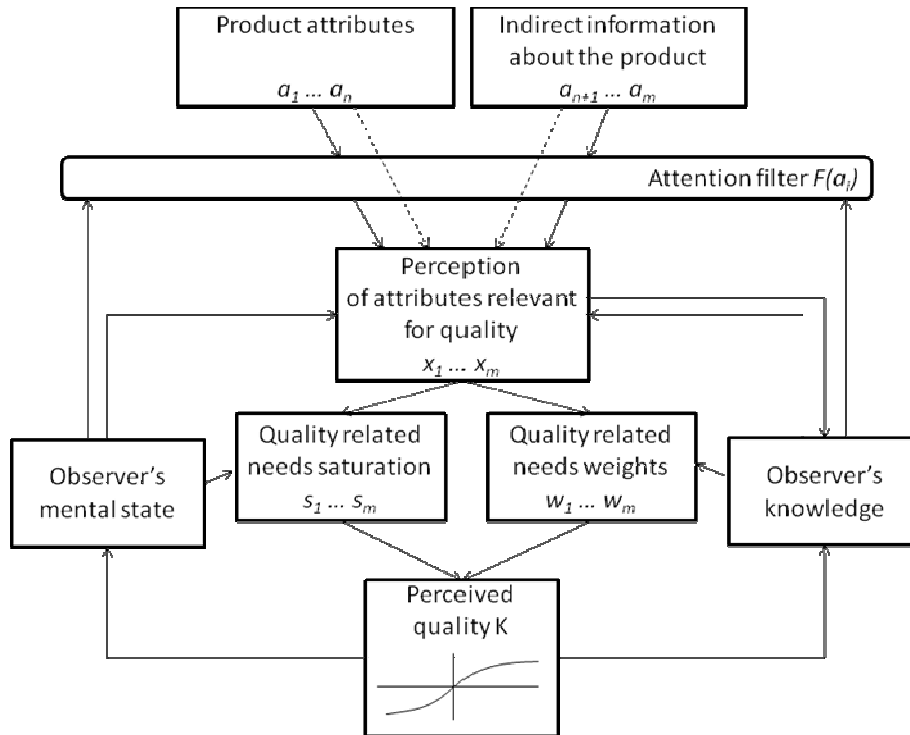


Figure 7-4 Theoretical descriptive model of software quality perception (source: own study)

The above model is based on the literature review, and forms a theoretical model of the software quality perception process. The first version of this model was proposed by the author of this dissertation in 2009. Following discussions with members of the scientific community, it has been improved to the current version, as published in 2011. The model described in this section was verified using empirical research based on behavioral economics methods.

## 7.2 Characteristics of the model

In this section, the deductive consequences of the hypothetical model described in the previous section are presented. The analysis of consequences serves both the assessment of the potential consequences for software market practices and the preparation of the verification of the model. The deductive part is presented in the context of the differences between proposed and commonly accepted normative models.

The first significant difference is related to the inclusion of the influence of attributes not related to the product's perceived quality. This difference reflects directly the empirical observations of the software market. The consequence of the influence of such attributes could be observed if identical observational situations were presented to two or more statistically equivalent individuals or groups. However, with the addition of information not

related to the product (for example, with the use of gossips or some information irrelevant in regard to the product, but relevant for the domain of its application), the influence of information not related to the product on the quality perception process would be manifested by a change of its assessed quality value (following the presentation of additional information to evaluators).

The second difference reflects the existence of the attention filter in the process. The deductive consequences of such a filter being used in the perception process could be observed if distractions are applied during modification, or when the evaluators react to some unusual occurrence, thereby losing their focus on the overall product. The influence would be confirmed if in analogous observational situations the assessed quality level was observed as a consequence of the treatments mentioned above.

Normative models reflect an observer's state as a single source of opinions. The mental state, knowledge, importance of needs, and the level of needs saturation is treated as a single source of weights for attributes. In the proposed model, the influence of the observer's mental state is defined as being distinct from the influence of the observer's knowledge (the distinction between knowledge about needs and their actual level of saturation is discussed below). The consequence of such a distinction could be observed if the assessed quality level was different depending on the observer's knowledge and mental state.

The attention filter is altered by the observer's knowledge and mental state. This fourth difference from the normative model produces a consequence which could be observed if the reaction in focusing or losing attention was dependent on the observer's state (e.g. if in comparable situations the focus of an angry observer differed from that of one who was emotionally calm). The influence of an observer's knowledge could be investigated in an analogous way.

The fifth difference reflects the hypothesized influence of the perceived attributes on the observer's knowledge. This relation underlines that even if the observer has not made their overall judgment on the quality level, their knowledge is being altered. The deductive consequences could be verified if the evaluation process revealed information not given to the observer at the beginning of the evaluation process, and if the information was adopted and used further on in the evaluation process.

It was mentioned above that the presented Software Quality Perception Model distinguishes between two groups of observer behavior. One of these was described above (the distinction between knowledge and mental state), while the difference between stated and actually satisfied needs is denoted as a sixth difference. This distinction results in different

assessments in situations where needs are said to be important but are actually not saturated by an old product, or are said to be less important but are saturated by an old product. Naturally, other combinations of these attribute values are also part of the consequences deduced from this hypothesis. The verification of the influence of actual needs saturation and the stated importance of needs could be achieved if comparable situations were observed and a new product was evaluated by evaluators with different opinions about the importance of various attributes and with different level of needs saturation (in these cases, it is more important to identify the observer's state, because it may be difficult to modify the saturation level of needs during the research).

The next difference between the proposed model and normative ones is the logistic function, which is used to calculate overall quality grade. Typical normative models are based on the weighted sum of attribute values (see section 2.4 for a review). The consequences of using a different function to calculate the overall quality grade would appear as a positive-negative asymmetry reaction to the quality change (compare chapter 4).

The eighth and last difference underlined in this section is the loop-back of the assessed quality level on the observer's mental state and knowledge. It is important to note that the subject of this loop-back is the judgment that was made by the observer. Therefore, this influence may contain biased information. A deductive consequence of this loopback would be a change of the observer's mental state or of their reaction to a similar product in the following tasks.

The above list presents the eight most important differences between the Software Quality Perception Model (SQPM) proposed in the previous section and normative software quality models. These differences are discussed in the context of related research methods in the next chapter, and are summarized in Table 8-1.

The SQPM is a hypothesis based on an extensive review of behavioral economics research results, and empirical data gathered from the software market. The consequences presented in this section would have been observed if the SQPM reflected actual processes related to software quality perception processes (it is important to note that none of these consequences would occur if a normative model could be assumed to express actual processes). According to the hypothetico-deductive research model, the next stage of the research should be devoted to verifying the deduced consequences. However, in the case of software products, research regarding product quality perception requires the construction of a dedicated research method and tools.



## 8 SOFTWARE QUALITY PERCEPTION MODEL VERIFICATION

This chapter discusses the research method used for verifying the hypothesis (i.e. the Software Quality Perception Model). In the first part of the chapter, the research methods are compared, then the requirements regarding the research design are discussed. In the following parts, the variables used in the verification process are identified, and in the last part the requirements regarding the execution process are presented.

### 8.1 Research methods overview

The empirical observations of the software market, which contradict normative software quality models and delegitimize current explanatory methods, has been discussed in chapter 6. The analysis of these methods was concluded with a discussion of the position of notable economists, who stated that decision making and judgment processes are cognitively limited.

The theoretical Software Quality Perception Model presented in chapter 7 summarizes the boundaries of cognitive processes documented in empirical research. However, it seems that the application of available research results is inadequate to verify this model. Because empirical research has limited external validity (Nęcka, et al., 2008), and because software products are assumed to differ from other kinds of marketable goods (Basili, 1993), the research problem should be solved by using empirical research.

The Software Quality Perception Model differs from the normative model in that the assumptions are made based on behavioral economics research results. Verification of the model should assess proposed modifications of the normative model. This will enable researchers to foresee the result of the judgment process more accurately. The assumptions of the theoretical model and identified verification methods are presented in Table 8-1.

No	Description of difference	Identified research method
1	Influence on perceived quality of attributes not related to the product	This may be researched empirically using the description or experience of the observer (e.g. the opinion of other people)
2	Existence of the attention filter in the process	The influence of this filter may be investigated empirically using a simulation of a real situation
3	Distinction between the influence of the observer's knowledge and mental state	This may be investigated empirically using circumstances affecting the observer's mental state and knowledge in different ways
4	Influence of knowledge and	This may be investigated using the same methods as those

No	Description of difference	Identified research method
	mental state on the attention filter	used for point 2 above, with additional manipulation causing hypothesized influence
5	Influence of the perceived attributes on knowledge	This may be investigated in a series of experiments. The model assumes influence from the overall perceived quality level and from the perception of attributes on knowledge. Therefore the influence of single attribute perception on knowledge has to be designed
6	Distinction between needs weights and needs saturation in the perception process	This distinction may be investigated by comparing declared and actual influence on the perceived quality of certain attributes
7	Logistic function used for the calculation of overall quality level	This may be investigated empirically using a series of experiments comparing increments of quality level with the overall quality grade
8	Influence of overall perceived quality level on mental state and knowledge	This may be investigated empirically by causing changes in mental state and knowledge

Table 8-1 Software Quality Perception Model differences to normative model (source: own study)

The research methods described in Table 8-1 are not an exhaustive list. They do not, for example, describe methods based on secondary research results or those based on prescriptive models of behavior, as discussed above.

Empirical research designed to cover the aspects listed in Table 8-1 consist of a series of experiments. As software product related empirical research is typically expensive (Hochstein, et al., 2008), the research design allows several aspects to be investigated in a limited number of experiments. The soundness of the evaluated model is dependent on the validity of the research results. Therefore, the research should be designed with adequate attention to the assurance of internal and external validity (discussed in section 9.6).

Behavioral economics research methods (described in section 5.1) employ laboratory or field experiments as empirical research methods (Camerer, et al., 2003). Laboratory experiments have limited external validity, while field experiments lack scientific control methods for the precise set up of the environment (Nęcka, et al., 2008). Natural field experiments, whereby researchers transfer strict control methods from the laboratory to the field, aim to address these issues. One important aspect of this experiment type is the subjects' unawareness of the research goals (or even of the fact that they are participating in the

experiment) (Levitt, et al., 2008). However, according to the ethical rules published by the American Psychological Association, the subject has to be informed about the goals of the research and has to commit to participation (APA, 2003). These requirements in the context of natural field experiments are typically fulfilled after the research has taken place (Nečka, et al., 2008). If the subject does not commit to the use of their record of participation (even anonymously) then the record is removed afterwards.

Natural field experiments seem to be adequate for the purpose of research regarding the research problem explored in this dissertation. However, there are no examples of such experiments regarding software products in the literature (see Hofman, 2011 for a review). Natural field experiments offer a means of scientific control. However, there are no known methods for the manipulation of software product quality without changing the nature of the product (e.g. if part of a software product is changed then the new version may be regarded as a new product, in which case the comparison cannot be limited to quality difference only; compare Basili, 2007).

To conclude the research method selection analysis, it is clear that the research design needs to cover the aspects shown in Table 8-1, that it should follow the natural field experiment concept, and that there is a need to design tools and methods to ensure an appropriate level of scientific control for the research (compare with the major obstacles for good software related research pointed out by Hochstein et al., 2008).

Additionally, the research method should utilize the quality measures selected for the dissertation (see section 1.3.3). Natural field experiments seem to fit into these requirements:

- 1) The research method may be used for both the verification and falsification of the models,
- 2) The experiments may be documented and ensure full disclosure,
- 3) The detailed design of the experiments may be detailed, and thus ensure repeatability and robustness, and
- 4) The experimental control mechanisms allow control of the influence of variables impacting on the results and observational errors.

Based on this comparison, the approach selected was one based on natural field experiments.

## **8.2 Requirements regarding the research design**

The considerations presented in the previous section suggest that natural field experiments (Levitt, et al., 2008) are the most appropriate research method in regard to this dissertation's

research problem. This type of research is based on setting up a research environment that emulates the real world.

In empirical research, researchers typically focus on people, processes or products. However, in regard to software products, researchers need to manage all three dimensions (Carver, et al., 2004): to analyze people in the process of using a product.

Software evaluation processes performed in organizations are the main focus of this dissertation. Software evaluation techniques and methods are described in sections 2.2.4 and 2.5. Users in organizations evaluating the quality of software products may be divided into two groups: the evaluators, and the non-evaluators. Despite the fact that only the evaluators have the opportunity to observe the product, it seems that non-evaluators also perceive the quality of the application (e.g. the manager making a decision regarding the acceptance of the product is typically not a member of the evaluating group). The research design has to reflect this distinction between organization members.

The software evaluation process in organizations may be typically influenced by group effects, emotions, the endowment effect, or even the mood of the evaluators (compare section 5.2). There are no research results regarding the typical (most commonly occurring) interactions within the evaluation team. However, defining the experiment as an ordinal task for evaluators (who would not be even aware that this is an experiment) should closely relate to the actual interactions in the evaluation team.

An important part of each evaluation process is the preparation of documentation and the acquisition of domain knowledge by evaluators. The focus of this research is on the evaluation process. Therefore, all parts of the evaluation task related to the actual evaluation process (including documentation) should be prepared by the experimenter. The product domain should be chosen in a manner that would minimize the need for the domain-specific training of evaluators. This would avoid threats to validity resulting from the training process.

The research topics listed in Table 8-1 were covered by the following set of experiments:

1. Sequential change of quality – in these experiments, the evaluators were asked to evaluate sequential versions of the same product but with different quality levels
2. Associations from the design – in these experiments, evaluators with previous experience of the Graphical User Interface (GUI) were asked to evaluate a product with a similar or different layout
3. Social pressure – in these experiments, the evaluators were put under social pressure during the evaluation

The mapping of these experiments onto research topics is presented in Table 8-2 below.

No	Topic	Experiment
1	Influence of attributes not related to the product on perceived quality	Second series
2	Existence of the attention filter in the process	All experiments
3	Distinction between the influence of knowledge and the influence of the mental state of the observer	Third series
4	Influence of knowledge and mental state on the attention filter	First series
5	Influence of the perceived attributes on observer's knowledge	All experiments
6	Distinction between needs weights and needs saturation in the perception process	All experiments
7	Logistic function used for the overall quality level calculation	First series
8	Influence of overall perceived quality level on observer's mental state and knowledge	First and second series

Table 8-2 Mapping of experiments onto research topics (source: own study)

The presented set of experiments was not the only one possible. However, it was sufficient to provide empirical verification data. Therefore, it was accepted as the basic assumption set for this research.

### 8.3 Variables identification

In this section, the variables hypothesized to influence the perception of software quality are identified. Most of these variables were not manipulated during the empirical research. However, their identification was important for scientific control and internal validity (see section 9.6.1).

The identification of the variables was based on the behavioral economics research results (see part I, especially section 5.2) and processes related to software quality assessment (see section 2.5). The variables which were expected to have reasonable influence and which were relevant to the software evaluation process (with a focus on the evaluation of professional products) were identified and classified into three groups: variables relevant to the environment, the product and the process conditions.

The environment related variables represent the influence that results from the typical process related to software product evaluation. In this group, the variables represent the configuration in which the quality assessment process is conducted (by an individual or by a structured team). The quantity of the group and the background of the evaluators in the context of the software product's domain also play an important role in the evaluation process, as these characteristics influence the course of the evaluation. Another variable

related to the environment is the setting of the attitude and approach of evaluators towards the product. In several cases, the evaluation of the new product is not treated seriously; it is regarded as an extra task which distracts the evaluator from their ordinary daily activities. However, in a professional approach the attention of the evaluators may be focussed only on the evaluation task. Therefore, setting this variable may influence perception. This variable is closely related to a more technical aspect of the environment: the area of distractions which may occur during the evaluation (e.g. when the evaluator is performing the evaluation in an open, noisy space).

The second group of variables consists of the variables related to the product itself. Naturally, the variable impacting on perceived quality level is the extrinsic quality of the product. Measurement of this attribute is unambiguous (see section 2.4). However, quality projected on a certain model should allow the researcher to observe the impact resulting from the quality change. Other variables related to the product which seem to influence the perception of its quality include the product's domain, functionality and ergonomics. For example, in some cases the product may be assessed as having low quality because it is lacking one function which is useful and required by users. Therefore, the functionality and ease with which the product is operated may influence user perceptions of its quality. Another variable influencing the product's quality is its interface design. There are several rules, guidelines etc. related to the design of the Graphical User Interface (GUI) (compare ISO/IEC9241-11, 1998). However, the design's impact may extend to measures such as ergonomics, learnability, clearness etc.

The third group of variables is related to the evaluation process (i.e. to the experiment in the planned research). In this group, the incidental influence resulting from the external information or opinion of non-evaluators is included. Conditions related to the evaluators' state of mind, moods or even synchronization with their peak-times (compare Gonzalez, et al., 2004 and Han, et al., 2007), as well as the task scope definition (clearness, completeness of documentation etc.), is also included. The last variable mentioned here is the possible impact from data gathering tools and techniques, which may influence the results or even affect evaluators' opinions (compare Ubel, et al., 2005).

The variables listed above are summarized in Table 8-3.

ID	Variable	Example (extreme values)
Environment		
E1	Evaluators team configuration	An individual, a team of peers, a structured team etc. This variable contains also the configuration

ID	Variable	Example (extreme values)
		in which opinion may be based on secondary perception (the opinion of others).
E2	Location of team members	Collocated in one room, in one building, in different cities etc.
E3	Quantity of the team	For example, 5 persons.
E4	Evaluators' experience in evaluation	Professional evaluators or people who have never performed professional evaluation.
E5	Evaluators' domain expertise level	Business experts in the domain, or people with no experience.
E6	Attitude of evaluators towards performing evaluation tasks	The evaluators may treat evaluation tasks as an interesting part of their job or as a boring routine.
E7	Approach of evaluators to evaluation tasks	The approach may be based on procedures and formalisms or be "ad-hoc".
E8	Parallel tasks of evaluators	The evaluators may have time devoted to the evaluation procedures or have more urgent tasks to perform at the same time.
E9	Workplace conditions	The workspace condition may support quiet and focused work or be noisy.
Product related		
P1	Extrinsic quality of the product	The product may contain no errors or not have a single function which would be adequate to the requirements.
P2	Functionality	The product may support all required functions properly, or have significant gaps in functionality.
P3	Ergonomics	The product may be user friendly or the opposite.
P4	Learnability	The product may be easy to learn (be consistent with good patterns) or be completely not-understandable.
P5	Clearness	The product may be unambiguous or may make the user feel uncertain of their actions
P6	Graphical design (GUI)	The GUI may generate positive associations
Evaluation process related		
S1	External information	During the evaluation the evaluators may receive positive or negative information about the product
S2	Evaluators' state of minds	Evaluators may be prejudiced toward the product

ID	Variable	Example (extreme values)
		in positive or negative direction
S3	Evaluators' moods	Evaluators may feel happy or angry during the evaluation process
S4	Evaluators' synchronization with peak time	Evaluators may be forced to work during their peak-off time or be left to work as they like
S5	Clearness of task scope	The task may be clear in terms of the evaluation scope, plan, test scenarios etc., or the evaluators may be asked to conduct explorative tests.
S6	Completeness of documentation	The documentation of the product may be complete and unambiguous or may be missing or incomplete
S7	Data gathering method	Evaluators may be asked to present their opinion before their manager, fill out reports etc.

Table 8-3 Variables hypothesized to impact on the software quality evaluation process (source: own study)

#### 8.4 Requirements regarding verification process execution

The planned research aims to verify the hypothetical model, and in particular the differences between the normative software quality model and the Software Quality Perception Model proposed in section 7.1. The assumptions described in the previous section are addressed by three types of experiments, which cover the research topics and may be used to verify the model.

The empirical identification of influence requires a causal approach (the tracing of cause-effect relations). The observation of causal relation requires adequate control methods of the environment and a mechanism to steer the independent variable(s). The comparison of different effects resulting from dependent variable(s') values may be conducted with the use of a replicated study (Hochstein, et al., 2008) and of an independent groups plan (Shaughnessy, et al., 2005).

According to the arguments presented in the previous section, the experiments are to be conducted as natural field experiments (Levitt, et al., 2008). Therefore, the control methods mentioned above have to be strict and precise. The control was planned in regard to the variables listed in Table 8-3.



### 8.4.1 Type 1 experiments

According to the assumptions presented in section 8.2, this type of experiment is used to trace the influence of sequential versions of the same product evaluation on attention, knowledge, mental state, and the overall level quality ascribed to the product. These goals were achieved by manipulating two independent variables: the extrinsic quality level of sequential versions, and the attitude (motivation) of evaluators. Remaining identified variables (see Table 8-3) were set to values typical for the evaluation process performed by professional evaluators (see section 9.2).

Although establishing motivation is purely organizational, the manipulation of variables is rather natural to execute. However, the purposive manipulation of software quality may cause difficulties. As was mentioned in section 2.4, there is no precise model for objective software quality assessment. Therefore, it is difficult to compare the quality levels of two applications. The comparison of different applications by independent groups would have generated a problem regarding the identification of the exact value of the quality difference between applications (the difference could rely on the subjective preferences of evaluators – e.g. a comparison between the quality levels of the Microsoft and Macintosh operating systems).

On the other hand, the evaluation of the same application by independent groups poses another problem: the deliberate manipulation of the application's quality level (e.g. if an additional feature is added, then it may be said that the application has changed and that it is a different one). In the literature, however, there are no examples of such manipulation.

Considering these restrictions, quality level manipulation may be limited to the manipulation of quality levels order. Denoting the quality level of a version  $v$  as  $Q_v$ , and the order relation " $<$ ", the statement  $Q_{v1} < Q_{v2}$  should be interpreted as: the quality level of  $v2$  is higher than the quality level of  $v1$ . The quality levels order is transitive: if  $Q_{v1} < Q_{v2}$  and  $Q_{v2} < Q_{v3}$  then  $Q_{v1} < Q_{v3}$ .

The manipulation of quality levels, which aims to construct a set of versions with a known order, is possible with the use of the fault probability function  $f_p$ . For versions of the same application, fault probability is (*ceteris paribus*<sup>8</sup>) negatively correlated with the quality level. The quality levels order may thus be indicated by having a set of versions ordered by fault probability.

As stated at the beginning of this subsection, in this type of experiment two independent variables were manipulated: the quality level of sequential versions ("history effect"), and the

---

<sup>8</sup> Latin – with other things being the same

motivation level of evaluators (“motivation effect”). In each case, secondary perceptions (by non-evaluators) were also analyzed. Each group thus had a “manager” receiving the evaluators’ reports. The general overview of this research plan is presented in Figure 8-1.

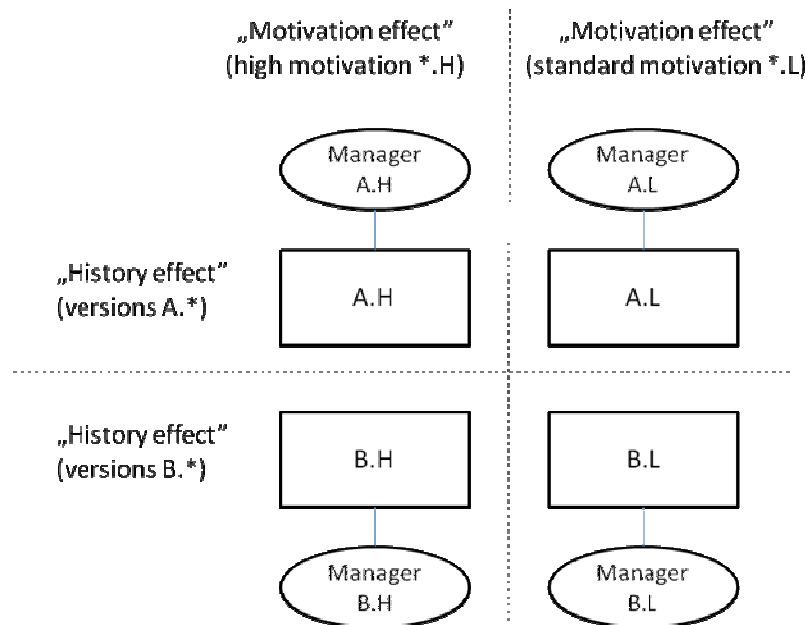


Figure 8-1 Four independent groups layout for experiment 1 (source: own study)

The plan of measuring two different conditions in this experiment addressed the potential difference among subjects who base their opinion on experience (evaluators) or on description (“managers”) (Hertwig, et al., 2004).

The experiment began with a personal survey. The aim of this survey was to gather information about each subject’s domain knowledge (regarding the scope of the evaluated application), and their preferences for software quality characteristics. The results of the survey were used to analyze the groups’ homogeneity (see section 9.6.1). The personal survey was followed by a pre-test, which also aimed to verify the homogeneity of quality assessment levels among groups. Then, for the succeeding five days the subjects evaluated sequential versions of the application. At the end of each evaluation task they filled out a survey regarding the application’s quality level. These phases of the experiment are shown in Figure 8-2.

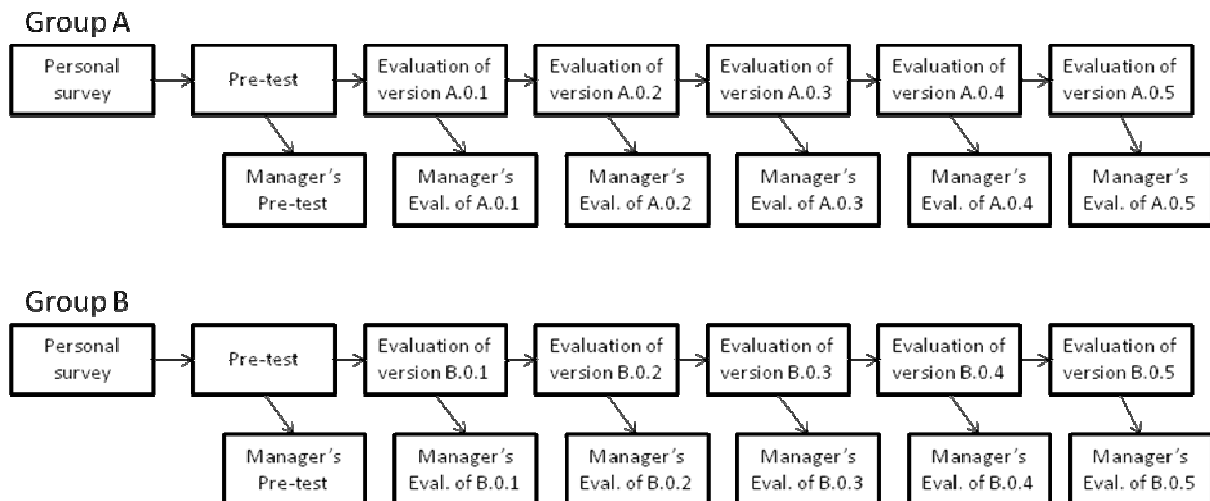


Figure 8-2 Phases of experiment 1 (source: own study)

The two treatments being investigated were: the different patterns of the quality levels of the sequential versions of the applications, and the existence of additional motivations for the evaluators. The fault probability related to the quality level of the versions of the application for patterns A and B is presented in Table 8-4.

version (v)	$f_p(v)$	version (v)	$f_p(v)$
A.0.1	0.00	B.0.1	0.00
A.0.2	0.00	B.0.2	0.00
A.0.3	0.10	B.0.3	0.80
A.0.4	0.15	B.0.4	0.50
A.0.5	0.10	B.0.5	0.10

Table 8-4 Fault probability ( $f_p$ ) patterns A and B (source: own study)

It should be noted that the final versions of both patterns (for all groups) have the same quality level. The goal of the experiment was to address the potential difference in the assessed quality level by both subjects and test managers. The levels of  $f_p$  for versions \*.0.1 and \*.0.2 serve two purposes: to verify homogeneity, and to avoid the potential setting up of a negative picture in subjects' minds, as the starting point could influence their opinions of succeeding versions (compare Hoeffler, et al., 2006).

The second treatment used additional motivations for groups A.H and B.H (groups A.L and B.L. had no such additional motivation treatment). The \*.H groups were told that the proper evaluation of the software was of key importance for their employer because of a strategic decision associated with the evaluation results.

### 8.4.2 Type 2 experiments

According to the assumptions presented in section 8.2, the purpose of this type of experiment was to trace the influence of associations related to product quality perception. The experiment had to manipulate two independent variables: the existence of associations, and the GUI layout of the application. Remaining identified variables (see Table 8-3) were set to values typical for the evaluation process performed by professional evaluators (see section 9.2).

Subjects who had associations with certain GUI (denoted A) evaluated two types of software product in layout A and C. Additionally, subjects without associations with either A or C evaluated these two layouts. In each case, secondary perception was analyzed by non-evaluators. The general overview of this research plan is presented in Figure 8-3.

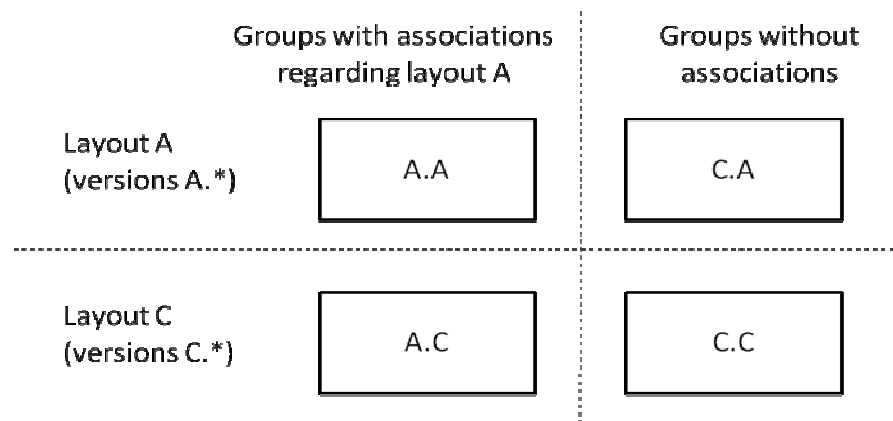


Figure 8-3 Four independent groups layout for second experiment (source: own study)

The experiment was preceded by a personal survey, which aimed to gather information about each subject's domain knowledge (regarding the scope of the evaluated application) and their preferences for software quality characteristics. The results of the survey were used to analyze the groups' homogeneity (see section 9.6.1). Then, groups selected for building up associations evaluated a product (application 1) in layout A. Afterwards, all groups evaluated the new product (application 2) in two layouts: A and C. At the end of each evaluation task, subjects filled out a survey regarding the application's quality level. These phases of the experiment are shown in Figure 8-4.

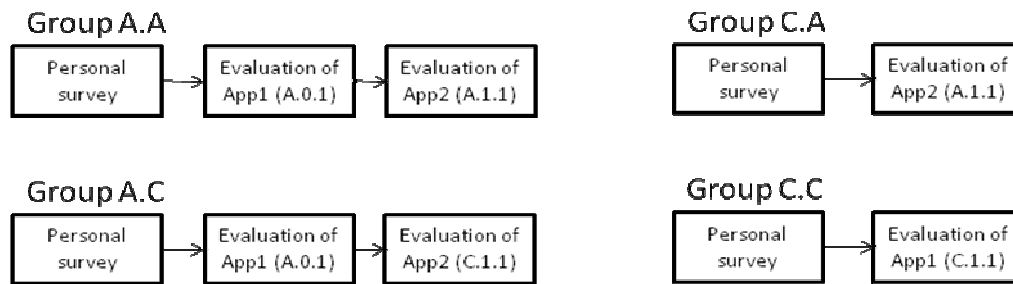


Figure 8-4 Phases of the second experiment (source: own study)

### 8.4.3 Type 3 experiments

According to the assumptions presented in section 8.2, the purpose of this type of experiment was to trace the influence of social context (group pressure) on quality perception. The experiment manipulated only one independent variable associated with the character of external influence. Remaining identified variables (see Table 8-3) were set to values typical for the evaluation process performed by professional evaluators (see section 9.2).

The experiment had to emulate a real situation where subjects evaluate a product using their personal computer and share their opinions with other group members. It was expected that during conversation the subjects would fall into a conformist position (compare Asch, 1951), while their own opinion may have remained unchanged. Therefore, at the end of the experiment subjects were asked to fill out a survey without interaction with other team members.

Pressure on the subjects was applied via the employment of figureheads who displayed positive or negative attitudes towards the evaluated application. In this experiment, the organization of the evaluation is crucial for validity, because the subjects had to be motivated to share their opinions. It is common in real life to organize short report meetings, therefore such a tool was used to encourage subjects to share their opinions (at these meetings the figureheads were asked to share their opinions before subjects).

The experiment was preceded by a personal survey, which aimed to gather information about each subject's domain knowledge (regarding the scope of the evaluated application) and their preferences for software quality characteristics. The survey results were used to analyze the groups' homogeneity (see section 9.6.1). Then, all groups evaluated the product, having 2 to 3 report meetings during the evaluation. At the end of evaluation task subjects filled out a confidential evaluation regarding the application's quality level.

## **8.5 Verification summary**

This chapter has described the verification approach to the Software Quality Perception Model proposed in the previous chapter. In the first step, the research methods were reviewed in the context of their suitability for the aims of SQPM verification. The important part of this review was associated with the explicit relation of identified differences between SQPM and the normative models. However, the identification of an appropriate research method required also the identification of variables, their values and manipulation techniques.

The verification requirement analysis was decomposed into three main experiments covering the differences mentioned above and the variables necessary to identify the actual perception processes. Additionally, as one of experiment required, a technique of deliberate manipulation of the quality level of a testable application version was proposed based on the fault probability function.

Having the hypothesis (the Software Quality Perception Model), and identified verification requirements, the author prepared dedicated experimental tools and conducted the experiments. These steps are described in part III of this dissertation.

### III. VERIFICATION

## 9 VERIFICATION: TOOLS AND RESULTS

Empirical research regarding software quality perception from the users' perspective was not analyzed in the literature, therefore the verification plan had to contain the design of the method, tools, experiment plan and other arrangements required to conduct a natural field experiment (Angner, et al., 2007). One of the most innovative aspects of the proposed research plan is related to the deliberate manipulation of the software quality level in such a way that the application is not changed but the quality level is controllably different.

In this chapter, the experiment plan is described, with a detailed description of the experiment variables, tools and environment setting. The track record of the experiments is then provided, followed by the research results. The final part of the chapter contains a discussion of validity issues and a summary.

### 9.1 Experiment plan

Detailed experiment design consists of the design of surveys (personal survey and post-evaluation survey) and the design of applications for the evaluation (named TestApps). The organization of the experiments is described in section 9.1.1.

Surveys designed for the purpose of this research consist of three sections. The personal survey consists of a set of questions about a subjects' experience (including questions regarding a subjects' domain experience), and a set of questions regarding their quality attributes preferences. The post-evaluation and "managers' survey" consists of a set of questions about the evaluation task performed (time spent, the number of failures observed etc.) and a set of questions regarding assessed quality.

The questions regarding quality preferences and assessed quality use the Likert-type scale presented in Figure 9-6 (self reported observations are assumed to be valid (Carver, et al., 2004), although they are perceived as being indirect measures of attitudes toward objects Alagumalai, et al., 2005). The question list is based on ISO/IEC 25010 (2011), which is assumed to be the normative model for software product quality assessment. The list contains the following criteria: rich functionality, compliance with formal rules, efficiency, productivity, satisfaction, learnability, adaptability, reliability, safety and security. Additionally, the subject is asked to assess the general quality of the product.

The ends of the scale are anchored to definitions. *Application is the absolute negation of this attribute* [value: 1] and *Application is the ideal example of this attribute* [value: 11]. At the mid-point, the neutral anchor is denoted as *Application has this quality attribute but not on an exceptional level* [value: 6]. The scale is intended to look like a continuous scale.

For the purpose of the experiments, four TestApps were prepared: an issue submitting system (TestApp1), an Internet banking system (TestApp2), and a brokerage house customers' application (TestApp3 and TestApp4). TestApp3 and TestApp4 have identical functionality. However, TestApp4 looks exactly like TestApp2, while TestApp3 has a different GUI. All graphical designs were based on real applications offered by banks and brokerage houses, although the names of the original institutions were replaced with artificial ones. For the purpose of evaluation, the complete documentation for evaluation purposes was prepared including: requirements description, test scenarios, detailed instructions etc.

TestApp1 was intended to be used for the purpose of the homogeneity test. Therefore, its functionality consists of only three functions: login, submitting a request, and a review of the submitted requests list. The application, when used by a group, allows other members of the same group to see the requests submitted by other group members. An example screen is presented in Figure 9-1. The documentation contains 15 requirements and 10 test scenarios.

03 sierpień 2010 [Rejestracja wniosku](#) | [Przeglądanie złożonych wniosków](#) | [Wyloquij test01](#)

Serwis eMiły

<p><b>BEZPIECZEŃSTWO</b></p> <ul style="list-style-type: none"> <li>» <a href="#">Zalecenia bezpieczeństwa</a></li> <li>» <a href="#">Bezpieczeństwo w systemie eMiły</a></li> <li>» <a href="#">Hasło maskowane</a></li> <li>» <a href="#">Konfiguracja przeglądarki</a></li> </ul>	<div style="background-color: #008000; color: white; text-align: center; padding: 5px; font-weight: bold; font-size: 1.1em;">Rejestracja wniosku</div> <p>Nr telefonu Klienta: <input type="text"/></p> <p>Imię i nazwisko: <input type="text"/></p> <p>Rodzaj Klienta: <input type="text" value="Indywidualny"/></p> <p>Tytuł zgłoszenia: <input type="text"/></p> <p>Treść zgłoszenia: <input style="width: 100%; height: 40px;" type="text"/></p> <p>Pilność: <input type="text" value="Nie pilne"/></p> <p style="text-align: center;"><input type="button" value="Zapisz"/></p>	<div style="background-color: #008000; color: white; padding: 5px; text-align: center;"> <p>Lokata Impet 3- 6- i 12-mies. nawet <span style="font-size: 2em; font-weight: bold;">7%</span></p> </div> <div style="background-color: #008000; color: white; padding: 5px; text-align: center;"> <p>Do wygrania ponad <span style="font-size: 2em; font-weight: bold;">5000</span> nagród</p> </div> <div style="background-color: #008000; color: white; padding: 5px; text-align: center;"> <p><b>OCiAC</b> Zniżka do 60% <a href="#">Oblicz składkę»</a></p> </div> <div style="background-color: #008000; color: white; padding: 5px; text-align: center;"> <p><b>Ulokuj nadwyżki</b> Lokaty dla firm</p> </div>
--	--	--

Kontakt: [Doradcy eMiły 61 111 222 333](#), [Doradcy kredytowi 61 444 555 666](#)  
 Zastrzeżenia prawne | **BANK DZIECIĘCYCH OSZCZĘDNOŚCI**

Figure 9-1 TestApp1 issue submission form in the Polish language (source: own study)



TestApp2 was used for the main part of research. Its functionality covers: login, information about the owned account, a transfer definition with the possibility of using pre-defined transfer patterns, transfer request validation and signature, reviewing submitted transfers, modifying or deleting a transfers whose due date is in the future, appending a signature to these operations, reviewing pre-defined transfer patterns, modifying or deleting a pre-defined transfer patterns, and reviewing the account's history. The application would be normally connected to a core system, but for the purpose of evaluation it is typical to use an emulation of such an interface. The emulator (as described in the evaluation documentation) executes all transfers on their due date and does not decrease the balance of the account. The application used by a group allows other members of the same group to see transfers, pre-defined transfer patterns, the account history etc. submitted by other group members. An example screen is presented in Figure 9-2. The documentation contains 24 requirements and 12 test scenarios.

Zalogowałeś się jako: test01 | Wyloguj | Pomoc

**PROFESJONALNY BANK**direct  
Bankowość bezpośrednia Profesjonalnego Banku

**BANK ZADOWOLONYCH KLIENTÓW**

---

- Strona startowa ▶
- Historia rachunku ▶
- Zarządzanie odbiorcami ▶
- Lista przelewów**
- Nowy przelew
- Regulaminy ☰
- Centrum formularzy 🎯
- Oprocentowanie %
- Prowizje i opłaty ★
- Kontakt 📞
- Pomoc ?
- Wyloguj ↻

**Pomocne informacje:**

**Dostępne środki** **22 500,00 PLN**  
Limit jednorazowy 1 000,00 PLN

**Definicja nowego przelewu:**

Konto obciążane ? Konto Super Biznes - 15 2130 0004 3001 0164 9458 000 ▼

**Dane odbiorcy:**

Wybierz odbiorcę ? ▼

**lub uzupełnij poniższe pola:**

Numer rachunku odbiorcy ?

Nazwa i adres odbiorcy ?

Tytuł operacji ?

Kwota ?

Data ? 3 ▼ sierpień ▼ 2010 ▼

Zapisz jako przelew wzorcowy ?

Nazwa przelewu wzorcowego ?

---

Polityka prywatności | Bezpieczeństwo | Kontakt
zalogowany jako: test01 | Wyloguj | Pomoc

Figure 9-2 TestApp2 transfer submission form in the Polish language (source: own study)

TestApp3 and TestApp4 were intended to be used in experiments of the second type (see section 8.4.2). The functionality of these applications includes: logging in, providing information about the brokerage account, presenting current share prices, the possibility of submitting buy and sell transactions within the requested price parameter, appending a signature to transactions, and reviewing executed transactions. Applications would be normally connected to a core system, but for the purpose of evaluation it is typical to use an emulation of such an interface. The emulator (as described in the evaluation documentation) executes all transactions if the buy price is greater than or equal to the current share price, and if the sell price is lower than or equal to the current share price. Applications used by a group allow other members of the same group to see transactions submitted by other group

members. Example screens are presented in Figure 9-3 (TestApp3) and Figure 9-4 (TestApp4). The documentation contains 15 requirements and 8 test scenarios.


The screenshot displays the 'Składanie nowego zlecenia' (New Order Form) in Polish. At the top, there is a navigation bar with links: 'Wyloguj', 'Mój profil', 'Mój rachunek', 'Moje skróty', and 'Waluty'. The user is logged in as 'Imię i nazwisko test01'. Below the navigation bar, there is a menu with options: 'Strona startowa', 'Zlecenia giełdowe', 'Nowe zlecenie', 'Rachunki', 'Waluty', 'Notowania', and 'Moje skróty'. The main heading is 'Składanie nowego zlecenia' with the subtext 'Aktywny rachunek: 00-99-300013'. The form is divided into several sections: 'Pomocne informacje:' (Helpful information) showing 'Limit należności Klienta: 48 582,43 PLN' and 'Limit wartości transakcji: 10 000,00 PLN'; 'Definicja nowego zlecenia:' (New order definition) with a dropdown for 'Dotyczy rachunku' set to '00-99-300013'; 'Dane transakcji:' (Transaction data) with fields for 'Kod waloru', 'Nazwa', and 'Bieżący kurs'; 'Kierunek transakcji', 'Liczba sztuk', and 'Limit ceny'; and 'Początek ważności' and 'Koniec ważności' with dropdowns for day, month, and year. At the bottom, there are 'anuluj' and 'dalej' buttons. The footer includes 'Polski | English | Deutsch' and 'PROTOTYP v0.1'.

Figure 9-3 TestApp3 transaction submission form in the Polish language (source: own study)

Zalogowałeś się jako: **test01** | Wyloguj | Pomoc

**DOM MAKLESKI direct**  
Dom Maklerski Profesjonalnego Banku

**DOM MAKLESKI ZADOWOLONYCH KLIENTÓW**



---

Strona startowa ▶

Zlecenia giełdowe

Nowe zlecenie

---

Regulaminy ☰

Centrum formularzy 🎯

Największe zyski %

Prowizje i opłaty ★

---

Kontakt 📞

Pomoc ?

Wyloguj 🔄

**Pomocne informacje:**

**Limit należności Klienta** **48 582,43 PLN**

Limit wartości transakcji 10 000,00 PLN

---

**Definicja nowego zlecenia:**

Dotyczy rachunku ?

---

**Dane transakcji:**

Kod waloru ?

Nazwa ?

Bieżący kurs ?

---

Kierunek transakcji ?

Liczba sztuk ?

Limit ceny ?

---

Początek ważności ?

Koniec ważności ?

---

Polityka prywatności | Bezpieczeństwo | Kontakt zalogowany jako: **test01** | Wyloguj | Pomoc

Figure 9-4 TestApp4 transaction submission form in the Polish language (source: own study)

### 9.1.1 Organizational design

Organizational design defined how the groups were to be selected, and assigned tasks and locations. For all experiments the subjects were located in physically separate locations. For the purpose of experiments 1 and 2, this assumption required that subjects were located in different cities to avoid the exchange of information between groups. Basili observes that it is easy to contaminate subjects (2007), and provides no clues as to how to exclude the potential inference of the anchoring effect (Tversky, et al., 1974). Therefore, the separation had to be strict. Additionally, in experiment 1 the test “managers” were located in different locations, and were able only to read the evaluation reports.

Experiments 1 and 2 were conducted among professional software evaluators. This assumption, and the requirement regarding separate locations, suggested a purposive sampling method. However, the profiles were to be assigned to groups randomly. Experiment 3 was conducted among doctoral seminar participants. Therefore, it is also classified as utilizing the

purposive sampling method. However, subjects were assigned to groups via the use of a randomization procedure.

Experiment 1 was intended to be run for 6 succeeding working days. It was planned to be initiated on a Friday with the TestApp1 evaluation task (although the personal survey was filled out before that day as part of the recruitment process). Next Monday the subjects were to start the evaluation of TestApp2 for five succeeding days. The subjects taking part in experiment 1 also participated in experiment 2 on the next Monday, and were asked to evaluate TestApp3 or TestApp4 (TestApps were assigned using a randomization procedure). On that day (just after the experiment 1), subjects who did not participate in experiment 1 were also asked to evaluate TestApp3 or TestApp4, which means that all groups participated in experiment 2 on the same day.

Experiment 3 was conducted during a seminar. Participants were asked to draw an assignment from an urn (figureheads were instructed previously to ignore the results and join the group that they were intended to join). Part of the group was then asked to move to another seminar room.

During the experiment, there were three summarizing meetings (see section 8.4.3). Therefore, an evaluation leader had to be chosen for each group. The experimenter selected one of the figureheads, who was asked before the experiment to sit on the first seat in a row, and was selected on that basis. The evaluation leaders interrupted the evaluation after 15 minutes and asked each participant about their opinion of the application, tasks that were performed, and any problems that occurred. Figureheads were selected to speak first, and their task was to praise or denigrate the application's quality level. After the third meeting, the subjects were asked to fill out individual surveys.

In all experiments, the real purpose of the experiment was revealed to the subjects at the end. They were asked to grant the experimenter permission to anonymously use their results. There were no consequences if this request was denied.

### **9.1.2 Communication techniques**

Communication techniques define the rules of communication during the experiments. All experiments were conducted simultaneously in all of the groups. Therefore, all external communications had an equal impact on all groups. This was also the general rule of the communication procedures during experiments; all communications contained the same information and were passed to all groups at the same time.

The most important information regarding tasks and applications was passed via the tool designed for the experiment management (“TestLab” – see section 9.3). When a subject received a task, they were able to open the page with instructions, documentation pertaining to the application, and further links to the application itself and to the post-evaluation survey.

Email was employed as an additional communication channel for the first and second experiments. Subjects received their task assignments and communications via email, although the emails hid the information about the other participants (each group believed that they were the only group evaluating the application). Responses to questions, suggestions etc. were posted to all participants.

During the third experiment, the communication procedure entailed the use of sheets of paper and eventually the spoken word. The task definitions were communicated via the sheets drawn by subjects during the randomization procedure. These sheets could also be used for the purpose of denying the request for permission to use data gathered during the experiment. The experimenter selected the leader of each group by speaking to the group and imposing an “ad-hoc” choice.

## **9.2 Experiment variables**

The variables considered in the context of planned research are divided into independent and dependent variables. Dependent variables were limited to judgment about the quality level of the evaluated product. This judgment was expressed as an opinion regarding chosen quality characteristics based on the SQuaRE model (ISO/IEC25010 FDIS, 2011). The details of the surveys are discussed in the previous section.

Independent variables represent environment settings and treatments applied during the experiment. The complete list of independent variables was not constructed. This is a consequence of the fact that an influence on the dependent variable is being investigated, which was not previously identified as a source of such an influence. Therefore, if the list was based on the commonly accepted normative approach, then it should exclude hypothesized influence from cognitive structures supplementary to the Software Quality Perception Model (see section 7.1).

The independent variables’ values relating to software product evaluation were set up based on common industry values. The evaluation task was defined, for the evaluators, as being a commercial project for a real professional evaluation service. However, differences could have occurred, and this was the subject of the homogeneity analysis (see section 9.6).

Analysis showed that the differences were negligible, and that the independent groups could be considered as equivalent (see section 9.5).

The summary of variables considered in the context of the research plan was based on the list of variables presented in Table 8-3. The summary presented in Table 9-1 lists the manipulation methods (or the values preset for the variables if they were not intended to be manipulated during the experiment).

ID	Variable	First experiment	Second experiment	Third experiment
Environment				
E1	Evaluators' team configuration	Structured teams with a "test manager"	Teams without leadership	Teams with test leaders coordinating meetings
E2	Collocation of team members	Each team was located in separate location; the manager was located in a separate location	Each team was located in separate location	Each team was located in separate room
E3	Quantity of the team	Each team consisted of four members plus a manager	Teams consisted of four to six members	Teams consisted of ten to twelve members, half of which were figureheads
E4	Evaluators' experience in evaluation	Professional evaluators took part in the experiment		Advanced users took part in the experiment
E5	Evaluators' domain expertise level	Evaluators declared to have rich knowledge in the domain		
E6	Attitude of evaluators towards performing evaluation tasks	Manipulated variable: the task was perceived as a commercial project, therefore the attitude was considered to be similar to typical. However, in half of the groups additional motivation was applied	The task was perceived as a commercial project, therefore the attitude was considered to be similar to typical	The task was defined as the evaluation of a new development framework, which was within the scope of participants' interests

ID	Variable	First experiment	Second experiment	Third experiment
E7	Approach of evaluators to evaluation tasks	Evaluators' comments suggest that they were involved in accomplishing the task		Evaluators got involved through the need to publicly express their opinion
E8	Parallel tasks of evaluators	The evaluators were performing other tasks during their business hours, which was typical for their type of organization		The evaluators were devoted to the evaluation for the duration of the experiment
E9	Workplace conditions	The task was performed on typical premises for these types of tasks		The task was performed during a doctoral seminar
Product related				
P1	Extrinsic quality of the product	Manipulated variable: the fault probability was manipulated between versions	The fault probability function was constant and set to 10%	
P2	Functionality	The functionality, ergonomics, learnability and clearness were constant; testable applications were copies of real systems		
P3	Ergonomics			
P4	Learnability			
P5	Clearness			
P6	Graphical design (GUI)	The GUI layout was copied from a real system, although the names of the real system were replaced with artificial ones	Manipulated variable: both GUI layouts were copied from real systems, although the names of the real systems were replaced with artificial ones	The GUI layout was copied from a real system, although the names of the real system were replaced with artificial ones
Evaluation process related				
S1	External information	All groups were performing tasks simultaneously and were physically separated to ensure that potential external information influence was equal		



ID	Variable	First experiment	Second experiment	Third experiment
S2	Evaluators' state of minds	The evaluation process was conducted on commercial premises, therefore the elements of evaluators' abilities to perform tasks were similar to typical evaluation projects		Manipulated variable: the evaluation was performed by advanced users who were encouraged to undertake the evaluation by the setting up of a semi-research context. However, figureheads presented positive or negative opinions about the product
S3	Evaluators' moods			
S4	Evaluators' synchronization with peak time	The evaluation was performed during normal business hours		
S5	Clearness of task scope	The documentation and scope were defined as per professional evaluation tasks.		
S6	Completeness of documentation			
S7	Data gathering method	Data was gathered through automatic processes and surveys filled out by subjects		

Table 9-1 Independent variables configuration for the research (source: own study)

The variable values presented in Table 9-1 indicate the manipulated and constant independent variables. The research could be extended to cover more areas of the possible values of independent variables. However, it was limited due to two main reasons.

First of all, according to the author's best knowledge, this research was the first scientific attempt to trace the relationship between non-technical aspects of the project and perceived software quality level. Therefore, the number of possible dimensions and independent variables to consider could not had been based on previous research. On the other hand, the research aims to verify the hypothesized Software Quality Perception Model proposed in section 7.1, which requires a limited number of experiments, and therefore a limited number of variables to be manipulated.

### **9.3 Experiment environment and tools**

In this section, the details regarding the technical aspects of the experiment plan are described. Specifically, this covers the tools prepared for the experiment, and the environment parameters resulting from the variables' manipulation requirements.

#### **9.3.1 Tools**

The research plan proposed in section 9.1 required adequate tools to support its execution. Research tools had to cover the following areas of experiment management:

- Subject management – the experimenter had to create profiles for the subjects (named or anonymous profiles), and eventually remove all data associated with certain profiles upon subjects' requests. The profile was important when experiments consisting of a set of tasks were executed, especially when sequential versions of an application were evaluated.
- Personal surveys management – homogeneity tests of independent groups as well as extensive results analysis required the gathering of additional information about subjects (age, sex, experience, declared preferences etc.), therefore the tool had to support dynamic surveys composition and the ascription of defined surveys to selected profiles (so the experimenter could build different personal surveys for different experiments).
- Manipulation of the application's quality level – experiments of the first type required a mechanism for the deliberate manipulation of an application's quality level. According to the plan (see 8.4.1), the experimenter used the fault probability function for this purpose. The faults presented in the application had to emulate real faults, offering a variety of fault types which had to be applied in a context aware manner (not all types of faults are effective in each action of the application; for example, a "save fault" is ineffective on the read action). The diversity of fault types settled an additional requirement for the tool: leveraging the frequency of each fault type when possible according to the user's actions and establishing a target distribution defined for the task.
- Deployment of testable applications – the tool had to allow the deployment of several applications at the same time for different experiments which could be conducted simultaneously. The tool had to offer an application programmable interface (API) for the testable applications of the features related to task management, faults

management and internal data management (applications were not allowed to be deployed with individual data models).

- Task management, recording the evaluation – the tool had to offer a feature to assign a task (the evaluation of a certain application or survey task) to a subject defining a set of steering parameters (application version, probability of fault, distribution of fault types, additional message to the subject etc.). During the evaluation, the actions performed by the subject had to be recorded. Additionally, the tool had to record the faults that occurred during the evaluation for comparison with target values and with subjects' feedback.
- Post-evaluation application's quality assessment management – the evaluation tasks could comprise a part where the subject was asked to fill out a survey regarding the quality level of the evaluated application. Like personal surveys management, the experimenter had to be able to dynamically define surveys and assign them to concrete tasks (different experiments conducted simultaneously could use different surveys)
- Support for the secondary perception tasks – the analysis of secondary perception was an additional task type which had to be supported by the tools. In this type of task, a subject (playing a role of “manager”) had to read the documentation of the evaluated application and afterwards be able to read quality level surveys filled out by their “team members”. The tool had to support the formation of such teams and support the gathering of feedback from “managers” in the form of surveys composed by the experimenter, as per the requirement above.
- General reporting – the tools had to offer a data reporting interface to the experimenter for analysis purposes.

Additionally, the tools required to conduct the planned experiments had to provide adequate performance, capability and reliability, because the aim of the experiments was related to the analysis of subjects' reactions to failures. If the tools had their own errors, then these errors could result in additional and unmanaged failures, which could spoil the results. This was significant, as the subjects could not be asked to perform the task for a second time. One of the hypotheses states that subjects' experience is enhanced during each interaction with application, thus after performing the task for the first time they acquire knowledge or their mental state changes. This hypothesis states that their performance would have been affected if they were asked to repeat it.

The tools may be in the future be used for international research, thus an additional requirement was multi-lingual support.

The set of tools designed and developed for the purpose of the proposed research was named “TestLab”. These tools should be regarded as a framework for conducting experiments covering all of the requirements stated above. A platform has been implemented with four applications (TestApps). The logical architecture of the platform is presented in Figure 9-5.

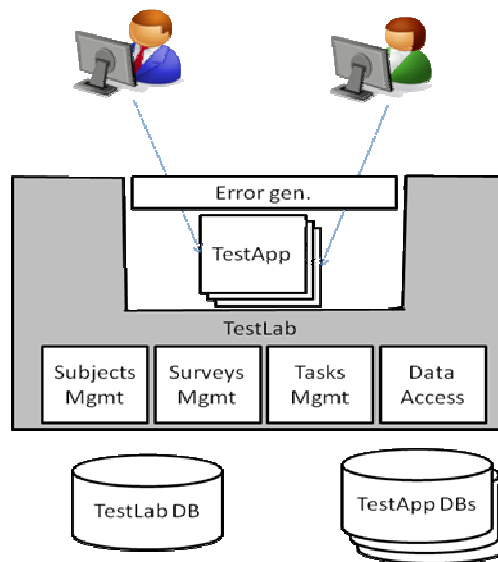


Figure 9-5 Logical architecture of the “TestLab” platform (source: own study)

The details of the design are not discussed in this dissertation, although the most important parts of the “TestLab” framework are presented. The first of these is the mechanism used to generate surveys. “TestLab” was equipped with the ability to assign an answer type to each question, and the survey’s designer may select one of the scale types proposed by Stevens (1959). “TestLab” allows for the selecting of an answer type from among the following formats:

- Input-text, input-text area – free text answer for comments on subjects
- Input-shaded – this type of free text input control appears as a password input field for use with answers which must not be seen by persons standing by the subject
- Input-integer, input-float, input-date – these types allow the experimenter to ask questions regarding various facts with answer format checking
- Radio, choice – these two types allow the experimenter to gather data with the use of radio-choice or a dropdown list
- Logical – this is a special case of the above types with only two possible answers: True or False

- Multi-choice – this type of answer may be used if more than one option is to be selected from among the possible answers
- Likert-type – this type of answer was specially designed for the purpose of “TestLab”. This type allows for the construction of interval or ratio scales (depending on the description of the anchors). The intended use is the expression of an interval scale following Osgood’s semantic differential (1957), with valuations having two bipolar terms at the ends. The scale is discrete (the experimenter decides on the magnitude). However, it is presented in a manner which simulates the continuity of the scale with the use of a gradient color. An example of this scale is presented in Figure 9-6.

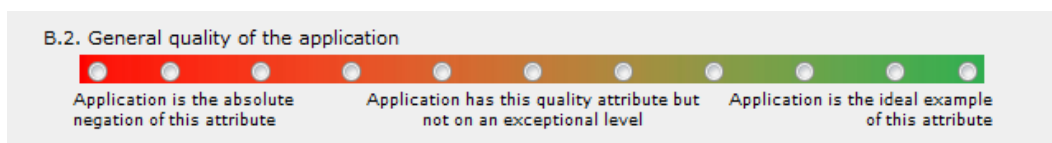


Figure 9-6 An example of the use of a Likert-type scale in the “TestLab” framework (source: own study)

A second important area of “TestLab” is related to the manipulation of the quality level. The design of this part of the application was initiated as a result of the analysis of failure types from real projects. According to the review results of more than 100 projects, the significant part of failure is related to requirements and design errors, which is consistent with Patton’s research results (2005). One of the assumptions regarding the quality level order (see 8.4.1) states that the application is not changed when failure occurs. Therefore, it is impossible to implement failures related to errors in the process design or application logic. These failures are also the most difficult to identify. Therefore, excluding them from the experiment should have no impact on the results.

The failure types identified during the review, which could be implemented in “TestLab” were:

- General failure – the application stops completely
- Efficiency – the application is unable to process requests within the required response time or loses the efficiency when the number of simultaneous users grows
- Data related failures – these are related to errors with the reading or saving of data
- Precision failure – this occurs when the result of computation is less precise than required
- Graphical interface failure – this is related to the incorrect presentation of an interface, errors in design etc.

- Maturity failure – this is related to the presentation in the application of the messages or texts which were used at the development stage by developers and contain debugging information

Based on the above list of types, twelve failures were implemented in “TestLab”. The mapping of these failures onto the above list is presented in Table 9-2 below.

Type of failure	Failure name	Severity
General failure	“Blue screen”	Blocking
Efficiency	Performance failure	Serious
Data related failures	Data lost on write	Critical
	Data lost on read	Critical
	Data change on write	Critical
	Data change on read	Critical
Precision failure	Calculation failure	Serious
Graphical interface failure	Presentation failure	Serious
	Static text failure	Normal
	Form reset failure	Serious
	Inaccessibility of functions	Serious
Maturity failure	Irritating messages failure	Low severity

Table 9-2 Faults implemented in “TestLab” (source: own study)

1. “Blue screen” - the application fails, producing literally a blue screen containing a vast amount of technical information about the error in the application. An example is shown in Figure 9-7.
2. Performance failure - after an action is initiated by a user, the application waits for 90 seconds before presenting the response.
3. Data lost on write - after a completed form is submitted to the application, the answer is “you have not completed the form”.
4. Data lost on read – when selecting data (a list of records or a single record), the application returns an empty information set.
5. Data change on write - after data is submitted to the application, it stores truncated data (for example “J” instead of “John”).
6. Data change on read – this type of failure is similar to number 5 above, but data is truncated only during the presentation. When the user requests data for the second time, they may receive correct information.
7. Calculation failure - the application returns an incorrect calculation result.

8. Presentation failure - the application presents the screen as a maze of objects (objects are located in their incorrect positions, and also have incorrect sizes, color sets etc).
9. Static text failure – the application randomly swaps static texts on the screen.
10. Form reset failure – when this error is generated, the user is unable to finish filling in a form on the screen. The form is cleared before completion (every 2-10 seconds).
11. Inaccessibility of functions – the application disables the possibility of using any of the active controls on the screen (for example, after filling out a form a user cannot submit it).
12. Irritating messages failure - the application shows sequential messages about errors but without any real failure. The user has to confirm each message by clicking on the “OK” button.



Figure 9-7 An example of “Blue screen” failure (source: own study)

An important aspect of the tool is associated with the algorithm that chooses the failure to be applied. As mentioned previously, not all types of failures could be applied to each function, therefore the algorithm had to be context sensitive, had to store historical information about generated failures etc. To enhance the manipulation ability of “TestLab” it was decided that the experimenter would use two parameters to control the failure generator:

- General fault probability in percent
- Vector of twelve weights associated with twelve failures

The algorithm idea used to generate failures is presented below with the use of pseudo-code:

```

Var numberOfActionsExecuted=getFromTaskContext('NOAE');
Var numberOfFailuresExecuted=getFromTaskContext('NOFE');
  
```

```

Var generalFaultProbability=getFromTaskContext('GFP');
Var <expectedWeights>=getFromTaskContext('EXPW');
Var <executedWeights>=getFromTaskContext('EXEW');
Var <failuresAllowed>=getFromApplicationContext('FA');

unless (userAction eq 'save')
{
    failuresAllowed[save*]=0;
}
Var tmpPlannedSum=sum(<expectedWeights>);
Var tmpExecutedSum=sum(<executedWeights>);
Var <tmpDeltas>;
Var tmpDeltasSum=0;
unless (tmpPlannedSum>0)
{
    numberOfActionsExecuted++;
    saveInTaskContext('NOAE', numberOfActionsExecuted+1)
    return 'NoFailure';
}
for (var i=0;i<count(<expectedWeights>);i++)
{
    if (failuresAllowed[i])
    {
        tmpDeltas[i]=2*expectedWeights[i]*(tmpExecutedSum/tmpPlannedSum)-executedWeights[i];
        tmpDeltasSum+= tmpDeltas[i];
    }
    else
    {
        tmpDeltas[i]=0;
    }
}
Var randomNumber=random(tmpDeltasSum);
Var generatedFailure=0;
Var tmp=0;
for (var i=0;i<count(<tmpDeltas>);i++)
{
    tmp+=tmpDeltas[i];
    if (tmp>=randomNumber)
    {
        generatedFailure=i;
        break;
    }
}
executedWeights[generatedFailure]++;
if (generatedFailure>0)
{
    numberOfFailuresExecuted++;
}
numberOfActionsExecuted++;
saveInTaskContext('NOAE', numberOfActionsExecuted);
saveInTaskContext('NOFE', numberOfFailuresExecuted);

```



```
saveInTaskContext('EXEW', <executedWeights>);  
return generatedFailure;
```

The leveraging algorithm has been evaluated in series of experiments. The main idea of the algorithm lies in the scaling of probability in accordance with the history of distributing failures in the context of the current request's boundaries. The algorithm is not exact, and the final distribution of failures is dependent upon the distribution of the requests' boundaries, although leveraging both the general failure ratio and the failures distribution allows for the efficient manipulation of  $f_p$  (see 8.4.1).

An additional feature of "TestLab" designed for the purpose of these experiments is the ability to define groups of users who have the possibility of exchanging data during the evaluation. This is a typical aspect of real evaluation tasks, where evaluators have to follow a set of rules to avoid disturbing other subjects.

"TestLab" underwent a series of tests, including performance and stress tests. The overall performance was measured using a laptop computer as a server (MacBook Air, Intel Pentium IV Centrino 1.4 GHz, 2 GB RAM, 80 GB HDD, Windows XP 3SP, Apache v2 with mod\_ssl, mod\_perl, MySQL v5.1, Ethernet 100 MBit/s) for 20 virtual users<sup>9</sup> (estimated equivalent of 140 real simultaneous users). The average response time was 1.5 seconds and the maximum time during the test was 3.5 seconds. These results were assumed to confirm "TestLab's" ability to be used in the planned research.

## 9.4 Experiment track record

The experiments were conducted according to the plan presented in section 9.1, and in accordance with verification objectives presented in section 8.4. The subjects for the first and second experiments were recruited from professional software evaluators, and the subjects for the third experiment were recruited from doctoral seminar participants. All subjects have declared a high level of domain knowledge (TestApp2 was intended to imitate an Internet banking application – a type of software product that was declared to have been used by the subjects for several years).

According to the plan, the first experiment began with a personal survey. Four independent groups were identified (located in Poznań, Wrocław (two separate sites) and Bydgoszcz), and four "managers" were appointed (located in Warszawa, Gdańsk, Poznań – second site, and Zielona Góra). The tasks (experiment conditions) were randomly assigned to groups. On the designated day (Friday), the evaluators were asked to evaluate TestApp1, and

---

<sup>9</sup> The typical ratio of virtual to real users assumed by the engineers who performed the evaluation was 1:7

then for five succeeding days they evaluated five sequential versions of TestApp2. During the evaluation of TestApp v 0.1, one of subjects from Bydgoszcz became ill and was excluded from the analysis (the group initially consisted of five people, therefore the experiment could be continued). Some evaluators did not fill out surveys at the end of the specific tasks. Their results were excluded from the analysis of the experiments, where the data from the complete series of experiments was required (see information provided with the results in section 9.5).

During the third day of the evaluation, when the quality level decreased significantly, evaluators generally assumed that they should discontinue the evaluation. They were asked via email to continue, and to accomplish as many tasks as possible (the email was the same to all groups). At each day, in the evening, the evaluators were sent the task for the following day with additional comments suggesting that the development team had improved something in the application (this was impossible to directly observe, but the aim was to simulate a real process). However, the only parameter changed was the number of the version and the fault probability parameter.

The second experiment was based on the groups participating in the first experiment and two additional groups. Once again, the tasks (experiment conditions) were randomly assigned to the groups, who received TestApp3 or TestApp4 to evaluate. Groups taking part in the first experiment were those who were familiar with the layout of TestApp4, while TestApp3 was new to all groups.

The third experiment was conducted two months later. Doctoral seminar participants were asked to select an assignment, and were placed in separate rooms according to the assignment. Figureheads taking part in the experiment also selected assignments, but had been instructed to go to specific room regardless of their drawn assignment (assignment allocation was conducted as a part of the figureheads' camouflage). In each room, the experimenter had appointed as leader the person sitting nearest to the door (the figureheads, as per prior arrangement). After first 15 minutes of evaluation, the leader asked each evaluator to describe the tasks they had performed, problems encountered, and their impression. Figureheads expressed their opinions first, while other participants generally agreed with figureheads. Then the second period of evaluation began, and once again the evaluators were asked to express their opinions publicly. After the third part of the evaluation, the evaluators were asked to fill out surveys, but without informing the other participants about their opinions.

The details regarding the gathered data and its analysis are presented in the next section.

## 9.5 Experiment results

In this section, the summary of the results is discussed. The empirical study consisted of three main experiments. The first experiment consisted of personal survey and tests of the experimentation methods, while the other experiments' homogeneity tests were based solely on surveys. The section's parts are devoted to discussion of the method's applicability for the conduction of the experiments, and the results of each experiment. Data is presented in this section in an aggregated form (the complete raw data set is presented in Appendix A). The raw data contains additional information gathered during the empirical study. Although it exceeded the direct objectives of this research, it is presented for the purpose of further studies of the results.

### 9.5.1 Method evaluation

The analysis of the results focuses on the general quality grade. The grade was assessed by subjects during the personal surveys, in the pre-test evaluation, and in the evaluations of each version after each evaluation task.

The type 1 experiment was designed to evaluate the research method (see validity discussion in section 9.6). Evaluation of the method focuses on the analysis of reactions to the manipulation of the application's quality level, and the assessment results in the situation where the presented version was the same. The positive analysis results verify the utility of the method to evaluate the influence of quality level manipulation, and also to validate the entire data collection process.

The method evaluation process consists of a series of tests reflecting the following statements:

1. If the groups represent a similar perspective on the software's quality, then the assessed quality grade of the same software product should be similar if no other interactions occurred
2. If the quality level of the product significantly changes, then the assessed quality level should also change in the same direction if no other interactions occurred

These statements were tested using following tests:

1. Homogeneity tests of the reaction
  - a. Analysis of the homogeneity of the perspective presented in personal surveys (the premises of the statement)
  - b. Analysis of the pre-test evaluation results in experiment 1
  - c. Analysis of the version \*.0.1 evaluation results in experiment 1

## 2. Analysis of the reaction

### a. Analysis of the transition from version \*.0.2 to \*.0.3 in experiment 1

The above tests were related to the assessment made by the evaluators and test managers. Data gathered during the experiment is presented in Table 9-3. The results are limited to evaluators who completed the following tasks: PersonalSurvey, TestApp1 evaluation, and TestApp2 versions \*.0.1, \*.0.2, \*.0.3, \*.0.5. The results were limited in order to preserve the ability to directly compare the results of the method evaluation and the experiment I results.

Stage of the experiment	A.L	A.H	B.L	B.H
Personal survey (evaluators)	10	9	11	11
	11	10	10	10
	11	8	10	10
	11	10	11	11
Personal survey (managers)	10	11	11	11
TestApp1 (evaluators)	7	7	6	8
	6	7	9	8
	9	8	6	8
	8	10	2	6
TestApp1 (managers)	6	7	4	6
TestApp2 version: *.0.1 (evaluators)	7	8	5	7
	6	8	9	8
	9	9	8	8
	6	6	5	4
TestApp2 version: *.0.1 (managers)	6	8	7	6
TestApp2 version: *.0.2 (evaluators)	9	9	4	8
	5	8	8	7
	7	9	8	9
	6	6	5	3
TestApp2 version: *.0.2 (managers)	6	8	7	4
TestApp2 version: *.0.3 (evaluators)	9	1	1	1
	3	3	1	1
	6	5	1	1
	5	2	2	1
TestApp2 version: *.0.3 (managers)	4	3	1	1

Table 9-3 Data gathered from the first surveys, the pre-test and the evaluation of versions \*.0.1, \*.0.2 and \*.0.3 in type 1 experiment (source: own study)

The main effects of the experiment were analyzed in the context of the joint groups (the “history effect” was expected by comparing the reaction of the A.\* and B.\* groups, and the “motivation effect” by analyzing the \*.L and \*.H groups). Therefore, the homogeneity tests were performed for both combinations of joint groups. The confidence level was set to  $\alpha=5\%$ . The results were interpreted using the ANOVA method (Shaughnessy, et al., 2005) and the null hypothesis verification procedure.

For the homogeneity analysis, the null hypothesis was assumed  $H_0: H_{A.*}=H_{B.*}$  and  $H_{*.L}=H_{*.H}$  for the analysis of the two groups’ results, as discussed above. The results from the analysis using the ANOVA method are presented in the following tables: Table 9-4 and Table 9-5.

Test	$M_{A.*}$	$M_{B.*}$	SS	SSE	F	p	$F_{crit\ 5\%}$
$H_0: M_{A.*}=M_{B.*}$ (personal survey)	10,0	10,5	1,0	10,0	1,4	0,27	4,6
$H_0: M_{A.*}=M_{B.*}$ (TestApp1)	7,8	6,6	5,1	45,4	1,6	0,23	4,6
$H_0: M_{A.*}=M_{B.*}$ (TestApp2 v: *.0.1)	7,4	6,8	1,6	35,4	0,6	0,44	4,6
$H_0: M_{A.*}=M_{B.*}$ (TestApp2 v: *.0.2)	7,4	6,5	3,1	51,9	0,8	0,38	4,6

Table 9-4 ANOVA table for homogeneity tests for  $H_0: M_{A.*}=M_{B.*}$  (source: own study)

Test	$M_{*.L}$	$M_{*.H}$	SS	SSE	F	p	$F_{crit\ 5\%}$
$H_0: M_{*.L}=M_{*.H}$ (personal survey)	10,6	9,9	2,3	8,8	3,6	0,08	4,6
$H_0: M_{*.L}=M_{*.H}$ (TestApp1)	6,6	7,8	5,1	45,4	1,6	0,23	4,6
$H_0: M_{*.L}=M_{*.H}$ (TestApp2 v: *.0.1)	6,9	7,3	0,6	36,4	0,2	0,65	4,6
$H_0: M_{*.L}=M_{*.H}$ (TestApp2 v: *.0.2)	6,5	7,4	3,1	51,9	0,8	0,38	4,6

Table 9-5 ANOVA table for homogeneity tests for  $H_0: M_{*.L}=M_{*.H}$  (source: own study)

There are no reasons to reject the null hypothesis for all of the above tests on the confidence level  $\alpha=5\%$ . Therefore, there are no clues that the groups were not comparable, or that the data collection process affected the results. It was also observable that the similarity between the groups was stronger regarding the TestApp2 version: \*.0.1 evaluation than in the TestApp1 application. This observation has the following implications: TestApp1 was an artificial application for a bank’s helpdesk (subjects could have no personal experience in using such an application), while TestApp2 was an Internet banking application (all of the subjects declared themselves to be Internet banking application users in the real world).

The second part of the method evaluation procedure compared the reaction to the quality level change. This test was performed by comparing the reaction to the quality level of

version \*.0.3 in the joint groups A.\* and B.\* (the homogeneity of the reaction for the quality level of version \*.0.2 was discussed above). The result was verified by an additional check of the reaction in joint groups \*.L and \*.H, which should be homogeneous.

Both results were analyzed with the use of the ANOVA method for testing the null hypothesis:  $H_0: M_{A.*} = M_{B.*}$  and  $H_0: M_{*.L} = M_{*.H}$ . The result is shown in Table 9-6 and Table 9-7.

Test	$M_{A.*}$	$M_{B.*}$	SS	SSE	F	p	$F_{crit\ 5\%}$
$H_0: M_{A.*} = M_{B.*}$ (TestApp2 v: *.0.3)	4,3	1,1	39,1	46,4	11,8	0,004	4,6

Table 9-6 ANOVA table for reaction to quality level change test for  $H_0: M_{A.*} = M_{B.*}$  (source: own study)

Test	$M_{*.L}$	$M_{*.H}$	SS	SSE	F	p	$F_{crit\ 5\%}$
$H_0: M_{*.L} = M_{*.H}$ (TestApp2 v: *.0.3)	3,5	1,9	10,6	74,9	2,0	0,18	4,6

Table 9-7 ANOVA table for reaction to quality level change test for  $H_0: M_{*.L} = M_{*.H}$  (source: own study)

The null hypothesis may not be rejected for the verification condition ( $H_0: M_{*.L} = M_{*.H}$ ), which means that although the reaction was significantly more diverse within joint groups, the overall result was comparable on the preset confidence level. However, analyzing the joint reaction to the quality level change (the condition  $H_0: M_{A.*} = M_{B.*}$ ), the null hypothesis has to be rejected. Therefore, the reaction is statistically different.

The second stage of the analysis requires an additional statistic to be calculated: Cohen's  $d$  (Shaughnessy, et al., 2005). The value of  $d$  for the comparison of A.\* and B.\* ("history effect") is  $d=1.31$ . According to Cohen (1988) this value is interpreted as a large effect.

The above analysis shows that the method and the observations made produced similar results when the evaluators evaluated the same application, and statistically different results when the evaluators evaluated the same applications but with differing quality levels. Therefore, the method is considered as having been evaluated in the context of the goals of the experiment.

### 9.5.2 Type 1 experiment

The analysis of the results focuses on the general quality grade. The grade was assessed by subjects during the personal survey, as well as after the pre-test and each version's evaluation. The first part of the experiment focused on the analysis of the groups' homogeneity and the evaluation of the method itself. The results were presented in the

previous section, while in this section the final results of the experiment are presented with the analysis of the secondary perception phenomenon. The data gathered during the final stage of the experiment (the evaluation of version \*.0.5) is presented in Table 9-8.

Stage of the experiment	A.L	A.H	B.L	B.H
TestApp2 version: *.0.5 (evaluators)	10	3	2	1
	3	3	4	2
	3	6	2	2
	4	3	1	2
TestApp2 version: *.0.5 (managers)	4	3	2	1

Table 9-8 Data gathered during the evaluation of version \*.0.5 in type 1 experiment (source: own study)

The reaction of subjects' teams seemed to be uniform except for one answer in the A.L group for the A.0.5 version of TestApp2 (grade 10 where all other subjects assessed the grade as being 3 or 4). The entry could have been removed if there were clues suggesting that it was entered mistakenly. The results were analyzed both with this entry and with its omission, and in both cases the conclusions were similar (the calculated values of  $F$  and  $d$  were different, but in both cases the result interpretation did not change).

The negative grades could have been affected by the floor of the scale. The negative end of the scale was described as the level where *the application is the absolute negation of quality*. Therefore, it seems that this effect could not be avoided. It seems that for negative emotions the evaluators selected the worst possible grade for the purpose of "punishing" the producers (compare the negative side of the Kahneman and Tversky (1979) curve).

For analyzing the main effects, a similar procedure to that presented in the previous section was followed. A null hypothesis was assumed  $H_0: M_{A,*} = M_{B,*}$  and  $H_0: H_{*,L} = H_{*,H}$  for the analysis of the two joint groups' results, as discussed in the beginning of this section. The analyses are presented in tables: Table 9-9 and Table 9-10.

Test	$M_{A,*}$	$M_{B,*}$	SS	SSE	F	p	$F_{crit 5\%}$
$H_0: M_{A,*} = M_{B,*}$ (TestApp2 v: *.0.5)	4,4	2,0	22,6	49,9	6,3	0,02	4,6

Table 9-9 ANOVA table for main effect test for  $H_0: M_{A,*} = M_{B,*}$  (source: own study)

Test	M* <sub>L</sub>	M* <sub>H</sub>	SS	SSE	F	p	F <sub>crit 5%</sub>
H <sub>0</sub> : M* <sub>L</sub> =M* <sub>H</sub> (TestApp2 v: *.0.5)	3,6	2,8	3,1	69,4	0,6	0,44	4,6

Table 9-10 ANOVA table for main effect test for H<sub>0</sub>: M\*<sub>L</sub>=M\*<sub>H</sub> (source: own study)

The null hypothesis has to be rejected because of the assumption that M<sub>A.\*</sub>=M<sub>B.\*</sub> (“history effect”). For the second assumption M\*<sub>L</sub>=M\*<sub>H</sub> (“motivation effect”), there are no reasons to reject the null hypothesis on selected confidence level.

An estimation of the effect size requires an additional statistic to be calculated: Cohen’s *d* (Shaughnessy, et al., 2005). The value of *d* for the comparison of A.\* and B.\* (“history effect”) is *d*=1.08. According to Cohen (1988) this value is interpreted as a large effect.

The results regarding insufficient arguments to reject the null hypothesis for the second effect (“motivation effect”) are discussed in section 9.7. The results are related to the professional character of the evaluation, where the motivation of employees is expected to be higher in comparison to the non-professional situation. Therefore, the additional treatment applied appeared to be insufficient to cause an observable difference among groups. However, the results should not be interpreted as a proof that additional motivation makes no difference in the professional environment. This topic is a potential future research question.

The manager’s opinion estimation was compared to the results of the evaluation of TestApp1 and all versions of TestApp2 (personal surveys were not analyzed, as the managers were not shown subjects’ personal surveys). For each version, the analysis is based on the evaluators’ reports of the test compared to the grades assessed by the managers. The comparison of chosen simple estimators is presented in Table 9-11.

	Estimator				
	Arithmetic mean	Geometric mean	<b>Harmonic mean</b>	Harmonic mean rounded	Median
Estimator value	4,77	4,53	<b>4,30</b>	4,29	4,69
Estimator error	0,48	0,24	<b>0,01</b>	0,00	0,40
Error std. dev.	0,84	0,76	<b>0,73</b>	0,78	0,92
Pearson’s <i>r</i>	0,94	0,95	<b>0,95</b>	0,94	0,93
Error range	2,75	2,24	<b>1,62</b>	2,00	3,50

Table 9-11 Simple estimators compared to managers’ opinions (source: own study)



The most effective estimator among those tested is the harmonic mean of the evaluators' answers. A version of this estimator where the values were rounded to the nearest integer (test managers provided answers on a discrete scale) was also tested, and the result was similar.

**9.5.3 Type 2 experiment**

The analysis of the results of the second experiment also focuses on the general quality grade. The research method was founded upon the assumption that the evaluation of graphically similar products influences their perceived quality level. The testing procedure included a homogeneity test of the groups based on personal surveys. The main effect was observed using the null hypothesis testing procedure.

The null hypothesis assumed that there were no statistical differences between groups A.C and C.C (refer to the experiment plan presented in section 8.4.2) in the context of evaluating the application in layout C. Consequently, the second null hypothesis assumed that there was no difference between groups A.A and C.A. The procedure is represented by an implication:  $H_{0(1)} \Rightarrow H_{0(2)}$ , or  $(M_{A.C}=M_{C.C}) \Rightarrow (M_{A.A}=M_{C.A})$ . The implication was shown to be false, which reveals the influence of the associations related to software quality on the basis of the GUI layout.

The empirical data from the second experiment is presented in Table 9-12. Groups A.A and A.C were analyzed on the basis of the first experiment's groups. However, the evaluators excluded in the first experiment analysis, who did not fill out any of the surveys at the end of the tasks, were analyzed in this experiment. They evaluated the application through the complete cycle (they skipped only the surveys in versions \*.0.1 or \*.0.5 of TestApp2, but completed the associated tasks), therefore their associations were established through these tasks.

Stage of the experiment	A.A	A.C	C.A	C.C
Personal survey	9	8	11	11
	10	10	11	10
	8	11	10	11
	10	11	11	11
	7	11	11	10
	11	11	-	11
	10	10	-	9
	10	10	-	-
	11	11	-	-
	-	11	-	-
Application evaluation results	6	6	9	4

Stage of the experiment	A.A	A.C	C.A	C.C
	3	6	8	7
	8	3	9	5
	5	3	6	6
	3	6	9	4
	8	3	-	6
	6	4	-	6
	2	6	-	-
	3	2	-	-
	-	7	-	-

Table 9-12 empirical record of second experiment (source: own study)

The homogeneity tests were performed between all pairs of groups. The results are presented in Table 9-13.

Test	First group M	Second group M*	SS	SSE	F	p	F <sub>crit</sub> 5%
H <sub>0</sub> : M <sub>A.A</sub> =M <sub>A.C</sub>	9,6	10,4	3,4	30,3	1,9	0,19	4,5
H <sub>0</sub> : M <sub>A.A</sub> =M <sub>C.A</sub>	9,6	10,8	5,0	15,0	4,0	0,07	4,7
H <sub>0</sub> : M <sub>A.A</sub> =M <sub>C.C</sub>	9,6	10,4	3,0	17,9	2,3	0,15	4,6
H <sub>0</sub> : M <sub>A.C</sub> =M <sub>C.A</sub>	10,4	10,8	0,5	16,9	0,4	0,53	4,7
H <sub>0</sub> : M <sub>A.C</sub> =M <sub>C.C</sub>	10,4	10,4	0,0	19,8	0,0	0,96	4,5
H <sub>0</sub> : M <sub>C.A</sub> =M <sub>C.C</sub>	10,8	10,4	0,4	4,5	0,9	0,37	5,0

Table 9-13 homogeneity tests for second experiment (source: own study)

All of the hypotheses presented in Table 9-13 could not be rejected on the confidence level  $\alpha=5\%$ , therefore it was assumed that the groups were statistically similar.

The experimental test procedure is based on the implication:  $(M_{A.C}=M_{C.C}) \Rightarrow (M_{A.A}=M_{C.A})$  (the assessment of TestApp3's quality is similar among groups regardless of whether or not they have evaluated A; therefore, the assessment of TestApp4's quality among groups should be similar regardless of whether or not they have evaluated A). The ANOVA results for both clauses are presented in Table 9-14.

Test	First group M	Second group M*	SS	SSE	F	p	F <sub>crit</sub> 5%
H <sub>0</sub> : M <sub>A.C</sub> =M <sub>C.C</sub>	4,6	5,4	2,8	36,1	1,2	0,30	4,5
H <sub>0</sub> : M <sub>A.A</sub> =M <sub>C.A</sub>	4,9	8,2	35,2	47,7	8,9	0,01	4,7

Table 9-14 treatment effect in the second experiment (source: own study)

It should be noted that the null hypothesis procedure allows for the rejection of the hypothesis on the predefined confidence level. Alternative result is always interpreted as a lack of evidence required to reject the null hypothesis.

The statement representing hypothesis in this experiment was redefined as:  $\neg(M_{A,C} \neq M_{C,C}) \Rightarrow \neg(M_{A,A} \neq M_{C,A})$  to show direct relation to typical null hypothesis testing procedures. There are no premises to reject the first null hypothesis. However, on the predefined confidence level the second null hypothesis has to be rejected. Therefore, the implication should be assigned with the values:  $(M_{A,C} = M_{C,C}) \Rightarrow \text{FALSE}$ . Although the testing procedure does not allow confirmation of the first clause, the ANOVA method allows for further logical implications. The empirical results are observable when the first hypothesis is true with a 30% probability.

On the other hand, the effect strength in the second comparison can be expressed by Cohen's  $d=1.31$ . According to Cohen (1988), this value is interpreted as a large effect (almost double the minimal value for this category). Therefore, it may be concluded that there were no significant statistical differences observed among groups during their first tasks. Both groups assessed the quality of TestApp3 (GUI in version C) on a similar level. However, the assessment of TestApp4 (GUI in version A, which had associations with the previous task for one group) has to be interpreted as statistically different among two groups. The effect size, used for external validity assessment (Shaughnessy, et al., 2005), indicates the strong effect size of the treatment.

#### 9.5.4 Type 3 experiment

The third type of experiment focused on the influence from group pressure. Two groups were randomly selected. However, in both groups half of the participants were figureheads who should be excluded from the analysis. The verification procedure was based on the null hypothesis testing procedure and the ANOVA method for inter-group variance analysis. The empirical data is presented in Table 9-15.

Stage of the experiment	POS Subjects	POS Figureheads	NEG Subjects	NEG Figureheads
Personal survey	9	11	9	10
	10	11	10	8
	10	11	10	11
	10	10	11	9

Stage of the experiment	POS Subjects	POS Figureheads	NEG Subjects	NEG Figureheads
	6	10	9	9
	9	1	-	-
Application evaluation results	4	7	2	3
	6	6	1	1
	9	4	1	1
	9	7	4	3
	7	4	3	3
	4	7	-	-

Table 9-15 empirical record of third experiment (source: own study)

The first verification was the null hypothesis regarding the homogeneity of the groups during the personal survey (figureheads' groups are omitted in the analysis).  $H_0: M_{POS}=M_{NEG}$ . The analysis is presented in Table 9-16.

Test	$M_{POS}$	$M_{NEG}$	SS	SSE	F	p	$F_{crit 5\%}$
$H_0: M_{POS}=M_{NEG}$	9,0	9,8	1,7	14,8	1,1	0,33	5,1

Table 9-16 ANOVA table for homogeneity test for  $H_0: M_{POS}=M_{NEG}$  (source: own study)

There are no clues that the null hypothesis should be rejected in the predefined confidence level  $\alpha=5\%$ . In the second verification step, the null hypothesis regarding there being no differences in the assessment of quality in both groups was tested. The results are presented in Table 9-17.

Test	$M_{POS}$	$M_{NEG}$	SS	SSE	F	p	$F_{crit 5\%}$
$H_0: M_{POS}=M_{NEG}$	6,5	2,2	50,4	32,3	14,1	0,00	5,1

Table 9-17 ANOVA table for main effect test for  $H_0: M_{POS}=M_{NEG}$  (source: own study)

The null hypothesis has to be rejected. Estimation of the effect size was based on Cohen's  $d=1.5$ . According to Cohen (1988) this value is interpreted as a large effect (more than double of the minimal value for this category).

### 9.5.5 Analysis of additional empirical data

The experiments aimed to trace the impact of certain circumstances on the general level of perceived quality. However, when the tools were being designed for the research, they were

equipped with additional logging tools, including parameters of execution and additional questions in the surveys regarding software quality characteristics. This empirical data, while not strictly related to the general research problem, was analyzed in the context of explaining the main topic.

In this section, additional conclusions are discussed. The scope of this discussion is related only to the analysis results where the result contradicts the normative view on software quality perception, or where the result may be discussed in the context of software evaluators' rationalism.

During the analysis of empirical data from the third experiment, the number of reported failures was compared with the real number of failures which appeared on users' screens. This data was compared with additional group tested in the same time, where figureheads were not only expressing negative opinions about the application, but also were disturbing (discussing off topic issues during the evaluation). The comparison is presented in Table 9-18.

Group	$f_p$	Real means		Users' subjective answers		
		Correct pages	Failures	Completed operations	Observed critical failures	Observed serious failures
POS	0,20	58,9	10,3	21,4	3,9	5,3
NEG	0,20	50,6	10,9	34,0	16,0	10,6
NEG+DIS	0,20	64,2	11,5	31,0	12,4	18,2

Table 9-18 Subjective number of failures reported (source: own study)

It should be noted that the total number of failures reported in the positive group was lower than the real number of failures that occurred. However, unexpectedly, the number of failures reported in the negative group was greater than the actual number of failures. The group with additional distraction factor reported the same overall quality grade of the application (group pressure effect), however their estimates regarding severity were different.

Another interesting observation was made during the analysis of the first and second experiment results. For all surveys regarding applications evaluation (TestApp1, TestApp2 versions \*.0.1 to \*.0.5, TestApp3 and TestApp4), the Pearson's correlation indicator (Shaughnessy, et al., 2005) was calculated as being between the general quality grade and other quality characteristics. It was expected that general quality was correlated with the pleasantness of the application ( $r=0.8$ ) or its reliability ( $r=0.75$ ). The unexpected correlation was between general quality grade and compliance with formal requirements ( $r=0.69$ ). Even after limiting the analysis only to TestApp2 (where only fault probability was manipulated) the indicators present similar phenomenon (for pleasantness  $r=0.87$ , for reliability  $r=0.8$ , for formal compliance  $r=0.7$ ). The expected result for this test (the normative approach) suggests

that there is no interaction between fault probability function and compliance with formal requirements. However, the empirical observation shows the opposite, corroborating the thesis of this dissertation.

## **9.6 Validity issues and discussion**

This section discusses validity. The research was the subject of major assumptions regarding subject selection, professional character of evaluation etc. Therefore, it is important to consider if the results may be considered as internally and construct valid, and also if they are applicable to other circumstances (i.e. are externally valid).

### **9.6.1 Internal validity**

Every experiment has to be considered against threats to its validity. The internal validity analyzes the probability that the result of the experiment is in a causal relation to applied treatment (Shaughnessy, et al., 2005).

In their book regarding field experiments, Cook and Campbell have compiled an exhaustive list of potential threats to validity (1979). The threats listed by Cook and Campbell have to be supplemented with a list of threats resulting from the consequences of the assumptions made for the research presented in this dissertation.

The first group of threats was associated with natural changes and the growth of experience among subjects. These threats were dependent upon the initial experience of subjects in the field related to the experiment, the length of the experiment, and the history effects affecting only some of the subjects. The selection of experienced software users and evaluators (especially for the longest experiments) was intended to mitigate these threats. Another threat to the internal validity was the possibility of external influence on the results (e.g. information in the mass media about a software security scandal, or even information causing strong emotions Zelenski, 2007), and the possibility of the uncontrolled flow of information between groups (e.g. a subject from one group could have shared their opinion about quality level with a subject from another group). To avoid these threats, the experiment was conducted simultaneously in all groups (the subjects would have had similar external information), but in physically separate locations.

Equally important to validity was a second group of threats associated with the data gathering process. In the designed experiments, the data gathering process was based on mechanical data gathering. Therefore, observer and experimenter effect were rather unlikely. Additionally, this threat was countered in the first experiment by having the pre-test and first

versions (\*.0.1) evaluated on the same quality level. The homogeneity check was used to compare the results, to verify if the reaction was similar in all groups. However, the threat associated with repeated measurements could be relevant for the results of the first experiment. Therefore, only professional software evaluators, who were used to repeatable assessments of software quality level, could be recruited for the purpose of this experiment.

The third most important factor affecting validity was related to the selection of subjects. The experiment plan used the purposive sampling method for each experiment, with the procedure of randomizing the assignment of tasks to groups (compare Harrison, et al., 2004). This type of experiment is considered as being characterized by high internal validity as a result of randomization (avoiding the regression to mean effect), and is conducted in similar to real conditions (Camerer, et al., 2003). The first and second experiments were performed among subjects who were professional software evaluators, and the third experiment was conducted among advanced software users. The population from which the subjects were recruited reflects the aim of the research. However, a threat to internal validity was related to potential systematic differences among groups. Therefore, homogeneity tests of the groups (revealed preferences, results of evaluation tasks, experience in the domain of application etc.) and the random assignation of tasks to groups were required.

An important source of threats was related to the experimental character of the task (compare the Hawthorne effect, Adair, 1984). The mitigation of these threats required a level of deception, and a definition of the task in such a way that subjects were not aware what was actually being measured (compare Angner, et al., 2007). Therefore, in all experiments the task was defined as an evaluation of a new kind of framework for rapid application development. This excluded the threat that the evaluation task could not be treated seriously (compare De Dreu, et al., 2006).

Communication in the evaluation tasks followed the typical patterns used in professional evaluation tasks, hiding the list of people taking part in the experiment (subjects had to assume that they were working as a team with a leader, as in real projects). During the third experiment, it was also important that subjects did not discover the existence of figureheads within the group. Therefore, figureheads drew sheets of paper to pretend the randomization of the assignment (i.e. that the leader was chosen spontaneously).

### **9.6.2 Construct validity**

Construct validity regards the theoretical ability of the experiment construction to reflect theoretical constructs in the hypothesis (Judd, et al., 1981). Roughly speaking, this dimension

of validity applies if the results reflect the judgment of the evaluators. An example of construct validity and its operational definitions for independent and dependent variables is presented in Figure 9-8.

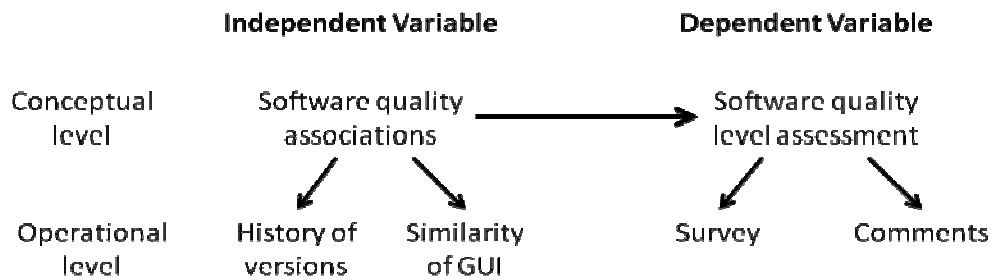


Figure 9-8 Conceptual and operational levels of independent and dependent variables (source: own study)

Typically, it is assumed that the measures reflecting subjects are accurate, and that the questionnaires provided by subjects are reliable and valid (Carver, et al., 2004). The mitigation of the risk associated with the potential unreliability of these method required the use of objective measures of experience (people tend to overestimate their experience). The risk associated with the reliability of the questionnaires was mitigated by the analysis of the sequential stages of first experiment and the internal convergence of different measurements (the set of questions from the survey). In this experiment, the quality level rose or fell in sequential stages, therefore it was checked if the assessed quality level followed this pattern.

The method of data collection could also affect the results. This threat was mitigated by the analysis of reactions to changing quality levels. However, it could not be fully eliminated. The scale was designed to present the most negative opinion at the lower end. However, it had been expected that when evaluators experienced negative emotions regarding the application then they would use the worst grade possible. Although this occurrence disturbed further data analysis, this empirical evidence supports the hypothesis that emotions affect judgment regarding software quality level.

### 9.6.3 External validity

External validity reflects the extent of probability that the results may be replicated in other circumstances (Shaughnessy, et al., 2005). Additionally, researchers analyze ecological validity, which reflects the ability of the results to be applied in real life situations (compare Sears, 1986).

A typical threat to external validity was related to a potential difference between a sample and the general population. However, for software products it is difficult to define the general



population of users, as software products are used directly or indirectly (e.g. in a mobile phone, in a DVD player etc.). According to List (2004), irrational judgment is more probable among naïve users than professional users, therefore it is assumed that effects observed among professional and advanced users are likely to occur among naïve users as well (compare Shaughnessy, et al., 2005). This decision could have an impact upon the sensitivity of the experiment. However, it increased the significance of the results. The external validity threat resulting from the selection of professional software evaluators was not mitigated. However, the results were discussed only in regard to commercial projects involving professional evaluators. Therefore, ecological validity should be perceived as being high, especially considering Simon's standpoint that in the majority of economic activities people act as employees (1995).

Another typical external validity threat results from the sample size and the possible lack of statistical significance. For experiments based on psychological research, external validity may be verified using the effect size (Mook, 1983). This observation uses a corollary that behavioral patterns are constant even in different situations (Underwood, et al., 1975), thus the size of the effect can be used to estimate the likelihood of the effect replication under different circumstances (Shaughnessy, et al., 2005). Natural field experiments are regarded as having high external validity, as they are conducted in circumstances similar to real situations (Camerer, et al., 2003).

The application of external validity to individual assessments was threatened by the design of the experiments, which assumed that the evaluation was performed by groups whose members could have exchanged opinions with each other. This threat could be mitigated by an experiment where evaluators would perform the same task in a group or individually. However, as this research aimed to analyze the evaluation in a typical commercial situation, it may be assumed that most software products are evaluated in groups or semi-groups (where people share opinions over discussion forums etc.). Nevertheless, an additional experiment was conducted during the evaluation of "TestLab", where master seminar participants were randomly split into two groups: one group performed the evaluation task in an environment where they could (and did) discuss the task, and the other group remained silent during the evaluation. This experiment was not part of the research aimed at verifying the Software Quality Perception Model, therefore the details were not discussed. However, the result has shown no significant differences between groups.

## 9.7 Empirical research results summary

The empirical data presented in this chapter supports the hypothesis that the actual software quality assessment process differs from the normative model. Irrespective of which normative model (see section 2.4) is compared, software quality assessment in the normative approach should not be affected by the evaluator's subjectivity. The results were obtained from professional software evaluators. Therefore, it is assumed that subjectivism was minimized. However, it was found that the impact of subjectivity on the results was significant.

The empirical results support the existence of the "history effect" (see section 8.4.1). This effect arises from the influence of the quality of previous versions of the software on quality assessment. By comparing these results with significant software products on the market, this effect may explain why software users still claim that Microsoft Windows operating systems have poor reliability, although it seems that its reliability is significantly greater than it was in 1991 when MS Windows 3.1 was released. The reliability of hardware, operating systems and applications has improved. However, software users (especially users who do not have software related education) still regard Microsoft Windows products as untrustworthy (SASO, 2010).

Partially surprising results were associated with the "motivation effect" (see section 8.4.1). Behavioral economics and psychological research results provide ample examples of the occurrence of the influence of motivation on cognitive processes, which are able to reverse the original effect (compare Baron, 2005 versus Asch, 1951 results). The results do not contradict the possibility of this effect occurring in a modified environment. However, considering Asch and Baron's results, it should be regarded as highly unlikely that this effect will diminish. The explanation of the results is associated with the professional character of activities, which underlines the need for further research focused on software quality assessment in real-like environments. Most behavioral economics research is focused on individual judgments and processes. However, making decisions on behalf of an employer or being responsible for the results in a professional manner was shown to influence the results (compare Ariely, et al., 2003).

In addition, the empirical results reveal the existence of the "association effect". The user interface was shown to have a significant influence on the assessed quality level. However, this influence is highly dependent upon the associations the evaluator has with a certain layout. This may be compared with the perception of the quality levels of two almost identical cars, which were launched on the automobile market in the late 1990's: the Volvo V40 and

the Mitsubishi Carisma. The cars were nearly the same, and in fact they were produced by the same company (Mitsubishi Motors owned the Volvo brand). However, customers often attested to the (high) quality of one of them over the (low) quality of the other (Kvist, 2004). Customers want to believe that they are immune to visual illusions, although the opinion that the Volvo V40 is more secure than the Mitsubishi Carisma was based purely on brand association. The “association effect” explains the success of quality certificates, the use of a graphical layout associated with a high quality product etc.

The empirical evidence has also shown that emotions influence quality level assessment. This influence was shown for both immediate emotions (when subjects lost their sense of rational quality assessment and reacted impulsively by “punishing” the application’s producer), and also for the anticipated emotions (when subjects anticipated how they would feel if they were embarrassed in front of the group).

Social pressure associated with anticipated embarrassment influenced spoken opinions when subjects were asked to express their opinion, although there was no contextual information which could have motivated subjects to conform. This may be perceived as sound, if the character of the subjects is considered, as well as their ability to participate in public discussions presenting independent opinions (in this research, subjects were recruited from doctoral seminar meetings and were used to expressing their opinions in public). A more important conclusion is associated with the opinions expressed by subjects after the experiment was conducted. Although they could have had an initial opinion about the quality of the evaluated software product, they “believed” in what they had heard from others. This was indicated in their own statements spoken in front of the group. In order to preserve their own internal consistency, they searched for evidence to support their new beliefs. In the final result, they presented a consistent opinion about software quality, the number of failures that occurred etc. (compare with Kant’s concept of *a priori* assumptions, Haden, et al., 1981).

Also, attention was shown to have an effect on the soundness of subjects’ beliefs about the application. When evaluators were distracted, they worked faster and spent less time analyzing the application’s behavior. However, they memorized faults with milder emotions than a comparable group where distractions were not applied. This caused these faults to be classified differently. This observation suggests that the above mentioned *a priori* beliefs may be affected. Low quality of the evaluated application has been suggested to subjects in this experiment. Although the objective quality was higher than in version B.0.3 in experiment 1, subjects assessed its quality as belonging to the lower end of the scale, therefore it is difficult to precisely estimate the strength of this effect.

A clearer observation relates to subjects' holistic view of the product's quality. In opposition to the majority of product quality models (which assume that the observer assesses the values of individual attributes, combining observations into a conveyed general conclusion about the quality level), the empirical results from all experiments suggest that the opposite is in fact the case. Among assessed quality attributes, the one representing conformity with external law at the axiological level should be independent from the density of faults, associations with the layout and social context. However, the empirical results reveal a strong correlation between assessed attribute value and general quality (a stronger correlation exists only among two attributes and the general quality level: pleasure associated with the application's use, and the product's reliability). This observation may be compared to learning processes, wherein learning about a new object starts with a general overview, after which the details are extracted (although this process may be affected by general attitude (compare Nečka, et al., 2008). It also supports the idea that experts look for similarities between new observations and known objects, and categorize new objects based on their assessed similarity to previously known ones (compare Simon, 1987).

Another important observation regarding calls for a separate research area related to the differences between decisions made upon description or experience (compare Hertwig, et al., 2004) may be made based on the empirical data from first experiment. Two types of subjects assessed the quality based on empirical observations or written reports (judgment from description). Their conclusions regarding the quality level of the assessed applications were, however, similar - in opposition to the modern findings of behavioral economists (compare Erev, et al., 2010). This phenomenon may be explained by the high level of the stakes involved (compare Parco, et al., 2002), or the professional character of the evaluators (compare Ariely, et al., 2003) etc.

It also seems, in contrast to natural expectations, that the process of integrating different opinions into one's own opinion about software quality reflects harmonic weighting rather than arithmetic weighting. Based on the experiments' records, it may be shown that the most effective estimator<sup>10</sup> of a manager's opinion is the harmonic mean of the evaluators' opinions. The harmonic mean result of positive values is the lowest among those analyzed (see (Hofman, 2011) for details). This finding supports Nisbett et al.'s conclusion that people tend not to use statistical reasoning in processes associated with cognitive analysis (1983).

---

<sup>10</sup> Among simple estimators

The aforementioned results should be considered in the context of the validity assumptions that were made (see section 9.6). The most important limitation was related to subjects' selection processes and the general settings of experiments. Therefore, the results should be regarded to the context of professional evaluation processes (e.g. employees who perform evaluations on behalf of their employers) and the professional character of the software product (i.e. the product for professional purposes). Following a typical configuration, the subjects worked in teams. However, the size of the teams was limited (in the industry, projects occur where the evaluation team size is 10 times larger). In analyzing the internal and external validity of the results, it should be noted that the restrictions mentioned above do not allow the application of the results to other contexts without relevant consideration of contexts' differences (e.g. the individual assessment of quality of an entertainment COTS product performed by a teenager). However, they do reflect typical business to business software delivery, which in turn reflects the general goal of this dissertation. Additionally, the research methods proposed in this dissertation may be applied to other conditions. Therefore, it will be possible to analyze the results caused by different restrictions sets in the future.

The results of the research support the theoretical Software Quality Perception Model proposed in chapter 7. This model may be enhanced by adopting concepts related to immediate or anticipated emotions, recent and permanent knowledge, the hierarchy of needs etc. In the future it is expected that models with higher predictive accuracy will be proposed and evaluated. However, in regard to current state-of-the-art theory, descriptive models of software quality perception do not exist. The software industry is focused on justifying the usage of normative models as descriptive ones (compare Stavrinoudis, et al., 2005, ISO/IEC25010 FDIS, 2011).

The assumption regarding the use of normative models as descriptive software quality models would have caused negligible differences among different groups in all of the experiments described in this thesis, because these models rule out that the overall quality may be influenced by personal experience, emotions, needs saturation, memories, associations, social context etc. The results have shown that these factors actually affect the perceived quality level. Therefore, the model adopting the behavioral approach will be able to predict judgment and associated decision processes with a higher degree of accuracy than normative models.

The summary of the above positively verifies the thesis of the dissertation.

# CONCLUSIONS AND OUTLOOK

## 10 CONCLUSION

This final chapter provides the conclusions and outlook for this thesis. Finally, the open issues that can be tackled in future research are mentioned.

### **10.1 Applicability of the results**

The mainstream approach to software quality modeling on the software market is based on normative software quality models. These models are used to define quality and to predict the quality assessment value that should be assigned by an independent evaluator. However, the accuracy of these predictions has not, until now, been investigated.

The research results presented in this dissertation emphasize the descriptive inaccuracy of normative software quality models. Although behavioral economics tools and methods have been used since the 1970's, their application to software quality was restricted mainly to simple judgment and decision making processes. Therefore, this research is opening up a new approach to software customers' descriptive analysis.

The results demonstrate the importance of the context and environment associated with software product delivery for the process of quality assessment. Unlike normative models, the actual quality assessment process does not seem to be objective; additionally, evaluators are not supplied with complete information about the products they are evaluating. The customer makes their judgment even though they may or may not be aware that they are making a biased judgment, or that they are overlooking important information about the product. In this section, the results' practical application is considered. The analysis of the application was preceded by a statistical analysis of software releases in a probe of 15 projects with a total budget exceeding US\$250 million (see Table 6-1 in section 6.1). This analysis reveals a common practice: a reliance on the positive assumption that finally the customer will evaluate the correct version and will assess its quality irrespective of previous versions. The empirical evidence shows that this approach is risky, because evaluators actually do assess the quality of the final product bearing in mind all the knowledge and emotions they have gathered during the project.

The research results related to the software market form a set of postulates, which may be the key to understanding customers and satisfying their needs, whilst also building the competitive advantage of the vendor in a cost efficient way.

1. The rationality and objectiveness of evaluators should not be assumed

The research results show that, in contrast to the normative approach, evaluators base their opinions on associations, emotions etc. Their attitude is justifiable, because they are evaluating complex, multi-attribute products (Hochstein, et al., 2008) where most of the attribute values are beyond their measurement ability (e.g. the security of the product). On the other hand, the combination of several attribute values is a computationally complex task - especially if one assumes that each attribute is compared to a reference point and that its contribution to combined value is assessed in terms of gains and losses (compare Köszegi, et al., 2006). In this approach, even if evaluators have complete information about the attributes, it is highly likely that they will not be able to compute them precisely (Kahneman, et al., 1979). There are several other arguments based on empirical research results which show that software evaluators are unable to be rational in terms of normative models (however, their behavior may still be regarded as rational in cognitive terms).

A practical use of this postulate may be put into a description of attitude which assumes that customers are human, and should therefore be approached like humans. This postulate is the key to all the following ones.

2. Saturation levels of identified customer needs should be assessed during the requirements gathering phase

The software engineering approach treats customer representatives as the ultimate source of information about requirements, their importance, their adequacy to future operational requirements etc. However, bearing postulate 1 in mind, it may be concluded that this approach is inaccurate. There are three main reasons for this: firstly, people are overconfident that their own point of view can be generalized to the group they are representing, therefore stakeholders present their own point of view in the certainty that their viewpoint is shared by others (especially by the evaluators who will accept the product) (compare Kunda, et al., 1988). The second problem is associated with hyperbolic time discounting (Loewenstein, et al., 1992), which explains inconsistency in behavior over time. In long projects, stakeholders may express their rational opinion about the future product. However, when the acceptance date approaches, they may be willing to simplify the costs and effort associated with its implementation (this problem is further discussed in the next section). The third problem is associated with ongoing changes in the customers' environment and in the natural process of preferences modification, which may lead to a different conception

regarding the business processes that are to be supported by a software product, and consequently to different needs associated with the product.

3. Management of the product outlook should not be overlooked regardless of the product's technical quality

Implicit in this postulate is an observation that a software product's quality is typically understood as being negatively correlated to the number of errors and failures resulting from these errors (compare Patton, 2005). Considering the limitations in software measurement processes, it may be observed that when failures occur, evaluators believe that they have found evidence that there is an error in the product. However, when they do not encounter failures, this does not mean that the product is error free. If the vendor built the belief about errors (delivering a version with low quality), then even if they have delivered a high quality version, the evaluators may still believe that there are errors in the product, although better hidden. Although not the subject of this research, it may be speculated that this rule is valid for all complex products delivered during software projects, including requirements documentation, design documentation etc.

4. Customer's associations mechanisms should be learned in order to address their associations with new products

The research results have shown that associations play an important role in the quality assessment process. The way the product will look will trigger positive or negative associations, therefore it is advisable to analyze the outlook of products, designs etc. the customer has contact with to identify positive and negative patterns. The same rule should be used to identify preferred compatibility requirements (e.g. Linux Friendly), components (e.g. Powered by Oracle), or even certificates (e.g. Secured by VeriSign).

5. Users' positions, fears, hopes and other limitations should be assessed

Consequently to postulate 1, one should also analyze other aspects related to the product which may affect quality assessment. For example, the implementation of the product may be associated with an organizational change, which will cause a reduction in employment, or there may be other reasons why employees are reluctant to adopt new processes. On the other hand, they may be expecting an organizational change because they anticipate that this change and the new product will solve a certain set of everyday problems. A more shallow limitation may be associated with the learning process itself: when people use a particular system for a long time, they are used to performing a certain sequence of actions without cognitive effort (automation of



actions, compare Nęcka, et al., 2008). When they evaluate the new product, they have to employ the higher structures of their brain, which is associated with cognitive effort. Comparing the old and new product, the evaluators may conclude that the new one is much more difficult and less intuitive (compare contrast effect Nęcka, et al., 2008).

6. Difficult situations have to be managed properly

There is an important stream of research. Results show that people tend to memorize not only facts, but also emotions. Several experiments have revealed that the continuous stimulation impression was memorized depending on the level of emotions at their peak and at the end (compare Redelmeier, et al., 2003). In the software delivery project, there are several situations when customer representatives' emotions may escalate. Typically, the vendor does not care, and is willing to solve the situation after some time. However, in this approach they lose the possibility to leave positive memories, which could be useful in encouraging future co-operation.

The above postulates do not cover all of the issues related to software production and delivery. However, in the author's opinion, they address problems that commonly occur in the software market. These problems lead to misunderstanding the vital needs associated with a new product, the rejection of high quality products, or dissatisfaction with products. Eliminating these problems may improve the competitive advantage of vendors who adopt the behaviorally based approach. However, more importantly, it will improve the size and meaning of the software market as a whole according to George Akerlof's predictions (2003), and research results regarding uncertainty (Curley, et al., 1986).

## **10.2 Analysis of the thesis and objectives**

The thesis of this dissertation aimed to provide a solution to the problem of the predictive accuracy of quality assessment models. Normative models of software quality were identified and described, and their applicability for the research problem was analyzed (see section 6.2). The normative approach was compared with descriptive models of behavioral economics, and a non-exhaustive list of violations assumed by the normative model approach was identified. However, these empirically based models do not provide unambiguous causal relations between effects and independent variables, therefore secondary research results could not be used to resolve the research problem.

Therefore, this dissertation utilized the behavioral economics research model (see List, 2004). Behavioral economics methods have not been used before for the purpose of software

quality assessment analysis. Therefore, for the purpose of thesis verification it was necessary to propose and evaluate relevant research tools and methods. The method required the preparation of a research plan, which may be used for further research efforts. The tools and parameters of the environment had to be recognized, and manipulation methods to control them had to be proposed.

The objectives of this dissertation resulting from the general objective addressed the following areas (compare section 1.2):

- 1) Identification of the variables impacting on the software quality assessment process during the perception process

Software market observations and conclusions regarding the actual quality assessment processes were described in section 6.1. The empirical observations showed that there are a set of issues which cannot be explained on the basis on the commonly accepted software quality models. These models bind quality with the inherent attributes of the product, user's needs, and the context of use. However, in several cases the same product used by similar organization (comparable needs and context of use) is assessed differently, and the root cause of such differences was hypothesized to be ingrained in cognitive processes. The areas of influence were identified on the basis of secondary empirical reports, and were used as the basis for the formulation of the Software Quality Perception Model.

- 2) Development of the descriptive Software Quality Perception Model

A descriptive Software Quality Perception Model was proposed in section 7.1. This model was based on the normative approach, and was extended via the inclusion of research results from behavioral economics research. Eight important differences between the proposed model and (the commonly accepted) normative models were identified. This list outlines the influence of attributes not related to the product on perceived quality, the existence of the attention filter in the process, the influence of knowledge and mental state on the attention filter, the distinction between the influence of the knowledge and of the mental state of the observer, the influence of the perceived attributes on knowledge, the distinction between needs weights and needs saturation in the perception process, the logistic function used for the overall quality level calculation, and the influence of the overall perceived quality level on mental state and knowledge. Identified differences were used for the adjustment of the research method (see Table 8-2).

- 3) Elaboration of a research method for the verification of the model

The research method was proposed in section 8.4. This method had to be evaluated against validity threats (see section 9.6) In particular, the construct validity addressed the question of whether or not the research method itself was reliable. The results based on the operational definitions of variables reflect interactions in the real world.

The analysis of the research method was planned to be performed in experiment 1, where the differences between groups could have been eliminated due to a series of measurements in each group. The external influence in each group was of the same nature: evaluators performed identical tasks, participated in an identical number of measurements etc. Therefore, it may be assumed that the observed effect was the effect of treatment applied in the experiment. Other parameters of the research method were designed according to a set of good practices for this type of research (the use of self reported attitudes, Carver, et al., 2004, semantic differentials, Osgood, et al., 1957, and the range of the scale Duckworth, et al., 2002).

- 4) Elaboration of methods for the manipulation of the environment configuration to emulate occurrences taking place in the software market

Experimental causal analysis is naturally threatened by confounding unless strict control methods are employed to manage all relevant variables during the research. During a software delivery project, occurrences associated with people, processes, external information etc. may be applied following behavioral economics experimental paradigms. However, if the research is to investigate the influence of the software quality level then a relevant method has to be expounded.

Software quality level comparison encounters the problem of a multitude of attributes being associated with a software product. Therefore, the comparison of quality between two (or more) products in most cases leads to unambiguous conclusions. The two groups who evaluated TestApp3 and TestApp4 in experiment 2 reveal this problem clearly: exactly the same functionality was assessed differently. The assessed quality in each level was dependent upon the evaluators' associations.

The normative definition of software product quality describes general quality assessment as a different process from the assessment of the functionality, learnability, productivity etc. of a software product. It is natural that when an application is being changed in the smallest way (e.g. a change related to one

control on a form), it may improve quality assessment for some users, or lower its assessment (or be irrelevant) for others. Therefore, it is difficult to manipulate the application's quality level and to correctly anticipate the quality level change.

The method elaborated for the purpose of this research was based on the fault probability function, and the observation that quality is negatively correlated with fault probability. This method has created an ability to manipulate variations of the variable, which is typically perceived as the key indicator of software quality (compare Patton, 2005).

The correct application of fault probability manipulation required the ability to manifest a fault's occurrence. A set of typical faults based on an extensive review of industrial projects was generated (see Table 9-2), along with a leveraging algorithm (see section 9.3).

#### 5) Elaboration of the required research environment

The research method that was developed for the purpose of this dissertation required the formulation of a dedicated set of tools. The assumptions and experiment boundaries (see section 8.4) - especially those related to the semi-automatic assignment of tasks, the collection of data, and the experiment's simultaneous execution in several cities - have stated requirements for tools. The most important functional requirements were grouped in the following areas: subject management, personal survey management, the manipulation of the application's quality level, the deployment of testable applications, task management, the recording of the evaluation, post-evaluation application's quality assessment management, support for the secondary perception tasks, and general reporting (see section 9.3.1). Special attention was given to the data collection ability of the tool. A set of scales and answer types was designed to reflect all possible requirements related to the data collection process in research based on surveys. Non-functional requirements covered the performance, reliability and security of the application.

During the experiments it was necessary to use 4 testable applications. These applications were duly prepared. TestApp1 was used for a pretest task in experiment 1, TestApp2 was used during the longest evaluation task in experiments 1 and 3, and TestApp3 and TestApp4 were used in experiment 2.

#### 6) Execution of the verification and the assessment of the proposed model

Data analysis and the conclusions were presented in section 9.5. The analysis has covered issues related to experimental internal validity (e.g. homogeneity tests of groups), construct validity (e.g. analysis of reactions to quality changes), and external validity (e.g. effect size). The aim of the analysis was to verify the Software Quality Perception Model, which was related to the research problem stated in section 6.1).

The above list presents the results of the research in respect to its goal and its decomposition to sub-goals. A second perspective on the research task is associated with the research problem statement and the measurable research questions stated. The thesis of this dissertation was decomposed into three research questions, which were assumed to have a positive answer (see section 6.1).

1. Is it possible to prepare a relevant method for the purpose of descriptive research among software products users?

The answer is positive because the method has been proposed and evaluated, confirming its relevance.

2. Is it possible to prepare a relevant method for the purpose of setting up and manipulating the research environment?

The answer is positive, because in this dissertation a method allowing the manipulation of variables typical of behavioral economics and also the most common variable associated with software delivery projects was elaborated upon and evaluated.

3. Is it possible to construct a descriptive model of users' behavior and to verify it using the prepared research methods?

This answer is also positive because the model proposed in this dissertation was evaluated and shown to be more accurate than existing normative models of software quality.

The above summary allows researchers to assess the validity of this dissertation's thesis: *The model of customer assessment processes related to software quality, which takes into account phenomena described by behavioral economics, allows for a more accurate prediction of customer decision than commonly used normative models.* According to selected research method, **this thesis was shown to be correct.**

The research related to understanding the actual processes of software quality assessment by customers opens up new possibilities for vendors in the market to understand and satisfy their customers, and to thereby effectively improve their competitive position. Considerations regarding the application of the results were presented in section 10.1. However, it should

be noted that this research is pioneering in regard to software products. The results give rise to further research directions, and these are described in the following section.

### **10.3 Further research**

There may exist several areas of influence which affect the quality level perceived by customers and users. Behavioral economists have reported an extensive list of biases which have the potential to influence the final assessment of a software product's quality, and by these means influence the decision regarding its acceptance. The identification, classification and modeling of the adoption patterns of descriptive methods pose a challenge for coming years. Behavioral economics has proven its usefulness for the understanding of market decisions, and the application of its achievements and research methods in software engineering opens up new possibilities for understanding and satisfying customers.

It is expected that a code of good practice for companies delivering software will be developed, following an investigation of several phenomena. The research executed for the purpose of this dissertation represents a limited set of potential sources affecting software quality perception. Other influences need to be investigated.

Although in professional activity there are no systematic problems with attention, it is still unknown how attention related processes influence the quality assessment level. Starting from James, who has described attention as an aware concentration on an object, research regarding attention has not yet provided the answers to many important questions. Current research analyzes the factors of objects attracting attention, levels of attention appearance, and parallelism in the attention processes (Nęcka, et al., 2008). Researchers mostly agree that attention processes take place, that the main role of attention is to relieve the cognitive system from processing too much information and that attention happens at more than one level. An experiment described by Triesman (1973) proves that within the cognitive system there is only one channel for cognitive processing, and consequently only one attention filter on this level. Attention may influence the whole encoding process. For example, in an experiment investigating double encoding, the subjects were given the same information in different modal representations sequentially (Paivio, et al., 1973). The results showed that the repetition of a message in dual channels resulted in a higher level of information retrieval. Bransford (1972) has shown that the meaning of information is memorized better than the form in which it was presented. The support of software products evaluation with training, meetings etc. that draws more attention to repeated factors of the product is an area that necessitates research.

In 1993, software engineering scientists were calling for the experimental evaluation of software related theories in specialized laboratories. Their concern was the understanding of processes and the building of descriptive models for explaining the problems related to the software construction process (Basili, 1993). This direction is still perceived as being the future of software engineering (Sjøberg, et al., 2007).

Akerlof (2003) calls for the extension of behavioral analysis to macroeconomics and the majority of economic analysis, as behavioral economics improves the explanatory power of economic research methods (Camerer, et al., 2003). Nowadays, his call has been answered in several areas, yet not by the software market.

For example, according to Mellers and McGraw (2001), it may be expected that during evaluation, the anticipated pleasure (or pain) associated with using the product will be considered by evaluators. However, evaluators often make their decision on behalf of others (compare Simon, 1987, Marshall, et al., 1986). Although this may mean that emotions will be analyzed, they may be anticipated inaccurately or omitted in the decision making process. An important direction is associated with stressors: multiple information sources, incomplete and conflicting information, rapidly changing or evolving scenarios, requirements for team coordination, performance and time pressure, high work or information load, threats or social pressure etc. (Cannon-Bowers, et al., 2000) and their influence on quality assessment.

Time difference between the analysis of requirements and acceptance testing is also a potential source of risk. According to the hyperbolic time discounting model (Loewenstein, et al., 1992), there is an inconsistency between planning for the future and actual decision making. Therefore, stakeholders may agree to reasonable tradeoffs for the future, although they will reject them on acceptance. A good example is provided by empirical results from an experiment where participants were asked to choose between “lowbrow” and “highbrow” movies to watch (Read, et al., 1999). The result depended on the time horizon for the planning: subjects choose “lowbrow” movies for immediate watching and “highbrow” movies for watching in the future. Similar problems related to the time gap between gathering requirements and acceptance testing may be caused by the misprediction that current preferences will last for longer than they actually will (Loewenstein, et al., 2003). An example of such misprediction is shown in experiments where participants were to predict the endowment effect (Van Boven, et al., 2003). Such misprediction and the occurrence of this effect during the acceptance testing phase may form a barrier for the acceptance of new products. These changes in attitudes may be perceived as risk factors because, according to

Marcus, people tend to forget how they felt in the past, assuming that their attitude has not changed and is thus identical to their current attitude (1986).

Another interesting research field is related to the elicitation of requirements. The formalized approach requires a holistic overview of the product following the expected utility perspective, which assumes that people analyze their overall level of wealth (Camerer, et al., 2003). Agile approaches assume the gradual discovery of requirements. These two approaches are similar to snack choosing experiments, where the moment of decision influenced the tendency to increase the variety of products (Simonson, 1990). The analysis regarding the dependence between the moment of decision making and its consequence may therefore be analyzed. According to Loewenstein et al., there exists an identified tendency to mispredict future tastes, which may be different from current ones (1995, 2003).

During the period when users are considering whether or not to accept the new product, they anticipate the effort of its adoption and their final level of satisfaction. This may be compared to people anticipating their happiness in case of severe disease or significant improvement of health (Loewenstein, et al., 2008). It seems that the average happiness level is similar in groups of patients and healthy subjects. However, patients anticipate significant improvement of happiness once their disease is over, while healthy people anticipate a significant decrease of happiness level in the case of losing their health. Both groups underestimate their adoption level, and this increases their fear and hope respectively. Facing a new product associated with the organization of their work, people may also underestimate their adoption level and misjudge the quality of the product.

Social relations also seem to play important role in the perception of software quality. In experiment 3 it was shown that subjects agreed with the opinion of others and adopted these opinions as their own. Experiments in the area of social rewards (Fehr, et al., 2000), the punishment of unjust attitudes (Charness, et al., 2002) (Sanfey, et al., 2003) (Aumann, 2006), social embarrassment (compare Wolosin, et al., 1975, Van Boven, et al., 2005) or moral choices (compare Greene, et al., 2004) may be the starting point for the development of a new approach to software project management. Simon's call for analysis in organizations (1979) needs to be heeded, and it may be defined as an important direction for further research - especially as most economic activities are performed by employees of organizations (1995). Decisions in organizations, according to Thompson's model, are often subject to political bargaining, negotiation etc. (1995). Camerer et. al point out that people often make friends and enemies in organizations (2007). Therefore, the decisions associated with the acceptance of a software product may be deeply connected with organizational decision theory. However,



the relationship between quality perception and the final acceptance decision requires further research.

An important research area is connected to group dynamics and self identification. People tend to identify with a group (Akerlof, et al., 2005 - feeling sympathy, for example, if they know that somebody else has something in common with them (for example, the same birthday date) (Miller, et al., 1998). This may influence evaluators' attitudes, and in consequence influence their decision regarding the product. It may be also interesting to see how quality is assessed after a longer period. This may encourage further co-operation between vendor and customer. There are empirical research results which suggest that people memorize the final state and the emotions that they had at the peak (compare Redelmeier, et al., 2003). On the other hand, Bargh et al. suggest that people tend to remember their attitudes toward people and objects even if they cannot memorize details (1996). In this context, the mechanism of building the long term attitudes of customer representatives seems an interesting issue to research (compare with the context of changing memories Markus, 1986).

The influence of experience on decision making is still poorly understood (Rakow, et al., 2010). Rakow and other researchers suggest that experts make their judgments based on intuition and experience (compare Klein, 2002). Each experience is coded in the declarative memory as a chunk containing context, choice and obtained outcome (Erev, et al., 2010). This idea follows David Hume's concept that from similar causes people expect similar outcomes (Gilboa, et al., 1995). Considering also the description-experience gap, it is remarkable that current state-of-the-art theory does not provide satisfying predictions of economics agents behavior (compare Erev, et al., 2010). However, according to Zukier and Pepitone, decision makers may be asked to behave rationally, and this may influence decision making processes (1984).

Software product vendors often base their approach on an escalation of commitment (Barry, 1976), sunk cost effect (Arkes, et al., 1985), or entrapment in investment (Rubin, et al., 1975). However, these effects have been shown to be reversed in certain situations (Garland, et al., 1998). Heath has proposed an explanation based on mental accounting (compare Thaler, 1985), arguing that if the decision maker has set budget limits they will dislike a situation where their limits might be exceeded (1995). In the modern approach to software acquisition processes, customers set their limits and often employ professional purchasers to negotiate with vendors. The research question may be then asked about the efficiency of current strategies based on the escalation of commitment and similar biases.

The list of other potential sources of influence on deliberative reasoning includes anchoring (compare Tversky, et al., 1974), time pressure factors (compare Finucane, et al., 2000), emotions activated in an unrelated manner (compare Johnson, et al., 1983), involvement in concurrent cognitive tasks (compare Shiv, et al., 1999), performing tasks during peak or off-peak times (compare Gonzalez, et al., 2004), mood (compare Han, et al., 2007), the manipulation of the status quo option (compare Samuelson, et al., 1988), embarrassment and the illusion of one's own and others' courage (compare Van Boven, et al., 2005), contrast with other products (compare Bettman, et al., 1998), disgust (compare Morales, et al., 2007) or the disruptive influence of Transcranial Magnetic Stimulation (compare Knoch, et al., 2006). Visceral factors such as intelligence (compare Stanovich, et al., 1999), the need for cognition (compare Shafir, et al., 2002), and the ability to think statistically (compare Agnoli, 1991) may also influence the result of the judgment process. More detailed analysis of these interactions is another important direction for further research.

## 11 LIST OF FIGURES

Figure 1-1 Organization of the dissertation (source: own study).....	18
Figure 2-1 Testing in the V model (ISTQB, 2008) .....	28
Figure 2-2 A typical waterfall lifecycle model (Rajlich, 2006) .....	32
Figure 2-3 A typical V model lifecycle (Boehm, 2006) .....	32
Figure 2-4 A modern spiral lifecycle model (ISO/IEC29119 CD, 2009) .....	33
Figure 2-5 Relations between process quality and product quality perspectives (ISO/IEC9126-1, 2001).....	37
Figure 2-6 Internal and external quality characteristics with sub-characteristics (ISO/IEC9126-1, 2001).....	38
Figure 2-7 SQuaRE model organization (ISO/IEC25000, 2005).....	39
Figure 2-8 Relation between software quality and system quality (ISO/IEC25010 FDIS, 2011) .....	40
Figure 2-9 Product's quality model (ISO/IEC25010 FDIS, 2011) .....	40
Figure 2-10 a) The software quality lifecycle; b) The decomposition of software quality characteristics (ISO/IEC25000, 2005) .....	41
Figure 2-11 Software product evaluation process overview from acquirer's perspective (ISO/IEC14598-4, 1999).....	45
Figure 2-12 Quality and responsibility layers for a web service (Abramowicz, et al., 2008)..	46
Figure 3-1 A typical gains and losses valuation function (Kahneman, et al., 1979).....	53
Figure 3-2 The system quality in use perspective (ISO/IEC25010 FDIS, 2011).....	58
Figure 3-3 Belief revision regarding software quality (Stavrinoudis, et al., 2005).....	59
Figure 3-4 Food quality perception model (Steenkamp, 1989) .....	61
Figure 4-1 Dual-system model (Kahneman, 2003).....	64
Figure 7-1 Normative-based model of software quality perception (source: own study).....	91
Figure 7-2 Normative-based model of software quality perception with perception filter (source: own study) .....	92
Figure 7-3 Normative-based model of software quality perception based on knowledge and mental state of the observer (source: own study).....	93
Figure 7-4 Theoretical descriptive model of software quality perception (source: own study) .....	94
Figure 8-1 Four independent groups layout for experiment 1 (source: own study).....	106
Figure 8-2 Phases of experiment 1 (source: own study) .....	107

Figure 8-3 Four independent groups layout for second experiment (source: own study).....	108
Figure 8-4 Phases of the second experiment (source: own study) .....	109
Figure 9-1 TestApp1 issue submission form in the Polish language (source: own study) ....	112
Figure 9-2 TestApp2 transfer submission form in the Polish language (source: own study)	114
Figure 9-3 TestApp3 transaction submission form in the Polish language (source: own study) .....	115
Figure 9-4 TestApp4 transaction submission form in the Polish language (source: own study) .....	116
Figure 9-5 Logical architecture of the “TestLab” platform (source: own study).....	124
Figure 9-6 An example of the use of a Likert-type scale in the “TestLab” framework (source: own study).....	125
Figure 9-7 An example of “Blue screen” failure (source: own study).....	127
Figure 9-8 Conceptual and operational levels of independent and dependent variables (source: own study).....	144

## 12 LIST OF TABLES

Table 1-1 Mapping of the hypothetico-deductive research model stages onto parts of this dissertation (source: own study).....	16
Table 5-1 Selected cognitive biases (source: own study) .....	75
Table 6-1 Summary of characteristics for 15 projects with a budget of over US\$ 250 million (source: own study) .....	82
Table 8-1 Software Quality Perception Model differences to normative model (source: own study).....	98
Table 8-2 Mapping of experiments onto research topics (source: own study) .....	101
Table 8-3 Variables hypothesized to impact on the software quality evaluation process (source: own study) .....	104
Table 8-4 Fault probability ( $f_p$ ) patterns A and B (source: own study).....	107
Table 9-1 Independent variables configuration for the research (source: own study) .....	121
Table 9-1 Faults implemented in “TestLab” (source: own study) .....	126
Table 9-3 Data gathered from the first surveys, the pre-test and the evaluation of versions *.0.1, *.0.2 and *.0.3 in type 1 experiment (source: own study) .....	132
Table 9-4 ANOVA table for homogeneity tests for $H_0: M_{A,*}=M_{B,*}$ (source: own study).....	133
Table 9-5 ANOVA table for homogeneity tests for $H_0: M_{*,L}=M_{*,H}$ (source: own study).....	133
Table 9-6 ANOVA table for reaction to quality level change test for $H_0: M_{A,*}=M_{B,*}$ (source: own study).....	134
Table 9-7 ANOVA table for reaction to quality level change test for $H_0: M_{*,L}=M_{*,H}$ (source: own study).....	134
Table 9-8 Data gathered during the evaluation of version *.0.5 in type 1 experiment (source: own study).....	135
Table 9-9 ANOVA table for main effect test for $H_0: M_{A,*}=M_{B,*}$ (source: own study) .....	135
Table 9-10 ANOVA table for main effect test for $H_0: M_{*,L}=M_{*,H}$ (source: own study) .....	136
Table 9-11 Simple estimators compared to managers’ opinions (source: own study).....	136
Table 9-12 empirical record of second experiment (source: own study).....	138
Table 9-13 homogeneity tests for second experiment (source: own study).....	138
Table 9-14 treatment effect in the second experiment (source: own study) .....	138
Table 9-15 empirical record of third experiment (source: own study) .....	140
Table 9-16 ANOVA table for homogeneity test for $H_0: M_{POS}=M_{NEG}$ (source: own study)...	140
Table 9-17 ANOVA table for main effect test for $H_0: M_{POS}=M_{NEG}$ (source: own study) .....	140

Table 9-17 Subjective number of failures reported (source: own study).....	141
Table A-1 Experiment 1 – Personal surveys raw data (source: own study) .....	200
Table A-2 Experiment 1 – Raw data of evaluation tasks (source: own study) .....	212
Table A-3 Experiment 2 – Personal surveys raw data (source: own study) .....	214
Table A-4 Experiment 2 – Raw data of evaluation task (source: own study).....	217
Table A-5 Experiment 3 – Personal surveys raw data (source: own study) .....	220
Table A-6 Experiment 3 – Raw data of evaluation task (source: own study).....	222

## 13 REFERENCES

- [1] Abramowicz W., Haniewicz K., Hofman R., Kaczmarek M., Zyskowski D., Decomposition of SQuaRE-Based Requirements For The Needs Of SOA Applications, Trends in Communication Technologies and Engineering Science / book auth. Ao Sio-long, Huang Xu and Wai Ping-kong Alexander (Eds.), Springer Netherlands, 2009, Vol. 33, ISBN 978-1-4020-9532-0.
- [2] Abramowicz W., Hofman R., Suryan W., Zyskowski D., SQuaRE based Web Services Quality Model, International Conference on Internet Computing and Web Services, Hong Kong: International Association of Engineers, 2008, ISBN: 978-988-98671-8-8.
- [3] Achinstein P., Barker S.F., The legacy of logical positivism: Studies in the philosophy of science, Johns Hopkins University Press, 1969.
- [4] Adair J.G., The Hawthorne effect: A reconsideration of the methodological artifact, Journal of Applied Psychology, 1984, 2: Vol. 69, pp. 334-345.
- [5] Agile International Agile Manifesto [Online], 2008, 2008, <http://www.agilemanifesto.org>.
- [6] Agnoli F., Development of judgmental heuristics and logical reasoning: Training counteracts the representativeness heuristic, Cognitive development, 1991, 2: Vol. 6, pp. 195-217.
- [7] Ainslie G., Derivation of "rational" economic behavior from hyperbolic discount curves, The American Economic Review, JSTOR, 1991, Vol. 81, pp. 334-340.
- [8] Ajzen I., Fishbein M., Understanding attitudes and predicting social behavior, Englewood Cliffs, NJ: Prentice-Hall, 1980.
- [9] Akerlof G.A., Kranton R.E., Identity and the Economics of Organizations, The Journal of Economic Perspectives, American Economic Association, 2005, 1: Vol. 19, pp. 9-32.
- [10] Akerlof G.A., Behavioral Macroeconomics and Macroeconomic Behavior, American Economist, Omicron Delta Epsilon, 2003, 1: Vol. 47, pp. 25-48.
- [11] Akerlof G.A., The Market for 'Lemons': Quality Uncertainty and the Market Mechanism, Quarterly Journal of Economics, 1970, 84.
- [12] Alagumalai S., Curtis D.D., Hungi N., Applied Rasch Measurement: A Book of Exemplars: Papers in Honour of John P. Keeves, Springer: The Netherlands, 2005.

- [13] Alchourron C., Gardenfors P., Makinson D., On, the Logic of Theory Change: Partial Meet Functions for Contraction and Revision, *Journal of Symbolic Logic*, 1985, Vol. 50.
- [14] Allais M., Le comportement de l'homme rationnel devant le risque: Critique des postulats et axiomes de l'école Américaine, *Econometrica: Journal of the Econometric Society*, JSTOR, 1953, 4: Vol. 21, pp. 503-546.
- [15] Anand P., *Foundations of Rational Choice Under Risk*, Oxford: Oxford University Press, 2002.
- [16] Anderson C., *Freeconomics, The Economist-The World in 2008*, 2007.
- [17] Anderson J., *Language, memory and thought*, Hillsdale, NY: Erlbaum, 1976.
- [18] Andrade E.B., Ariely D., Short and long-term consequences of emotions in decision making, University of California-Berkeley, forthcoming.
- [19] Andreasson A.R., Attitudes and customer behavior: A decision model, *New Research in Marketing*, Preston, 1965, pp. 1-16.
- [20] Andreoni J., Miller J., Giving according to GARP: An experimental test of the consistency of preferences for altruism, *Econometrica*, JSTOR, 2002, pp. 737-753.
- [21] Angner E., Loewenstein G.F., *Behavioral Economics, Philosophy of Economics*, Elsevier, Amsterdam, 2007, Vol. 13, Available: <http://ssrn.com/abstract=957148>.
- [22] Anscombe F.J., Aumann R.J., A definition of subjective probability, *Annals of mathematical statistics*, JSTOR, 1963, pp. 199-205.
- [23] APA Code of Ethics, American Psychological Association, 2003, 10th.
- [24] Apelbaum J., *Rapid Application Development Framework*, Technology Press, 2002.
- [25] Ariely D., Loewenstein G.F., Prelec D., Coherent Arbitrariness: Stable Demand Curves Without Stable Preferences, *Quarterly Journal of Economics*, MIT Press, 2003, 1: Vol. 118, pp. 73-105.
- [26] Arkes H.R., Blumer C., The psychology of sunk cost, *Organizational behavior and human decision processes*, Elsevier, 1985, 1: Vol. 35, pp. 124-140.
- [27] Aronson E., Wilson T.D., Akert R., *Social Psychology The Heart and The Mind* Harper Collins, New York, 1994.
- [28] Asch S., Effects of group pressure upon the modification and distortion of judgments, *Groups, leadership and men*, Guetzkow H. (ed), Pittsburgh: Carnegie, 1951.
- [29] Aumann R.J., Agreeing to disagree, *The annals of statistics*, JSTOR, 1976, 6: Vol. 4, pp. 1236-1239.



- [30] Aumann R.J., War, peace, Proceedings of the National Academy of Sciences, National Academy of Sciences, 2006, 46: Vol. 103, p. 17075.
- [31] Ayer A.J., Language, Truth and Logic, London, 1936.
- [32] Babulak E., Carrasco R., The IT Quality of Service Provision Analysis in Light of User's Perception and Expectations, International Symposium on CSNDSP, Staffordshire University, 2002.
- [33] Bargh J.A., Chaiken S., Raymond P., Hymes C., The automatic evaluation effect: Unconditional automatic attitude activation with a pronunciation task, Journal of Experimental Social Psychology, Elsevier, 1996, 1: Vol. 32, pp. 104-128.
- [34] Barnett S., Ceci S., The Role of Transferable Knowledge in Intelligence, Cognition and intelligence, Sternberg R. and Pretz J., Cambridge: Cambridge University Press, 2005.
- [35] Baron J., Thinking and Deciding, Cambridge: Cambridge University Press, 2000, third edition.
- [36] Baron R., So Right It's Wrong: Groupthink and the Ubiquitous Nature of Polarized Group Decision Making, Advances in experimental social psychology, Zanna M. (Ed.), San Diego: Elsevier Academic Press, 2005, Vol. 37.
- [37] Barron G., Erev I., Small feedback-based decisions and their limited correspondence to description-based decisions, Journal of Behavioral Decision Making, John Wiley & Sons, 2003, 3: Vol. 16, pp. 215-233.
- [38] Barros A.P., Dumas M., Bruza P.D., The move to web service ecosystems, BPTrends, 2005, 3: Vol. 3.
- [39] Barry S., Knee-deep in the Big Muddy: A Study of Escalating Commitment to a Chosen Course of Action, Organizational Behavior and Human Performance, 1976, 16: Vol. 1.
- [40] Basili V.R., The experimental paradigm in software engineering, Lecture Notes in Computer Software, Rombach D., Basili V.R. and Selby R., Springer-Verlag, 1993.
- [41] Basili V.R., The role of controlled experiments in software engineering research, Empirical Software Engineering Issues, Basili V.R., Berlin: Springer-Verlag, 2007.
- [42] Basu K., Deb R., Pattanaik P.K., Soft sets: An ordinal formulation of vagueness with some applications to the theory of choice, Fuzzy Sets and Systems, Elsevier, 1992, 1: Vol. 45, pp. 45-58.
- [43] Bazzana G., Anderson O., Jokela T., ISO9126 and ISO9000: friends or foes?, Software Engineering Standards Symposium, 1993.

- [44] Beck K., *Extreme programming eXplained: embrace change*, Addison-Wesley, 2000, ISBN: 0-201-61641-6.
- [45] Becker G., *Crime and punishment: An Economic Approach*, *Journal of Political Economy*, 1968.
- [46] Bell D.E., *Regret in decision making under uncertainty*, *Operations research*, JSTOR, 1982, pp. 961-981.
- [47] Benhabib J., Bisin A., *Modeling internal commitment mechanisms and self-control: A neuroeconomics approach to consumption-saving decisions*, *Games and Economic Behavior*, Elsevier, 2005, 2: Vol. 52, pp. 460-492.
- [48] Bernheim B.D., Rangel A., *Addiction and cue-triggered decision processes*, *The American Economic Review*, American Economic Association, 2004, 5: Vol. 94, pp. 1558-1590.
- [49] Bernoulli D., *Exposition of a new theory on the measurement of risk*, *Econometrica: Journal of the Econometric Society*, JSTOR, [1738] 1954, 1: Vol. 22, pp. 23-36.
- [50] Bettman J.R., *The decision maker who came in from the cold*, *Advances in consumer research*, 1993, 1: Vol. 20, p. 7.
- [51] Bettman J.R., Luce M.F., Payne J.W., *Constructive consumer choice processes*, *Journal of consumer research*, University of Chicago Press, 1998, 3: Vol. 25, pp. 187-217.
- [52] Björk S., Holopainen J., Ljungstrand P., Mandryk R., *Special issue on ubiquitous games*, *Personal and Ubiquitous Computing*, Springer-Verlag, 2002, 5-6: Vol. 6, pp. 358-361.
- [53] Blokdijk G., *SaaS 100 Success Secrets-How companies successfully buy, manage, host and deliver software as a service (SaaS)*, Emereo Pty Ltd, 2008.
- [54] Blount S., *When Social Outcomes Aren't Fair: The Effect of Causal Attributions on Preferences*, *Organizational behavior and human decision processes*, Elsevier, 1995, 2: Vol. 63, pp. 131-144.
- [55] Bobkowska A., *Prognozowanie jakości oprogramowania na podstawie modeli UML*, Gdańsk: Politechnika Gdańska, Rozprawa Doktorska, 2001.
- [56] Boehm B., Brown J., Lipow M., MacCleod G., *Characteristics of software quality*, New York: American Elsevier, 1978.
- [57] Boehm B., Bose P., *A collaborative spiral software process model based on theory W*, 1994, pp. 59-68.

- [58] Boehm B., Some future trends and implications for systems and software engineering processes, *Systems Engineering*, John Wiley & Sons, 2006, 1: Vol. 9, pp. 1-19.
- [59] Boring E., *A History of Experimental Psychology*, Prentice-Hall, 1950, Second Edition.
- [60] Bornstein R., Exposure and affect: overview and meta-analysis of research, 1968-1987, *Psychological Bulletin*, 1989, 106.
- [61] Braddon-Mitchel D., Jackson F., *Philosophy of Mind and Cognition*, Oxford: Blackwell, 1996.
- [62] Brandstätter E., Comparison based satisfaction: contrast and empathy, *European Journal of Social Psychology*, John Wiley & Sons, 2000, 5: Vol. 30, pp. 673-703.
- [63] Bransford J., Barclay J., Franks J., Sentence memory: constructive versus interpretive approach, *Cognitive Psychology*, 1972, 3.
- [64] Brocas I., Carrillo J.D., The brain as a hierarchical organization, *The American Economic Review*, American Economic Association, 2008, 4: Vol. 98, pp. 1312-1346.
- [65] Brody T.A., de la, Peña L., Hodgson P.E., *The philosophy behind physics*, Springer Verlag, 1993.
- [66] Brooks F.P., *No Silver Bullet*, 1986.
- [67] Brooks F.P., *The mythical man-month*, Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 1995, (anniversary ed.).
- [68] Brookshire D., Coursey D., Measuring the Value of a Public Good: An Empirical Comparison of Elicitation Procedures, *American Economic Review*, 1987, 4: Vol. 77.
- [69] Burke S.J., The dimensional effects of missing information on choice processing, *Journal of Behavioral Decision Making*, John Wiley & Sons, 1995, 4: Vol. 8, pp. 223-244.
- [70] Burks A.W., Peirce's theory of abduction, *Philosophy of Science*, JSTOR, 1946, 4: Vol. 13, pp. 301-306.
- [71] Camerer C., Loewenstein G.F., *Behavioral Economics: Past, Present, Future* (introduction for *Advances in Behavioral Economics*), Mimeo: Carnegie Mellon University, 2003.
- [72] Camerer C., Weber M., Recent developments in modeling preferences: Uncertainty and ambiguity, *Journal of risk and uncertainty*, Springer, 1992, 4: Vol. 5, pp. 325-370.
- [73] Camerer C., Loewenstein G.F., Prelec D., Neuroeconomics: How Neuroscience Can Inform Economics, *Journal of Economic Literature*, 2005, Vol. XLIII.

- [74] Camerer C.F., Malmendier U., Behavioral organizational economics, Behavioral Economics and Its Applications. Princeton University Press, Princeton, NJ, 2007.
- [75] Camerer C.F., Prospect theory in the wild: Evidence from the field, Choices, Values and Frames, Kahneman D. and Tversky A., Cambridge: Cambridge University Press, 2001.
- [76] Cannon-Bowers J.A., Salas E., Making decisions under stress: Implications for individual and team training, American Psychological Association Washington, DC, 2000.
- [77] Carmon Z., Ariely D., Focusing on the Forgone: How Value Can Appear So Different to Buyers and Sellers, Journal of Consumer Research, 2000.
- [78] Carver J., Voorhis J.V. and Basili V.R. Understanding the impact of assumptions on experimental validity, Proceedings of the 2004 International Symposium on Empirical Software Engineering, The Institute of Electrical and Electronic Engineers, Inc., 2004.
- [79] Casadesus-Masanell R., Ghemawat P., Dynamic mixed duopoly: A model motivated by Linux vs. Windows, Management Science, Citeseer, 2006, 7: Vol. 52, p. 1072.
- [80] Case K., Fair R., Principles of Economics, Prentice-Hall, 1999, 5 ed, ISBN: 0-13-961905-4.
- [81] Charness G., Rabin M., Understanding Social Preferences with Simple Tests, Quarterly journal of Economics, MIT Press, 2002, 3: Vol. 117, pp. 817-869.
- [82] Cherian J., Harris B., Capricious Consumption and the Social Brain Theory: Why Consumers Seem Purposive even in the Absence of Purpose, Advances in Consumer Research, 1990, Vol. 17, pp. 518-523.
- [83] CNN Money Money CNN com [Online], CNN, 2010, 2010, <http://monet.cnn.com>.
- [84] Coase R.H., The problem of social cost, The journal of Law and Economics, University of Chicago Press, 1960, Vol. 3.
- [85] Cohen J.D., Statistical power analysis for the behavioral sciences, Hillsdale, NJ: Erlbaum, 1988, Second Edition.
- [86] Cook T.D., Campbell D.T., Quasi-experimentation: Design & analysis issues for field settings, Houghton Mifflin Boston, 1979.
- [87] Côté, M.A., Suryn W. and Georgiadou E. Software Quality Model Requirements for Software Quality Engineering, Software Quality Management & INSPIRE Conference (BSI), 2006.

- [88] Cox A., Granbois D.H., Summers J., Planning, search, certainty and satisfaction among durable buyers: a longitudinal study, *Advances in Consumer Research*, 1983, Vol. 10, pp. 39-499.
- [89] Curley S.P., Yates J.F., Abrams R.A., Psychological sources of ambiguity avoidance, *Organizational behavior and human decision processes*, Elsevier, 1986, 2: Vol. 38, pp. 230-256.
- [90] Cusumano M.A., *The changing software business: Moving from products to services*, Computer-IEEE Computer Society, Institute Of Electrical And Electronics, 2008, 1: Vol. 41, p. 20.
- [91] Datamonitor, *Software: Global Industry Guide*, Report Code: DO-4959, Datamonitor, 2009, [http://www.infoedge.com/product\\_type.asp?product=DO-4959](http://www.infoedge.com/product_type.asp?product=DO-4959).
- [92] Davies A., Brady T., Hobday M., Organizing for solutions: systems seller vs. systems integrator, *Industrial Marketing Management*, Elsevier, 2007, 2: Vol. 36, pp. 183-193.
- [93] Davis F., Perceived usefulness, perceived ease of use, and user acceptance of information technology, *MIS Quarterly*, 1989, 13: Vol. 3.
- [94] Davis M., *Great Software Debates*, Wiley-IEEE Computer Society Press, 2004.
- [95] de Araujo I.E., Rolls E.T., Velazco M.I., Margot C., Cayeux I., Cognitive modulation of olfactory processing, *Neuron*, Elsevier, 2005, 4: Vol. 46, pp. 671-679.
- [96] De Dreu C.K.W., Beersma B., Stroebe K., Euwema M.C., Motivated information processing, strategic choice, and the quality of negotiated agreement, *Journal of Personality and Social Psychology*, APA American Psychological Association, 2006, 6: Vol. 90, p. 927.
- [97] De Martino B., Kumaran D., Seymour B., Dolan R.J., Frames, biases, and rational decision-making in the human brain, *Science*, AAAS, 2006, 5787: Vol. 313, p. 684.
- [98] Dewey J., *How We Think*, [1910] 1978.
- [99] Diao Y., Bhattacharya K., Estimating business value of IT services through process complexity analysis, 2008, pp. 208-215.
- [100] Dijkstra E., The Humble Programmer (EWD340), *Communications of the ACM*, 1972.
- [101] Doyle J., Thomason R.H., Background to qualitative decision theory, *AI magazine*, 1999, 2: Vol. 20, p. 55.
- [102] Drake S., *Galileo at work: his scientific biography*, Dover Pubns, 2003.

- [103] Dromey R., A Model For Software Product Quality [Online], Australian Software Quality Research Institute, 1994, [http://www.sqi.gu.edu.au/docs/sqi/technical/Model\\_For\\_S\\_W\\_Prod\\_Qual.pdf](http://www.sqi.gu.edu.au/docs/sqi/technical/Model_For_S_W_Prod_Qual.pdf).
- [104] Duckworth K.L., Bargh J.A., Garcia M., Chaiken S., The automatic evaluation of novel stimuli, *Psychological Science*, John Wiley & Sons, 2002, 6: Vol. 13, pp. 513-519.
- [105] Duesenberry J.S., *Income, Saving and the Theory of Consumer Behavior*, Harvard University press, 1952.
- [106] Dymek D., *Zarządzanie jakością oprogramowania komputerowego*, Kraków: Akademia Ekonomiczna w Krakowie, Rozprawa Doktorska, 2000.
- [107] Edwards W., Experiments on economic decision-making in gambling situations, *Econometrica*, 1953, 2: Vol. 21, pp. 349-350.
- [108] Egan A., Some counterexamples to causal decision theory, *The Philosophical Review*, Duke University Press, 2007, 1: Vol. 116, p. 93.
- [109] Ellsberg D., Risk ambiguity, and the Savage axioms, *The Quarterly Journal of Economics*, JSTOR, 1961, pp. 643-669.
- [110] Elster J., Emotions and economic theory, *Journal of economic literature*, JSTOR, 1998, 1: Vol. 36, pp. 47-74.
- [111] eMarketer E-Commerce market size in Europe, eMarketer, 2006.
- [112] Engel J.F., Blackwell R.D., Kollat D.T., *Consumer Behavior*, Chicago: Dryden, 1982, 4th Edition.
- [113] Engel J.F., Blackwell R.D., Miniard P.W., *Consumer Behavior*, New York: The Dryden Press, 1995, 8th Edition.
- [114] Engel J.F., Kollat D.T., Blackwell R.D., *Consumer Behavior*, New York: Holt, Rinehart and Winston Inc., 1968.
- [115] Erasmus A.C., Boshoff E., Rousseau G.G., Consumer decision-making models within the discipline of consumer science: a critical approach, *Journal of Family Ecology and Consumer Sciences*, 2010, 0: Vol. 29.
- [116] Erev I., Ert E., Roth A.E., Haruvy E., Herzog S.M., Hau R., Hertwig R., Stewart T., West R.F., Lebiere C., A choice prediction competition: Choices from experience and from description, *Journal of Behavioral Decision Making*, John Wiley & Sons, 2010, 1: Vol. 23, pp. 15-47.

- [117] Ernst N., Mylopoulos J., On, the perception of software quality requirements during the project lifecycle, *Requirements Engineering: Foundation for Software Quality*, Springer, 2010, pp. 143-157.
- [118] Evans J., Dual-processing accounts of reasoning, judgment, and social cognition, *Annual Review of Psychology*, 2008, Vol. 59, pp. 255-278.
- [119] Fehr E., Gächter S., Fairness and retaliation: The economics of reciprocity, *The Journal of Economic Perspectives*, JSTOR, 2000, 3: Vol. 14, pp. 159-181.
- [120] Fenton N., Pfleeger S., *Software Metrics. A Rigorous and Practical Approach*, Boston: PWS Publishing Company, 1997, second edition.
- [121] Festinger L., *A theory of cognitive dissonance*, Stanford, CA: Stanford University Press, 1957.
- [122] Festinger L., *A theory of social comparison processes*, Human relations, Tavistock Inst, 1954, 2: Vol. 7, p. 117.
- [123] Feyerabend P.K., *Against method*, London: New Left Books, 1975.
- [124] Fienberg S., Tanur J., Reconsidering the fundamental contributions of Fisher and Neyman on experimentation, *International Statistical Review*, 1996, 64.
- [125] Finucane M.L., Alhakami A., Slovic P., Johnson S.M., The affect heuristic in judgments of risks and benefits, *Journal of Behavioral Decision Making*, John Wiley & Sons, 2000, 1: Vol. 13, pp. 1-17.
- [126] Firat A.F., A critique of the orientations in theory development in consumer behavior: Suggestions for the future, *Advances in Consumer Research*, 1985, 1: Vol. 12, pp. 3-6.
- [127] Fischhoff B., Judgment and decision making, *The psychology of human thought*, Sternberg R. and Smith, E. (eds), New York: Cambridge University Press, 1988.
- [128] Fisher R., *The Design of Experiments*, Edinburgh: Oliver and Boyd, 1935.
- [129] Frederick S., Loewenstein G.F., O'Donoghue T., Time discounting and time preference: A critical review, *Journal of Economic Literature*, American Economic Association, 2002, 2: Vol. 40, pp. 351-401.
- [130] Freud S., Bonaparte M., Fließ W., *The origins of psycho-analysis*, Basic Books New York, 1954.
- [131] Friedman M., *Essays in positive economics*, University of Chicago Press, 1953.
- [132] Friedman M., *The methodology of positive economics*, *The Philosophy of economics: an anthology*, 1953, Vol. 2, pp. 180-213.
- [133] Fudenberg D., Levine D.K., A dual-self model of impulse control, *The American Economic Review*, JSTOR, 2006, 5: Vol. 96, pp. 1449-1476.

- [134] Garland H., Conlon D.E., Too Close to Quit: The Role of Project Completion in Maintaining Commitment, *Journal of Applied Social Psychology*, John Wiley & Sons, 1998, 22: Vol. 28, pp. 2025-2048.
- [135] Gartner Research Top End User Predictions for 2010, Gartner Inc., 2010.
- [136] Gibbard A., Harper W., Counterfactuals and Two Kinds of Expected Utility, *Foundations and applications of decision theory*, [1978] 1981, Vol. 1, pp. 125-162.
- [137] Gilbert D.T., Pinel E., Wilson T.D., Blumberg S., Wheatley T., Immune neglect: A source of durability bias in affective forecasting, *Journal of Personality and Social Psychology*, 1998, 75.
- [138] Gilboa I., Schmeidler D., Case-based decision theory, *The Quarterly Journal of Economics*, JSTOR, 1995, 3: Vol. 110, pp. 605-639.
- [139] Glimcher P.W., Kable J., Louie K., Neuroeconomic Studies of Impulsivity: Now or Just as Soon as Possible?, *The American economic review*, JSTOR, 2007, 2: Vol. 97, pp. 142-147.
- [140] Godfrey-Smith P., *Theory and reality*, University of Chicago Press, 2003.
- [141] Gonzalez R., Loewenstein G.F., Effects of circadian rhythm on cooperation in an experimental game, Working Paper, Cornege Mellon University, 2004.
- [142] Google Annual Financial Report, 2010, <http://investor.google.com/financial/tables.html>.
- [143] Gossen H., *Die Entwicklung der Gesetze des menschlichen Verkehrs und der daraus fließenden Regeln für menschliches Handel (The Development of the Laws of Human Intercourse and the Consequent Rules of Human Action)*, 1854.
- [144] Greene J.D., Nystrom L.E., Engell A.D., Darley J.M., Cohen J.D., The neural bases of cognitive conflict and control in moral judgment, *Neuron*, Elsevier, 2004, 2: Vol. 44, pp. 389-400.
- [145] Grether D.M., Plott C.R., Economic theory of choice and the preference reversal phenomenon, *The American Economic Review*, JSTOR, 1979, 4: Vol. 69, pp. 623-638.
- [146] Grönroos C., A service quality model and its marketing implications, *European Journal of Marketing*, 1984, 4: Vol. 18.
- [147] Grossman S.J., Stiglitz J.E., Information and competitive price systems, *The American Economic Review*, JSTOR, 1976, 2: Vol. 66, pp. 246-253.
- [148] Grove A., Two modellings for theory change, *Journal of Philosophical Logic*, 1988, Vol. 17.



- [149] Gu X., Nahrstedt K., Chang R., Ward C., QoS-Assured Service Composition in managed Service Overlay Networks, 23rd International Conference on Distributed Computing Systems (ICDCS), Providence, RI, 2003.
- [150] Gul F., A theory of disappointment aversion, *Econometrica: Journal of the Econometric Society*, JSTOR, 1991, 3: Vol. 59, pp. 667-686.
- [151] Güth W., Schmittberger R., Schwarze B., An, experimental analysis of ultimatum bargaining, *Journal of Economic Behavior and Organization*, 1982, 3.
- [152] Haden J., Körner S., *Kants Leben und Lehre (Kant's Life and Thought)* / trans. Cassirer E., Yale University Press, 1981.
- [153] Haisley E., Mostafa R., Loewenstein G.F., Subjective relative income and lottery ticket purchases, *Journal of Behavioral Decision Making*, John Wiley & Sons, 2008, 3: Vol. 21, pp. 283-295.
- [154] Hale J., Householder B., Greene K., *The theory of reasoned action*, *The persuasion handbook: Developments in theory and practice*, Dillard J. and Pfau M., Thousand Oaks, CA: Sage, 2003.
- [155] Hamrol A., Mantura W., *Zarządzanie jakością: teoria i praktyka*, Warszawa: Wydawnictwo Naukowe PWN, 2006, third edition.
- [156] Han S., Lerner J.S., Keltner D., Feelings and consumer decision making: The appraisal-tendency framework, *Journal of Consumer Psychology*, Citeseer, 2007, 3: Vol. 17, p. 158.
- [157] Hansen F., *Consumer choice behavior: A cognitive theory*, Free Press, 1972.
- [158] Hansson S., *Decision Theory: A Brief Introduction* [Online], Department of Philosophy and the History of Technology, Royal Institute of Technology in Stockholm, 2005, 07 28, 2009, <http://www.infra.kth.se/~soh/decisiontheory.pdf>.
- [159] Harding S.G., *Can, theories be refuted?: essays on the Duhem-Quine thesis*, D. Reidel, 1976.
- [160] Harper D.A., *How, entrepreneurs learn: a Popperian approach and its limitations*, Economics Department New York University, 1999.
- [161] Harrison G.W., List J.A., Field experiments, *Journal of Economic Literature*, American Economic Association, 2004, 4: Vol. 42, pp. 1009-1055.
- [162] Hastie R., Dawes R., *Rational choice in an uncertain world*, Sage Publications, 2001.
- [163] Hau R., Pleskac T.J., Kiefer J., Hertwig R., The description-experience gap in risky choice: The role of sample size and experienced probabilities, *Journal of Behavioral Decision Making*, John Wiley & Sons, 2008, 5: Vol. 21, pp. 493-518.

- [164] Hau R., Pleskac T.J., Hertwig R., Decisions from experience and statistical probabilities: Why they trigger different choices than a priori probabilities, *Journal of Behavioral Decision Making*, John Wiley & Sons, 2010, 1: Vol. 23, pp. 48-68.
- [165] Hausman D.M., Economic methodology in a nutshell, *The Journal of Economic Perspectives*, American Economics Association, 1989, 2: Vol. 3, pp. 115-127.
- [166] Heath C. Escalation and de-escalation of commitment in response to sunk costs: The role of budgeting in mental accounting., *Organizational Behavior and Human Decision Processes*, 1995, 1: Vol. 62, pp. 38-54.
- [167] Hemenover S.H., Zhang S., Anger, personality, and optimistic stress appraisals, *Cognition & Emotion*, Psychology Press, 2004, 3: Vol. 18, pp. 363-382.
- [168] Hempel C.G., Oppenheim P., *Studies in the Logic of Explanation*, Philosophy of Science, JSTOR, 1948, 2: Vol. 15, pp. 135-175.
- [169] Henslin J., *Sociology: a down to earth approach*, Pearson Education, 2008, ninth edition, ISBN 978-0-205-57023-2.
- [170] Hertwig R., Barron G., Weber E.U., Erev I., Decisions from experience and the effect of rare events in risky choice, *Psychological Science*, Sage Publications, 2004, 8: Vol. 15, p. 534.
- [171] Hirschheim R., *Information systems epistemology: An historical perspective*, Research methods in information systems, Elsevier, 1985, pp. 13-35.
- [172] Hochstein L., Nakamura T., Shull F., Zazworka N., Voelp M., Zerkowicz M.V., Basili V.R., *An Environment for Conducting Families of Software Engineering Experiments*, Advances in Computers, Elsevier, 2008, Vol. 74, pp. 175-200.
- [173] Hoeffler S., Ariely D., West P., Path Dependent Preferences: The Role of Early Experience and Biased Search in Preference Development, *Organizational Behavior and Human Decision Processes*, 2006, 2: Vol. 101.
- [174] Hofman R., Behavioral economics in software quality engineering, *Empirical Software Engineering*, Springer, April 2011, 2: Vol. 16, pp. 278-293, ISSN 1382-3256.
- [175] Hofman R., Modele jakości oprogramowania - historia i perspektywy, *Zwinność i dyscyplina w inżynierii oprogramowania*, J. Nawrocki B. Walter, Poznań: NAKOM, 2007, ISBN: 978-83-89529-45-9.
- [176] Hofman R., Software quality perception, *Innovations and Advanced Techniques in Systems*, Computing Sciences and Software Engineering, CISSE2008, Springer 2009.

- [177] Holzman M., Theories of choice and conflict in psychology and economics, The Journal of conflict resolution, The Journal of Conflict Resolution, 1958, 4: Vol. 2, pp. 310-320, issn: 0022-0027.
- [178] Horowitz J.K., A test of intertemporal consistency, Journal of Economic Behavior & Organization, Elsevier, 1992, 1: Vol. 17, pp. 171-182.
- [179] Howard J.A., Sheth J.N., The theory of buyer behavior, Wiley New York, 1969.
- [180] Howard J.A., Marketing management: Analysis and planning, RD Irwin, 1963.
- [181] Hsee C.K., Loewenstein G.F., Blount S., Bazerman M.H., Preference reversals between joint and separate evaluations of options: A review and theoretical analysis, Psychological Bulletin, 1999, 5: Vol. 125, pp. 576-590.
- [182] Hutchison T.W., Professor Machlup on verification in economics, Southern Economic Journal, JSTOR, 1956, 4: Vol. 22, pp. 476-483.
- [183] Hwang C.L., Yoon K., Multiple attribute decision making: methods and applications: a state-of-the-art survey, Springer Verlag, 1981.
- [184] IEEE 1061 Standard for a Software Quality Metrics Methodolog, New York: Institute of Electrical and Electronics Engineers, 1998.
- [185] IEEE SwEBok Software Engineering Body of Knowledge, Institute of Electrical and Electronics Engineers, <http://www.swebok.org/>, 2004.
- [186] ISO/IEC TR 19759 Software Engineering Body of Knowledge, Geneve: International Standardization Organization, 2006.
- [187] ISO/IEC12207 FDIS Systems and software engineering - Software life cycle, Geneve: International Standardization Organization, 2007.
- [188] ISO/IEC14598 Information technology - Software product evaluation, Geneve: International Standardization Organization, 1999.
- [189] ISO/IEC14598-4 Software engineering - Product evaluation, Part 4: Process for acquirers, Geneve: International Standardization Organization, 1999.
- [190] ISO/IEC14598-5 Information technology - Software product evvaluation, Part 5: Process for evaluators, Geneve: International Standardization Organization, 1998.
- [191] ISO/IEC25000 Software Engineering - Software product Quality Requirements and Evaluation (SQuaRE), Geneve: International Standardization Organization, 2005.
- [192] ISO/IEC25010 FDIS Software engineering-Software product Quality Requirements and Evaluation (SQuaRE) Quality model, Geneve: International Standardization Organization, 2011.

- [193] ISO/IEC29119 CD Software Testing Standard, Geneva: International Standardization Organization, 2009.
- [194] ISO/IEC9126-1 Software engineering - Product quality - Part 1: Quality model, Geneva: ISO, 2001.
- [195] ISO/IEC9241-11 Ergonomic requirements for office work with visual display terminals (VDTs) - Part 11: Guidance on usability, Geneva: International Standardization Organization, 1998.
- [196] ISO9001 Quality management systems - Requirements, Geneva: International Standardization Organization, 2008.
- [197] ISTQB Certified Tester Foundation Level Syllabus [Online], International Software Testing Qualifications Board, 2008, <http://www.istqb.org/downloads/syllabi/SyllabusFoundation.pdf>.
- [198] IT, Service Management Forum An Introductory Overview of ITIL® V3, The UK Chapter of the itSMF, 2007, ISBN 0-9551245-8-1.
- [199] ITGI COBIT 4.1, ISACA, Rolling Meadows, IL (USA), IT Governance Institute, 2007.
- [200] Iverson W., Real World Web Services, O'Reilly, 2004.
- [201] Jaeger M., Rojec-Goldmann G., SENECA-simulation of algorithms for the selection of web services for compositions, Technologies for E-Services, Springer, 2006, pp. 84-97.
- [202] Jaskiewicz A., Inżynieria Oprogramowania, Warszawa: Helion, 1997.
- [203] Jevons W.S. The Theory of Political Economy, London, 1871.
- [204] Johnson E.J., Tversky A., Affect, generalization, and the perception of risk, Journal of Personality and Social Psychology, 1983, 1: Vol. 45, pp. 20-31.
- [205] Joyce J.M., The foundations of causal decision theory, Cambridge Univ Press, 1999.
- [206] Judd C.M., Kenny D.A., Estimating the effects of social interventions, Cambridge Univ Press, 1981.
- [207] Juran J.M., How, to, think about Quality, Juran's quality handbook, 1998.
- [208] Kahneman D., A perspective on judgment and choice: Mapping bounded rationality, American Psychologist, 2003, Vol. 58.
- [209] Kahneman D., Frederick S., Frames and brains: elicitation and control of response tendencies, Trends in Cognitive Sciences, Elsevier, 2007, 2: Vol. 11, pp. 45-46.

- [210] Kahneman D., Frederick S., Representativeness revisited: Attribute substitution in intuitive judgment, *Heuristics and biases: The psychology of intuitive judgment*, 2002, pp. 49-81.
- [211] Kahneman D., Tversky A., "Prospect" theory: an analysis of decision under risk, *Econometrica*, 1979, 47.
- [212] Kahneman D., Tversky A., Subjective probability: A judgment of representativeness, *Cognitive psychology*, Elsevier, 1972, 3: Vol. 3, pp. 430-454.
- [213] Kahneman D., Knetsch J., Thaler R.H., Experimental Test of the endowment effect and the Coase Theorem, *Journal of Political Economy*, 1990, 98: Vol. 6.
- [214] Kalepu S., Krishnaswamy S., Loke, S.W., Reputation = f(User Ranking, Compliance, Verity), *Proceedings of the IEEE International Conference on Web Services*, San Diego, CA, 2004.
- [215] Kan S.H., *Metrics and models in software quality engineering*, Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, 2002.
- [216] Kan S.H., Basili V.R., Shapiro L.N., Software quality: an overview from the perspective of total quality management, *IBM Systems Journal*, IBM, 2010, 1: Vol. 33, pp. 4-19.
- [217] Kang G., James J., Alexandris K., Measurement of internal service quality: application of the Servqual battery to internal service quality, *Managing Service Quality*, 2002, 5: Vol. 12.
- [218] Kano N., Seraku N., Takahashi F., Tsuji S., Attractive quality and must-be quality, *The Journal of the Japanese Society for Quality Control*, 1984, 2: Vol. 14, pp. 39-48.
- [219] Kant I., *Critique of Pure Reason*, Macmillan, London, 1932.
- [220] Kassarijian H.H., The development of consumer behavior theory, *Advances in Consumer Research*, 1982, 1: Vol. 9, pp. 20-22.
- [221] Keeney R.L., A utility function for examining policy affecting salmon on the Skeena River, *Journal of the Fisheries Research Board of Canada*, 1977, 1: Vol. 34, pp. 49-63.
- [222] Keeney R.L., Raiffa H., *Decisions with multiple objectives: Preferences and value tradeoffs*, Cambridge Univ Press, 1993.
- [223] Kermer D.A., Driver-Linn E., Wilson T.D., Gilbert D.T., Loss aversion is an affective forecasting error, *Psychological Science*, Sage Publications, 2006, 8: Vol. 17, p. 649.
- [224] Keynes J.M., *The general theory*, London, New York, 1936.
- [225] Kiliński A., *Jakość*, Warszawa: Wydawnictwo Naukowo Techniczne, 1979.

- [226] Kitchenham B., Pfleeger S., Software Quality: The Elusive Target, IEEE Software, 1996, 13 (1).
- [227] Klein G.A., Intuition at work, Intuition at work, Currency, 2002.
- [228] Knight F.H., Risk, uncertainty and profit, Boston and New York, 1921.
- [229] Knoch D., Pascual-Leone A., Meyer K., Treyer V., Fehr E., Diminishing reciprocal fairness by disrupting the right prefrontal cortex, Science, AAAS, 2006, 5800: Vol. 314, p. 829.
- [230] Knutson B., Wimmer G.E., Rick S., Hollon N.G., Prelec D., Loewenstein G.F., Neural antecedents of the endowment effect, Neuron, Elsevier, 2008, 5: Vol. 58, pp. 814-822.
- [231] Kobyliński A., Modele jakości produktów i procesów programowych, Warszawa: Oficyna Wydawnicza Szkoły Głównej Handlowej, 2005.
- [232] Kokash N., Dandrea V., Evaluating Quality of Web Services - A Risk-Driven Approach, Technical Report # DIT-06-099, 2006.
- [233] Koopmans T.C., Stationary ordinal utility and impatience, Econometrica: Journal of the Econometric Society, JSTOR, 1960, 2: Vol. 28, pp. 287-309.
- [234] Köszegi B., Rabin M., A Model of Reference-Dependent Preferences, The Quarterly Journal of Economics, MIT Press, 2006, 4: Vol. 121, pp. 1133-1165.
- [235] Kramer A., Twigg B., Quality control in the food industry, Avi, 1983, 3rd edition.
- [236] Krawczyk-Bryłka B. Psychologiczne aspekty jakości oprogramowania, Informatyka, 2000, 12.
- [237] Krzanik L., Enactable models for quantitative evolutionary software processes, Proceedings of the Forth International Software Process Workshop (ISPW '88), Tully C. (ed.), Moretonhampstead, Devon, UK: IEEE Computer Society, 1988.
- [238] Kuhn T.S., Logic of discovery of psychology of research, Criticism and the Growth of Knowledge, Lakatos I. and Musgrave A. (eds.), Cambridge: Cambridge University Press, 1970.
- [239] Kuhn T.S., The function of measurement in modern physical science, Isis, JSTOR, 1961, 2: Vol. 52, pp. 161-193.
- [240] Kuhn T.S., The structure of scientific revolutions, University of Chicago press, 1996.
- [241] Kunda Z., Nisbett R.E., Predicting Individual Evaluations from Group Evaluations and Vice Versa, Personality and Social Psychology Bulletin, 1988, 14: Vol. 2, pp. 326-334.
- [242] Kvist P., The Swedish Automobile Market and Quality Uncertainty, Department of Economics, School of Economics and Management, Lund University, 2004.

- [243] Laibson D.I., Golden Eggs and Hyperbolic Discounting, *Quarterly Journal of Economics*, MIT Press, 1997, 2: Vol. 112, pp. 443-477.
- [244] Lakatos I., *Falsification and the methodology of scientific research programmes*, Cambridge University Press, Cambridge, 1970, pp. 91-196.
- [245] Lakatos I., *Proofs and refutations, The logic of mathematical discovery*, Worrall J. and Zahar E., Cambridge: Cambridge Univ Press, 1976.
- [246] Lakatos I., The role of crucial experiments in science, *Philosophy of Science*, 1974, 4: Vol. 4.
- [247] Lakatos I., Feyerabend P.K., Motterlini M., *For and against method*, University of Chicago Press, 1999.
- [248] Lakatos I., Worrall J., Currie G., *The methodology of scientific research programmes*, Cambridge Univ Press, 1980.
- [249] Lambert C., The marketplace of perceptions, *Harvard Magazine*, Harvard Magazine Inc., 2006, 4: Vol. 108, p. 50.
- [250] Latsis S.J., Situational determinism in economics, *British Journal for the Philosophy of Science*, JSTOR, 1972, 3: Vol. 23, pp. 207-245.
- [251] Lehman M., Ramil J., Wernick P., Perry D., Turski W., *Metrics and Laws of Software Evolution - The Nineties View*, Fourth International Software Metrics Symposium (METRICS'97), 1997.
- [252] Leibenstein H., Allocative Efficiency vs. "X-efficiency", *The American Economic Review*, JSTOR, 1966, 3: Vol. 56, pp. 392-415.
- [253] Lerner J.S., Keltner D., *Beyond valence: Toward a model of emotion-specific influences on judgement and choice*, *Cognition & Emotion*, Psychology Press, 2000, 4: Vol. 14, pp. 473-493.
- [254] Lerner J.S., Small D.A., Loewenstein G.F., *Heart Strings and Purse Strings*, *Psychological Science*, SAGE Publications, 2004, 5: Vol. 15, p. 337.
- [255] Levine G., Parkinson S., *Experimental Methods in Psychology*, Hillsdale, NJ: Lawrence Erlbaum, 1994.
- [256] Levitt S., List J.A., *Field Experiments in Economics: The Past, The Present, and The Future*, Cambridge: National Bureau of Economic Research Working Paper Series, 2008.
- [257] Lewin S.B., Economics and psychology: Lessons for our own day from the early twentieth century, *Journal of Economic Literature*, JSTOR, 1996, 3: Vol. 34, pp. 1293-1323.

- [258] Lewisohn D., Mill and Comte on the methods of social science, *Journal of the History of Ideas*, JSTOR, 1972, 2: Vol. 33, pp. 315-324.
- [259] Lind E.A., Kray L., Thompson L., The social construction of injustice: Fairness judgments in response to own and others' unfair treatment by authorities, *Organizational Behavior and Human Decision Processes*, ACADEMIC PRESS INC, 1998, Vol. 75, pp. 1-22.
- [260] Lindstrom L., Jeffries R., Extreme programming and agile software development methodologies, *Information Systems Management*, Taylor & Francis Group Ltd, 2 Park Square Oxford OX 14 4 RN UK, 2004, 3: Vol. 21, pp. 41-52.
- [261] List J.A., Neoclassical Theory Versus Prospect Theory: Evidence From The Marketplace, *Econometrica*, 2004, 72.
- [262] Liu Y., Ngu A.H.H., Zeng L., QoS Computation and Policing in Dynamic Web Service Selection, ACM Press, New York, 2004.
- [263] Loewenstein G.F., Adler D., A bias in the prediction of tastes, *The Economic Journal*, JSTOR, 1995, pp. 929-937.
- [264] Loewenstein G.F., Lerner J.S., The role of affect in decision making, *Handbook of affective science*, Davidson R., Scherer K. and Goldsmith H. (eds), Oxford University Press, 2003.
- [265] Loewenstein G.F., O'Donoghue T., Animal Spirits: Affective and deliberative influences on economic behavior, Social Science Research Network, 2004.
- [266] Loewenstein G.F., Prelec D., Anomalies in intertemporal choice: Evidence and an interpretation, *The Quarterly Journal of Economics*, JSTOR, 1992, 2: Vol. 107, pp. 573-597.
- [267] Loewenstein G.F., Ubel P.A., Hedonic adaptation and the role of decision and experience utility in public policy, *Journal of Public Economics*, Elsevier, 2008, 8-9: Vol. 92, pp. 1795-1810.
- [268] Loewenstein G.F., Anticipation and the valuation of delayed consumption, *The Economic Journal*, JSTOR, 1987, 387: Vol. 97, pp. 666-684.
- [269] Loewenstein G.F., *The Fall and Rise of Psychological Explanations*, Russell Sage Foundation Publications, 1992.
- [270] Loewenstein G.F., O'Donoghue T., Rabin M., Projection Bias In Predicting Future Utility, *Quarterly Journal of Economics*, MIT Press, 2003, 4: Vol. 118, pp. 1209-1248.
- [271] Loewenstein G.F., Rick S., Cohen J.D., *Neuroeconomics*, Annual Reviews, 2007.



- [272] Loomes G., Sugden R., Disappointment and dynamic consistency in choice under uncertainty, *The Review of Economic Studies*, JSTOR, 1986, pp. 271-282.
- [273] Lutz M.A., Lux K., Boulding K.E., *The challenge of humanistic economics*, Benjamin Cummings Pub. Co., 1979.
- [274] Maccheroni F., Marinacci M., Rustichini A., *Social decision theory: Choosing within and between groups*, Carlo Alberto Notebooks, Collegio Carlo Alberto, 2008.
- [275] Maciaszek L., *Architecture-Centric Software Quality Management*, *Web Information Systems and Technologies*, WEBIST 2008, LNBIP, Cordeiro J., Hammoudi S. and Filipe J. (eds.), Springer-Verlag Berlin Heidelberg (to appear), 2009.
- [276] Markin R.J., *The supermarket: an analysis of growth, development, and change*, Washington State University Press, 1968.
- [277] Markowitz H., The utility of wealth, *The Journal of Political Economy*, JSTOR, 1952, 2: Vol. 60, pp. 151-158.
- [278] Markus G.B., Stability and change in political attitudes: Observed, recalled, and "explained", *Political Behavior*, Springer Netherlands, 1986, Vol. 8, pp. 21-44, 10.1007/BF00987591.
- [279] Marshall J.D., Knetsch J.L., Sinden J.A., Agents' evaluations and the disparity in measures of economic loss, *Journal of Economic Behavior & Organization*, Elsevier, 1986, 2: Vol. 7, pp. 115-127.
- [280] Mather M., Johnson M., Choice-supportive source monitoring: Do our decisions seem better to us as we age?, *Psychology and Aging*, 2000, 15.
- [281] Maximilien E., Singh M., *Reputation and Endorsement for Web Services*, ACM SIGecom Exchanges, 2001.
- [282] McCall J., Richards P., Walters G., *Factors In software quality*, Griffiths Air Force Base, NY, Rome Air Development Center Air Force Systems Command, 1977.
- [283] McClure S.M., Li J., Tomlin D., Cypert K.S., Montague L.M., Montague P.R., Neural correlates of behavioral preference for culturally familiar drinks, *Neuron*, Elsevier, 2004, 2: Vol. 44, pp. 379-387.
- [284] McClure S.M., Laibson D.I., Loewenstein G.F., Cohen J.D., Separate neural systems value immediate and delayed monetary rewards, *Science*, AAAS, 2004, 5695: Vol. 306, p. 503.
- [285] McConnell S., *Code Complete*, Microsoft Press, 2004, second edition, ISBN 0-7356-1967-0.

- [286] Medvec V.H., Madely S.F., Gilovich T., When less is more: Counterfactual thinking and satisfaction among Olympic medalists, *Journal of Personality and Social Psychology*, 1995, Vol. 69 (4), pp. 603-610.
- [287] Mellers B.A., Schwartz A., Ho K., Ritov I., Decision affect theory: Emotional responses to risky options, *Psychological Science*, 1997, Vol. 8, pp. 423-429.
- [288] Mellers B.A., McGraw A.P., Anticipated emotions as guides to choice, *Current Directions in Psychological Science*, Sage Publications, 2001, 6: Vol. 10, p. 210.
- [289] Mellers B.A., Ritov I., How, beliefs influence the relative magnitude of pleasure and pain, *Journal of Behavioral Decision Making*, John Wiley & Sons, 2009.
- [290] Menasce D., QoS-aware software components, *IEEE Internet Computing*, 2004, 2: Vol. 8.
- [291] Merchant C., *The death of nature*, Organization and Environment, Sage Inc., 1980, 2: Vol. 11.
- [292] Microsoft Annual Revenue Report, Microsoft, 2010, [http://www.microsoft.com/msft/reports/ar09/10k\\_fh\\_fin.html](http://www.microsoft.com/msft/reports/ar09/10k_fh_fin.html).
- [293] Miguel P., daSilva M., Chiosini E., Schützer K., Assessment of service quality dimensions: a study in a vehicle repair service chain, POMS College of Service Operations and EurOMA Conference New Challenges in Service Operations, London, 2007.
- [294] Mill J.S., *System of logic*, Classworks, 1986.
- [295] Miller D.T., Downs J.S., Prentice D.A., Minimal conditions for the creation of a unit relationship: The social bond between birthdaymates, *European Journal of Social Psychology*, John Wiley & Sons, 1998, 3: Vol. 28, pp. 475-481.
- [296] Mintzberg H., Raisinghani D., Théorêt A., The Structure of 'Unstructured' Decision Processes, *Administrative Sciences Quarterly*, 1976, 21.
- [297] Montague P.R., Dayan P., Sejnowski T.J., A framework for mesencephalic dopamine systems based on predictive Hebbian learning, *Journal of Neuroscience*, Soc Neuroscience, 1996, 5: Vol. 16, p. 1936.
- [298] Montes L., Smith and Newton: some methodological issues concerning general economic equilibrium theory, *Cambridge Journal of Economics*, CPES, 2003, 5: Vol. 27, p. 723.
- [299] Mook D., In defense of external invalidity, *American Psychologist*, 1983, 38.

- [300] Morales A.C., Fitzsimons G.J., Product Contagion: Changing Consumer Evaluations Through Physical Contact with "Disgusting" Products, *Journal of Marketing Research*, American Marketing Association, 2007, 2: Vol. 44, pp. 272-283.
- [301] Murphy G., Medin D., The role of theories in conceptual coherence, *Psychological Review*, 1985, 92.
- [302] Nagle D.B., *The household as the foundation of Aristotle's polis*, Cambridge University Press, 2006.
- [303] Nelson R., Tracking Propaganda to the Source: Tools for Analyzing Media Bias, *Global Media Journal*, 2003, 3: Vol. 2, available: <http://lass.calumet.purdue.edu/cca/gmj/fa03/gmj-fa03-nelson.htm>.
- [304] Nęcka E., Orzechowski J., Szymura B., *Psychologia poznawcza*, Warszawa: Wydawnictwo Naukowe PWN, 2008.
- [305] Nicosia F.M., *Consumer decision processes: marketing and advertising implications*, Prentice Hall, 1966.
- [306] Nielsen J., Usability 101: introduction to usability [Online], 2003, <http://www.useit.com/alertbox/20030825-.html>.
- [307] Nijland G.O., The tetrahedron of knowledge acquisition: a meta-model of the relations among observation, conceptualization, evaluation and action in the research on socio-ecological systems, *Systems Research and Behavioral Science*, John Wiley & Sons, 2002, 3: Vol. 19, pp. 211-221.
- [308] Nisbett R.E., Krantz D.H., Jepson C., Kunda Z., The use of statistical heuristics in everyday inductive reasoning, *Psychological Review*, Elsevier, 1983, 4: Vol. 90, pp. 339-363.
- [309] Nobel Foundation Nobelprize.org [Online], All Laureates in Economics, 07 27, 2009, [http://nobelprize.org/nobel\\_prizes/economics/laureates/](http://nobelprize.org/nobel_prizes/economics/laureates/).
- [310] Nozick R., Newcomb's problem and two principles of choice, *Essays in honor of Carl G. Hempel*, 1969, pp. 107-33.
- [311] OASIS Web Services Quality Model (WSQM) TC [Online], 2005, <http://www.oasis-open.org/committees/wsqm/charter.php>.
- [312] Oehler A., Reisch L., *Behavioral Economics--eine neue Grundlage für die Verbraucherpolitik*, Eine Studie im Auftrag des vzbv eV Bamberg, Berlin, Kopenhagen: Verbraucherzentrale Bundesverband eV, 2008.
- [313] Olshavsky R.W., Granbois D.H., Consumer decision making-fact or fiction?, *Journal of Consumer Research*, JSTOR, 1979, 2: Vol. 6, pp. 93-100.

- [314] Oprel L., Kwaliteit in breder perspectief, Proefstation voor de Bloemisterij Aalsmeer, 1989.
- [315] Osgood C., Suci G., Tannenbaum P., The measurement of meaning, Urbana, IL: University of Illinois Press, 1957.
- [316] Ozanne J.L., Keyword Recognition: A New Methodology for the Study of Information Seeking Behavior, *Advances in Consumer Research*, 1988, Vol. 15.
- [317] Paivio A., Caspo K., Picture superiority in free recall: imagery or dual coding?, *Cognitive psychology*, 1973, 5.
- [318] Pande P.S., Holpp L., What is six sigma?, McGraw-Hill Professional, 2002.
- [319] Papaioannou I., Tsesmetzis D., Roussaki I., Anagnostou M., A QoS Ontology Language for Web-Services, 20th International Conference on Advanced Information Networking and Applications - Volume 1 (AINA'06), 2006.
- [320] Papazoglou M., Yang J., Design methodology for web services and business processes, *Technologies for E-Services*, Springer, 2002, pp. 175-233.
- [321] Parasuraman A., Zeithaml V., Berry L., A conceptual model of services quality and its implication for future research, *Journal of Marketing*, 1985, 4: Vol. 49.
- [322] Parco J.E., Rapoport A., Stein W.E., Effects of financial incentives on the breakdown of mutual trust, *Psychological Science*, SAGE Publications, 2002, 3: Vol. 13, p. 292.
- [323] Patton R., *Software Testing*, Sams Publishing, 2005, second edition.
- [324] Paul R., *Mises and Austrian economics: A personal view*, The Ludwig von Mises Institute of Auburn University, 1984.
- [325] Paulus M.P., Stein M.B., An insular view of anxiety, *Biological Psychiatry*, Elsevier, 2006, 4: Vol. 60, pp. 383-387.
- [326] Peart S.J., Levy D.M., *Post-Ricardian british economics, 1830-1870*, Blackwell Companion to the History of Economic Thought; Warren S.; Biddle, J.; Davis, J., Malden, MA: Wiley Online Library, 2003, pp. 130-44.
- [327] Peirce C.S., A neglected argument for the reality of God, *The Hibbert Journal*, 1908, 1: Vol. 7, pp. 90-112.
- [328] Peirce C.S., Houser N., Kloesel C.J.W., *The Essential Peirce: Selected Philosophical Writings, 1893-1913*, Indiana Univ Press, 1998.
- [329] Peppas P., Williams M., Constructive modellings for theory change, *Notre Dame Journal of Formal Logic*, 1995, 1: Vol. 36.
- [330] Pfleeger S., *Software Engineering: Theory and Practice*, Prentice Hall, 2001, second edition.

- [331] Pigosky T., *Practical Software Maintenance*, New York: Wiley & Sons, 1996.
- [332] Polič M. *Decision Making: Between Rationality and Reality*, *Interdisciplinary Description of Complex Systems*, 2009, 2: Vol. 7, pp. 78-89.
- [333] Popper K.R., *Conjectures and refutations: The growth of scientific knowledge*, Psychology Press, 2002.
- [334] Popper K.R., *The logic of scientific discovery*, Hutchinson, London, 1959.
- [335] Pressman R., *Software Engineering. A Practitioner's Approach*, McGraw Hill, 1992, 3rd edition, ISBN: 0-07-050814-3.
- [336] Pressman R., *Software Engineering: A practitioner's approach*, Boston, 2001, fifth edition.
- [337] Preston J., Munévar G., Lamb D., Ebooks Corporation, *The worst enemy of science?: essays in memory of Paul Feyerabend*, Oxford University Press, 2000.
- [338] Quiggin J., *A theory of anticipated utility*, *Journal of Economic Behavior & Organization*, Elsevier, 1982, 4: Vol. 3, pp. 323-343.
- [339] Raaij W.F., *Attribution of causality to economic actions and events*, *Kyklos*, John Wiley & Sons, 1985, 1: Vol. 38, pp. 3-19.
- [340] Rajlich V., *Changing the paradigm of software engineering*, *Communications of the ACM*, ACM, 2006, 8: Vol. 49, p. 70.
- [341] Rakow T., Newell B.R., *Degrees of uncertainty: An overview and framework for future research on experience-based choice*, *Journal of Behavioral Decision Making*, John Wiley & Sons, 2010, 1: Vol. 23, pp. 1-14.
- [342] Ran Sh., *A Model for Web Services Discovery With QoS*, *ACM SIGecom Exchanges*, 2003, 1: Vol. 4.
- [343] Rangel A., Camerer C., Montague P.R., *A framework for studying the neurobiology of value-based decision making*, *Nature Reviews Neuroscience*, Nature Publishing Group, 2008, 7: Vol. 9, pp. 545-556.
- [344] Rasmussen D., *Ayn Rand on Obligation and Value*, Libertarian Alliance, 1990, ISBN: 1-85637-120-4.
- [345] Read D., Orsel B., Rahman J., Frederick S., *Four score and seven years from now: the "date/delay effect" in temporal discounting*, *Management Science*, Department of Operational Research, London School of Economics and Political Science, 2005, Vol. 51 (9), pp. 1326-1335.

- [346] Read D., Loewenstein G.F., Kalyanaraman S., Mixing virtue and vice: Combining the immediacy effect and the diversification heuristic, *Journal of Behavioral Decision Making*, John Wiley & Sons, 1999, 4: Vol. 12, pp. 257-273.
- [347] Redelmeier D.A., Katz J., Kahneman D., Memories of colonoscopy: A randomized trial, *Pain*, Elsevier, 2003, 1-2: Vol. 104, pp. 187-194.
- [348] Reder L., Ross B., Integrated knowledge in different tasks: The role of retrieval strategy on fan effects, *Journal of Experimental Psychology: Learning, Memory and Cognition*, 1983, 9.
- [349] Reed S., Pattern recognition and categorization, *Cognitive Psychology*, 1972, 3.
- [350] Rick S., Loewenstein G.F., The Role of Emotions in Economic Behavior, *Handbook of emotions*, Lewis M., Haviland-Jones J.M. and Barrett L.F., The Guilford Press, 2008.
- [351] Robbins L. *An Essay on the Nature and Significance of Economic Science*, London: Macmillan & Co., 1932.
- [352] Roger S.P., *Software Engineering-a practitioner's approach*, Boston: McGraw-Hill Companies Inc, 2001, 5th Edition (European Adaptatio).
- [353] Rosch E., Simpson C., Miller R., Structural bases for typical effects, *Journal of Experimental Psychology, Human Perception and Performance*, 1976, 2.
- [354] Ross D., Robbins, positivism and the demarcation of economics from psychology, *Lionel Robbins's essay on the nature and significance of economics science, 75th anniversary conference proceedings*, Cowell F. and Witztum A., London School of Economics and Political Science, 2007.
- [355] Roth A.E., Murnighan J.K., Schoumaker F., The deadline effect in bargaining: Some experimental evidence, *The American Economic Review*, JSTOR, 1988, 4: Vol. 78, pp. 806-823.
- [356] Rubel M., *Marx Without Myth: A Chronological Study of his Life and Work*, Blackwell, 1975, ISBN: 0-631-15780-8.
- [357] Rubin J.Z., Brockner J., Factors affecting entrapment in waiting situations: The Rosencrantz and Guildenstern effect, *Journal of Personality and Social Psychology*, 1975, 6: Vol. 31, pp. 1054-1063.
- [358] Sääksvuori A., Immonen A., *Product lifecycle management*, Springer Verlag, 2008.
- [359] Sagan A. *Modele Zachowań Konsumenta*. [Online], CEM - Instytut Badań Rynku i Opinii Publicznej, 2004, 08 16, 2010, <http://www.cem.pl/?a=pages&id=42>.
- [360] Sahakian W.S., *History and systems of psychology*, Schenkman Pub. Co., 1975.

- [361] Samuelson P.A., A note on measurement of utility, *The Review of Economic Studies*, JSTOR, 1937, pp. 155-161.
- [362] Samuelson W., Zeckhauser R., Status quo bias in decision making, *Journal of risk and uncertainty*, Springer, 1988, 1: Vol. 1, pp. 7-59.
- [363] Sanfey A.G., Rilling J.K., Aronson J.A., Nystrom L.E., Cohen J.D., The neural basis of economic decision-making in the ultimatum game, *Science*, AAAS, 2003, 5626: Vol. 300, p. 1755.
- [364] Sanfey A.G., Social decision-making: insights from game theory and neuroscience, *Science*, AAAS, 2007, 5850: Vol. 318, p. 598.
- [365] SAP Annual Financial Report, 2010, [http://www.wikinvest.com/stock/SAP\\_AG\\_\(SAP\)/Data/Revenue/2009](http://www.wikinvest.com/stock/SAP_AG_(SAP)/Data/Revenue/2009).
- [366] SASO Internal research results [Report], Poznań: Polish Software Attestation and Standardization Organization, 2010.
- [367] Sauer C., Jeffery D.R., Land L., Yetton P., The effectiveness of software development technical reviews: a behaviorally motivated program of research, *IEEE Transactions on Software Engineering*, 2000, 1: Vol. 26, pp. 1-14.
- [368] Savage L.J., *The Foundations of Statistics*, New York: Wiley, 1954.
- [369] Savage L.J., *The Foundations of Statistics*, Dover Pubns, 1972, Dover Edition.
- [370] Sawyer S., A market-based perspective on information systems development, *Communications of the ACM*, ACM, 2001, 11: Vol. 44, p. 102.
- [371] Schach S., *Practical Software Engineering*, Boston: IRWIN and Aksen Associats, 1992.
- [372] Schiffman L.G., Kanuk L.L., *Consumer Behavior*, Prentice Hall, Inc, 2000, 7th Edition.
- [373] Schmalensee R., Antitrust Issues in Schumpeterian Industries, *American Economics Review*, 2000, 2: Vol. 90.
- [374] Schoemaker P., The Expected Utility Model: Its Variants, Purposes, Evidence and Limitations, *Journal of Economic Literature*, 1982, 20.
- [375] Schumpeter J., *Capitalism, socialism, and democracy*, New York: Harper, 1950, 3rd ed..
- [376] Sears D.O., College sophomores in the laboratory: Influences of a narrow data base on social psychology's view of human nature., *Journal of Personality and Social Psychology*, 1986, 3: Vol. 51, pp. 515-530.

- [377] Shafir E., LeBoeuf R.A., Rationality, *Annual Review of Psychology*, Annual Reviews, Inc., 2002, pp. 491-518.
- [378] Shaughnessy J., Zechmeister E., Zechmeister J., *Research Methods in Psychology*, McGraw-Hill, 2005, Seventh edition.
- [379] Sheth A., Cardoso J., Miller J., Kochut K., Kang M., Qos for Service-oriented middleware, 6th World Multiconference on Systemics, Cybernetics and Informatics, Orlando, Florida, 2002.
- [380] Shiv B., Fedorikhin A., Heart and mind in conflict: The interplay of affect and cognition in consumer decision making, *Journal of Consumer Research*, UChicago Press, 1999, 3: Vol. 26, pp. 278-292.
- [381] Shogren S., Shin S., Hayes D., Kliebenstein J., Resolving Differences in Willingness to Pay and Willingness to Accept, *The American Economic Review*, 1994, 1: Vol. 84.
- [382] Sijtsema S., Your health!?! Transforming health perception into food product characteristics in consumer-oriented product design (WAU no. 3359), Wageningen University Dissertation, 2003.
- [383] Simon H.A., Hayes J.R., The understanding process: Problem isomorphs, *Cognitive Psychology*, 1976, 2: Vol. 8, pp. 165-190.
- [384] Simon H.A., From substantive to procedural rationality, *Methodological Appraisal in Economics*, Latsis S.J. (ed), Cambridge Univ Press, 1976.
- [385] Simon H.A., Making management decisions: The role of intuition and emotion, *The Academy of Management Executive (1987-1989)*, JSTOR, 1987, 1: Vol. 1, pp. 57-64.
- [386] Simon H.A., Organizations and markets, *Journal of public administration research and theory*, PMRA, 1995, 3: Vol. 5, p. 273.
- [387] Simon H.A., Rational choice and structure of environments, *Psychological review*, 1956, 63.
- [388] Simon H.A., Rational decision making in business organizations, *The American economic review*, JSTOR, 1979, 4: Vol. 69, pp. 493-513.
- [389] Simon H.A., *The New Science of Management Decision*, 1960.
- [390] Simonson I., The Effect of Purchase Quantity and Timing on Variety-Seeking Behavior, *Journal of Marketing Research*, 1990, 27.
- [391] Sjøberg, D.I.K.; Anda, B.; Arisholm, E.; Dybå, T.; Jørgensen, M.; Karahasanovic, A.; Koren, E.F.; Vokác, M., Conducting realistic experiments in software engineering, *First International Symposium on Empirical Software Engineering*, Nara, Japan: ISESE2002, 2002.



- [392] Sjøberg D.I.K., Dybå T., Jorgensen M., The future of empirical methods in software engineering research, IEEE Computer Society, 2007.
- [393] Smith A., An Inquiry into the Nature and Causes of the Wealth of Nations, 1776, Chapter IV: Of the Origin and Use of Money.
- [394] Smith A., The theory of moral sentiments, London: A. Millar, 1759.
- [395] Solomon M.R., Consumer behavior, Pearson Education, 2006.
- [396] Stangor C., Research Methods for the Behavioral Sciences, Boston, MA: Houghton Mifflin Company, 2007, third edition.
- [397] Stanovich K.E., West R.F., Discrepancies between normative and descriptive models of decision making and the understanding/acceptance principle, Cognitive Psychology, 1999, 3: Vol. 38, pp. 349-385.
- [398] Starmer C., Developments in non-expected utility theory: The hunt for a descriptive theory of choice under risk, Advances in behavioral economics, Princeton University Press, 2004, pp. 105-147.
- [399] Stavrinoudis D., Xenos M., Peppas P., Christodoulakis D., Early Estimation of Users' Perception of Software Quality, Software Quality Journal, 2005, Vol. 13.
- [400] Steenkamp J.B.E.M., Product quality: an investigation into the concept and how it is perceived by consumers (WAU no. 1253), Wageningen University Dissertation, 1989.
- [401] Steenkamp J.B.E.M., Wierenga B., Meulenberg M., Kwali-teits-perceptie van voedingsmiddelen deel 1. Swoka, Den Haag, 1986.
- [402] Sternberg R., Critical Thinking in Psychology: It really is critical, Critical Thinking in Psychology, Sternberg R., Roediger H. and Halpern D., Cambridge University Press, 2007, ISBN 0521608341.
- [403] Steuer M.D., The scientific study of society, Springer, 2003.
- [404] Stevens S., Mathematics, measurement and psychophysics, Handbook of experimental psychology, (Ed.) Stevens S., New York: Wiley, 1951.
- [405] Stevens S., Measurement, Psychophysics and Utility, Definitions and Theories, Churchman C. and P. Ratoosh, New York: Wiley, 1959.
- [406] Stigler G.J., Becker G.S., De gustibus non est disputandum, The American Economic Review, JSTOR, 1977, 2: Vol. 67, pp. 76-90.
- [407] Stigler S.M., The history of statistics: The measurement of uncertainty before 1900, Belknap Press, 1986.
- [408] Strack F., Schwarz N., Chassein B., Kern D., Wagner D., The salience of comparison standards and the activation of social norms: consequences for judgments of happiness

- and their communication, *British journal of social psychology*, 1990, 4: Vol. 29, pp. 303-314.
- [409] Strotz R.H., Myopia and inconsistency in dynamic utility maximization, *The Review of Economic Studies*, JSTOR, 1955, pp. 165-180.
- [410] Stroud B., *Hume*, London & New York: Routledge, 1977.
- [411] Student, On, Testing varieties of cereals, *Biometrika*, 1923, 15.
- [412] Suryan W., Abran A., ISO/IEC SQuaRE. The second generation of standards for software product quality., *IASTED2003*, 2003.
- [413] Szajna B., Software evaluation and choice: predictive evaluation of the Technology Acceptance Instrument, *MIS Quarterly*, 1994, 18: Vol. 3.
- [414] Szenberg M., Ramrattan L., *New, frontiers in economics*, Cambridge University Press, 2004.
- [415] Tavani H.T., The Classic Debate On The Relationship Between Faith And Reason: Some Contemporary Challenges From The Perspectives Of Relativism And Postmodernism, *Rivier Academic Journal*, 2008, Vol. 4 (1).
- [416] Thagard P.R., The best explanation: Criteria for theory choice, *The Journal of Philosophy*, JSTOR, 1978, 2: Vol. 75, pp. 76-92.
- [417] Thagard P.R., Why cognitive science needs philosophy and vice versa, *Topics in Cognitive Science*, Wiley Online Library, 2009, 2: Vol. 1, pp. 237-254.
- [418] Thaler R.H., Shefrin H.M., An economic theory of self-control, *The Journal of Political Economy*, JSTOR, 1981, pp. 392-406.
- [419] Thaler R.H., Mental accounting and consumer choice, *Marketing science*, JSTOR, 1985, 3: Vol. 4, pp. 199-214.
- [420] Thaler R.H., Toward a positive theory of consumer choice, *Journal of Economic Behavior & Organization*, Elsevier, 1980, 1: Vol. 1, pp. 39-60.
- [421] Thompson L., The impact of minimum goals and aspirations on judgments of success in negotiations, *Group decision and negotiation*, Springer, 1995, 6: Vol. 4, pp. 513-524.
- [422] Tian M., Gramm A., Ritter H., Schiller J., Efficient Selection and Monitoring of QoS-aware Web services with the WS-QoS Framework, *IEEE/WIC/ACM International Conference on Web Intelligence (WI 2004)*, Beijing, China, 2004.
- [423] Tom S.M., Fox C.R., Trepel C., Poldrack R.A., The neural basis of loss aversion in decision-making under risk, *Science*, AAAS, 2007, 5811: Vol. 315, p. 515.

- [424] Triesman A., Davies A., Divided attention to ear and eye, Attention and performance, Kronblum S. (ed.), New York: Academic Press, 1973, Vol. IV.
- [425] Turner M., Budgen D., Brereton P., Turning software into a service, Computer, Citeseer, 2003, 10: Vol. 36, pp. 38-44.
- [426] Tversky A., Kahneman D., Judgement under uncertainty: heuristics and biases, Judgement under uncertainty: heuristics and biases, Kahneman D., Tversky A. and Slovic P., Cambridge: Cambridge University Press, 1982.
- [427] Tversky A., Kahneman D., Judgment under Uncertainty: Heuristics and Biases, Science, 1974, 185.
- [428] Department of Defence Verification, Validation and Accreditation Recommended Practice Guide, Office of the Director of Defence Research and Engineering Defence Modelling and Simulation, 1996.
- [429] Ubel P.A., Loewenstein G.F., Schwarz N., Smith D., Misimagining the unimaginable: the disability paradox and health care decision making, Health Psychology, American Psychological Association, 2005, 4: Vol. 24, pp. S57-S62.
- [430] Underwood B., Shaughnessy J., Experimentation in psychology, New York: Wiley, 1975.
- [431] Van, Boven L., Loewenstein G.F., Dunning D., Mispredicting the endowment effect: Underestimation of owners' selling prices by buyer's agents, Journal of Economic Behavior & Organization, Elsevier, 2003, 3: Vol. 51, pp. 351-365.
- [432] Van, Boven L., Loewenstein G.F., Dunning D., The illusion of courage in social predictions: Underestimating the impact of fear of embarrassment on other people, Organizational Behavior and Human Decision Processes, Elsevier, 2005, 2: Vol. 96, pp. 130-141.
- [433] van Fraassen B.C., The scientific image, Oxford University Press, 1980.
- [434] Veblen T.B., The place of science in modern civilisation, New York BW Heubsch, New York BW Heubsch, 1919.
- [435] Veblen T.B., The theory of the leisure class: an economic study in the evolution of institutions, Macmillan, New York, 1899.
- [436] Vickers B., Francis Bacon and the progress of knowledge, Journal of the History of Ideas, JSTOR, 1992, 3: Vol. 53, pp. 495-518.
- [437] Vitvar T., Mocan A., Kerrigan M., Zaremba M., Zaremba M., Moran M., Cimpian E., Haselwanter T., Fensel D., Semantically-enabled service oriented architecture:

- concepts, technology and application, *Service Oriented Computing and Applications*, Springer, 2007, 2: Vol. 1, pp. 129-154.
- [438] Von Neumann J., Morgenstern O., *Theory of Games and Economic Behavior*, Princeton, NJ: Princeton University Press, 1944.
- [439] Vu L.H., Hauswirth M., Aberer K., QoS-based service selection and ranking with trust and reputation management, *On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE*, Springer, 2005, pp. 466-483.
- [440] Wallenius J., Dyer J.S., Fishburn P.C., Steuer R.E., Zionts S., Deb K., Multiple criteria decision making, multiattribute utility theory: Recent accomplishments and what lies ahead, *Management Science*, Citeseer, 2008, 7: Vol. 54, pp. 1336-1349.
- [441] Wang Y., Singh M., Formal Trust Model for Multiagent Systems, *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI)*, 2007.
- [442] Weber E.U., Shafir S., Blais A.R., Predicting Risk Sensitivity in Humans and Lower Animals: Risk as Variance or Coefficient of Variation., *Psychological review*, 2004, 2: Vol. 111, pp. 430-445.
- [443] Wetherick N., A further study of the inferential basis of concept attainment, *British Journal of Psychology*, 1968, 59.
- [444] Whewell W., *History of inductive sciences*, 1858.
- [445] Wiederhold G., What is your software worth?, *Communications of the ACM*, ACM, 2006, 9: Vol. 49, p. 75.
- [446] Wieser F., *Der natürliche Werth (Natural Value)*, 1889.
- [447] Wilkie W.L., Pessemier E.A., Issues in marketing's use of multi-attribute attitude models, *Journal of Marketing Research*, JSTOR, 1973, pp. 428-441.
- [448] Wolosin R.J., Sherman S.J., Cann A., Predictions of own and other's conformity, *Journal of Personality*, John Wiley & Sons, 1975, 3: Vol. 43, pp. 357-378.
- [449] Wylie G., Allport D., Task switching and the measurement of 'switch costs', *Psychological Research*, 2000, 63.
- [450] Xenos M., Christodoulakis D., Software quality: The user's point of view, *Software Quality and Productivity: Theory, practice, education and training*, Lee M., Barta B. and Juliff P., Chapman and Hall Publications, 1995, ISBN: 0-412-629607.
- [451] Zeiler K., Plott C.R., The Willingness to Pay/Willingness to Accept Gap, the Endowment Effect, Subject Misconceptions and Experimental Procedures for Eliciting Valuations., *American Economic Review*, 2004.

- [452] Zelenski J., *The Role of Personality in Emotion, Judgment and Decision Making, Do Emotions Help or Hurt Decision Making?*, Vohs K., Baumeister R. and Loewenstein G.F. (eds.), Russell Sage Foundation, 2007, ISBN: 978-0-87154-877-1.
- [453] Zeng L., Benatallah B., Dumas M., Kalagnanam J., Sheng Q.Z., *Quality driven web services composition*, ACM, 2003, p. 421.
- [454] Zukier H., Pepitone A., *Social roles and strategies in prediction: Some determinants of the use of base-rate information*, *Journal of Personality and Social Psychology*, Elsevier, 1984, 2: Vol. 47, pp. 349-360.
- [455] Zweig J., *Your money and your brain. How the new Science of Neuroeconomics Can Make You Rich*, Simon & Schuster, 2007.

---

BEHAVIORAL  
PRODUCTS QUALITY ASSESSMENT MODEL  
ON THE SOFTWARE MARKET

---

APPENDIX A – RAW EMPIRICAL DATA

PROMOTER: PROF. DR HAB. WITOLD ABRAMOWICZ

POZNAŃ, 2011

# 1. METHOD EVALUATION AND EXPERIMENT 1

Subject	Group	Task	A.1: Age	A.2: Working experience in years	A.3: Software tester experience in years	A.4: ISTQB Testing Foundation certificate	A.5: Have you ever evaluated banking related software?	A.6: Are you Internet banking user?	A.7: For how many years?	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-A.L-1	H,S:A.C	PersonalSurvey - A.L	47	22	5	0	0	1	6	10	6	11	11	11	6	9	6	11	11	11
Subject-A.L-2	H,S:A.C	PersonalSurvey - A.L	40	16	4	1	1	1	5	11	3	11	11	9	6	8	3	9	11	11
Subject-A.L-3	H,S:A.C	PersonalSurvey - A.L	31	7	6	1	1	1	6	11	8	11	8	7	7	6	3	11	11	11
Subject-A.L-4	H,S:A.C	PersonalSurvey - A.L	22	4	1	0	1	1	5	11	9	11	9	9	8	8	8	11	11	11
Subject-A.L-M	H	PersonalSurvey - A.L	40	21	19	1	1	1	13	10	11	11	9	9	7	11	7	11	11	11
Subject-A.H-1	H,S:A.A	PersonalSurvey - A.H	29	6	5	1	1	1	5	9	6	11	9	9	10	8	6	11	11	11
Subject-A.H-2	H,S:A.A	PersonalSurvey - A.H	34	7	7	1	1	1	6	10	8	7	10	10	8	10	9	10	11	10
Subject-A.H-3	H,S:A.A	PersonalSurvey - A.H	27	3	3	1	1	1	8	8	8	9	10	9	8	7	9	9	11	11
Subject-A.H-4	H,S:A.A	PersonalSurvey - A.H	34	7	5	1	1	1	10	10	4	10	11	11	7	8	6	11	11	10
Subject-A.H-M	H	PersonalSurvey - A.H	32	9	6	0	1	1	6	11	8	11	9	9	8	8	9	11	11	11
Subject-B.L-1	H,S:A.C	PersonalSurvey - B.L	27	2	2	1	0	1	2	11	8	11	11	11	11	11	8	11	11	11
Subject-B.L-2	H,S:A.C	PersonalSurvey - B.L	26	3	3	1	0	1	3	10	7	11	11	10	9	11	8	11	11	11
Subject-B.L-3	H,S:A.C	PersonalSurvey - B.L	33	9	4	1	0	1	11	10	7	11	9	10	6	10	4	11	11	11
Subject-B.L-4	H,S:A.C	PersonalSurvey - B.L	29	5	5	1	1	1	5	11	7	11	10	9	10	9	8	11	11	11
Subject-B.L-M	H	PersonalSurvey - B.L	29	7	8	1	1	1	10	11	6	9	9	10	8	9	8	11	11	11
Subject-B.H-1	H,S:A.A	PersonalSurvey - B.H	41	16	5	1	1	1	8	11	8	11	11	11	10	9	8	11	11	11
Subject-B.H-2	H,S:A.A	PersonalSurvey - B.H	34	14	6	0	1	1	7	10	8	11	10	11	9	8	7	9	11	11

Subject	Group	Task	A.1: Age	A.2: Working experience in years	A.3: Software tester experience in years	A.4: ISTQB Testing Foundation certificate	A.5: Have you ever evaluated banking related software?	A.6: Are you Internet banking user?	A.7: For how many years?	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-B.H-3	H,S:A.A	PersonalSurvey - B.H	24	6	4	1	0	1	6	10	9	9	11	9	11	10	10	11	11	11
Subject-B.H-4	H,S:A.A	PersonalSurvey - B.H	33	6	3	1	1	1	6	11	11	11	11	11	6	10	10	11	11	11
Subject-B.H-M	H	PersonalSurvey - B.H	34	11	5	0	1	1	11	11	6	11	9	9	8	6	4	10	11	11

Table A-1 Experiment 1 – Personal surveys raw data (source: own study)

Subject	Group	Task	$f_p$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-A.L-1	H,S:A.C	Evaluation App1 - A.L	0%	99	100%	30	100	2		8	8	7	11	11	8	10	3	6	6	11
Subject-A.L-2	H,S:A.C	Evaluation App1 - A.L	0%	79	100%	10	15	2	3	3	3	6	7	7	6	9	1	4	3	6
Subject-A.L-3	H,S:A.C	Evaluation App1 - A.L	0%	277	100%	200	100	2	4	9	4	9	9	8	8	9	1	9	9	4
Subject-A.L-4	H,S:A.C	Evaluation	0%	117	100%	40	90	2	2	6	3	8	11	9	6	9	1	11	1	1



Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
		App1 - A.L																		
Subject-A.L-M	H	Evaluation App1 - A.L	0%	n/a	n/a	320	n/a	7	10	6	3	4	11	9	6	9	2	8	1	1
Subject-A.H-1	H,S:A.A	Evaluation App1 - A.H	0%	194	100%	60	50	1	4	6	1	7	8	8	5	9	1	8	7	2
Subject-A.H-2	H,S:A.A	Evaluation App1 - A.H	0%	253	100%	70	70	0	2	5	3	7	10	10	6	9	1	9	6	
Subject-A.H-3	H,S:A.A	Evaluation App1 - A.H	0%	187	100%	60	100	0	1	9	7	8	9	10	7	9	8	9	9	9
Subject-A.H-4	H,S:A.A	Evaluation App1 - A.H	0%	351	100%	90	120	0	3	9	4	10	9	9	9	10	6	7	6	6
Subject-A.H-M	H	Evaluation App1 - A.H	0%	n/a	n/a	50	n/a			7	4	7	8	8	6	9	1	7	6	6
Subject-B.L-1	H,S:A.C	Evaluation App1 - B.L	0%	182	100%	60	80	6	7	4	5	6	11	11	4	8	1	10	1	1
Subject-B.L-2	H,S:A.C	Evaluation App1 - B.L	0%	147	100%	60	60	1	0	4	7	9	8	10	10	10	7	7	4	6
Subject-B.L-3	H,S:A.C	Evaluation App1 - B.L	0%	158	100%	75	130	5	4	5	3	6	7	7	7	9	1	8	7	6
Subject-B.L-4	H,S:A.C	Evaluation App1 - B.L	0%	317	100%	60	80	3	2	4	2	2	5	4	3	4	1	6	5	3
Subject-B.L-M	H	Evaluation App1 - B.L	0%	n/a	n/a	60	n/a	5	5	4	4	6	9	7	7	8	2	7	3	3

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-B.H-1	H,S:A.A	Evaluation App1 - B.H	0%	158	100%	80	110	0	0	8	5	8	10	10	10	11	11	10	10	10
Subject-B.H-2	H,S:A.A	Evaluation App1 - B.H	0%	143	100%	80	100	0	3	7	7	8	9	8	9	11	10	8	6	6
Subject-B.H-3	H,S:A.A	Evaluation App1 - B.H	0%	254	100%	100	100	2	6	5	7	8	11	11	10	11	6	6	10	9
Subject-B.H-4	H,S:A.A	Evaluation App1 - B.H	0%	293	100%	120	150	2	6	4	7	6	11	11	9	11	1	3	4	4
Subject-B.H-M	H	Evaluation App1 - B.H	0%	n/a	n/a	45	n/a	3	4	6	4	6	10	10	7	6		5	2	2
Subject-A.L-1	H,S:A.C	Evaluation App2v0.1 - A.L	0%	184	100%	40	100	1	1	7	8	9	11	11	9	10	5	9	8	10
Subject-A.L-2	H,S:A.C	Evaluation App2v0.1 - A.L	0%	299	100%	60	40	0	3	6	4	8	8	8	7	8	3	6	3	3
Subject-A.L-3	H,S:A.C	Evaluation App2v0.1 - A.L	0%	386	100%	180	150	2	7	9	7	7	8	9	9	11	3	8	3	4
Subject-A.L-4	H,S:A.C	Evaluation App2v0.1 - A.L	0%	79	100%	50	70	3	0	6	6	1	9	9	9	10	7	6	1	1
Subject-A.L-M	H	Evaluation App2v0.1 - A.L	0%	n/a	n/a	300	n/a	5	10	6	5	1	9	9	8	10	3	7	1	1
Subject-A.H-1	H,S:A.A	Evaluation App2v0.1 - A.H	0%	167	100%	90	80	0	3	8	8	8	10	9	8	9	1	9	8	6
Subject-A.H-2	H,S:A.A	Evaluation	0%	308	100%	90	80	0	4	8	3	8	10	9	8	10	1	8	6	6

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
		App2v0.1 - A.H																		
Subject-A.H-3	H,S:A.A	Evaluation App2v0.1 - A.H	0%	302	100%	180	300	1	10	9	9	10	10	10	9	9	9	10	6	5
Subject-A.H-4	H,S:A.A	Evaluation App2v0.1 - A.H	0%	501	100%	120	200	1	3	6	7	8	9	10	9	10	10	9	9	2
Subject-A.H-M	H	Evaluation App2v0.1 - A.H	0%	n/a	n/a	50	n/a	1		8	7	8	10	9	7	9	6	8	7	5
Subject-B.L-1	H,S:A.C	Evaluation App2v0.1 - B.L	0%	229	100%	80	80	8	5	5	6	6	11	8	6	6	4	9	5	3
Subject-B.L-2	H,S:A.C	Evaluation App2v0.1 - B.L	0%	172	100%	120	200	1	0	9	9	10	9	9	7	8	7	9	9	7
Subject-B.L-3	H,S:A.C	Evaluation App2v0.1 - B.L	0%	309	100%	90	150	0	4	8	7	8	9	8	8	10	4	9	9	8
Subject-B.L-4	H,S:A.C	Evaluation App2v0.1 - B.L	0%	294	100%	120	80	0	4	5	3	5	3	2	2	5	4	4	4	2
Subject-B.L-M	H	Evaluation App2v0.1 - B.L	0%	n/a	n/a	60	n/a	8	3	7	6	6	5	4	5	7	4	8	6	4
Subject-B.H-1	H,S:A.A	Evaluation App2v0.1 - B.H	0%	8	100%	180	190	0	2	7	7	7	5	5	6	10	9	10	10	10
Subject-B.H-2	H,S:A.A	Evaluation App2v0.1 - B.H	0%	314	100%	170	200	1	3	8	7	9	6	8	8	10	8	8	5	5
Subject-B.H-3	H,S:A.A	Evaluation App2v0.1 - B.H	0%	229	100%	2	150	2	3	8	8	9	10	11	10	11	9	9	9	10

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-B.H-4	H,S:A.A	Evaluation App2v0.1 - B.H	0%	661	100%	210	300	8	15	4	5	6	7	7	6	11	6	4	4	8
Subject-B.H-M	H	Evaluation App2v0.1 - B.H	0%	n/a	n/a	20	n/a	8	8	6	6	7	4	6	7	10	8	6	3	4
Subject-A.L-1	H,S:A.C	Evaluation App2v0.2 - A.L	0%	179	100%	30	100	0	2	9	8	10	11	11	10	10	10	8	6	9
Subject-A.L-2	H,S:A.C	Evaluation App2v0.2 - A.L	0%	169	100%	60	40	1	5	5	3	6	7	8	7	9	3	6	6	6
Subject-A.L-3	H,S:A.C	Evaluation App2v0.2 - A.L	0%	372	100%	180	150	2	10	7	7	7	9	9	9	10	3	8	2	2
Subject-A.L-4	H,S:A.C	Evaluation App2v0.2 - A.L	0%	82	100%	40	50	2	2	6	6		6	8	8	10	1	9	1	1
Subject-A.L-M	H	Evaluation App2v0.2 - A.L	0%	n/a	n/a	300	n/a	5	16	6	6	1	8	8	8	9	2	9	1	1
Subject-A.H-1	H,S:A.A	Evaluation App2v0.2 - A.H	0%	154	100%	90	100	0	3	9	6	9	9	8	8	10	1	10	6	1
Subject-A.H-2	H,S:A.A	Evaluation App2v0.2 - A.H	0%	166	100%	60	60	0	1	8	3	6	8	9	7	9	1	7	6	6
Subject-A.H-3	H,S:A.A	Evaluation App2v0.2 - A.H	0%	274	100%	3	400	2	10	9	8	9	9	9	7	9	7	9	6	6
Subject-A.H-4	H,S:A.A	Evaluation App2v0.2 - A.H	0%	601	100%	120	250	1	4	6	7	5	9	9	9	9	8	8	6	3
Subject-A.H-M	H	Evaluation	0%	n/a	n/a	60	n/a	2	12	8	7	8	9	9	8	9	5	8	6	6

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
		App2v0.2 - A.H																		
Subject-B.L-1	H,S:A.C	Evaluation App2v0.2 - B.L	0%	148	100%	80	40	1	8	4	6	4	10	10	5	5	5	10	9	3
Subject-B.L-2	H,S:A.C	Evaluation App2v0.2 - B.L	0%	82	100%	60	100	1	7	8	9	8	9	9	8	9	9	9	6	5
Subject-B.L-3	H,S:A.C	Evaluation App2v0.2 - B.L	0%	191	100%	60	180	1	4	8	4	9	11	10	8	9	3	8	9	6
Subject-B.L-4	H,S:A.C	Evaluation App2v0.2 - B.L	0%	183	100%	60	4	1	2	5	5	5	5	4	2	5	2	5	3	3
Subject-B.L-M	H	Evaluation App2v0.2 - B.L	0%	n/a	n/a	60	n/a	1	6	7	7	6	8	9	9	8	5	7	7	4
Subject-B.H-1	H,S:A.A	Evaluation App2v0.2 - B.H	0%	199	100%	180	240	3	2	8	8	8	8	10	8	10	8	8	8	8
Subject-B.H-2	H,S:A.A	Evaluation App2v0.2 - B.H	0%	331	100%	180	250	3	3	7	7	8	8	8	7	9	5	7	6	6
Subject-B.H-3	H,S:A.A	Evaluation App2v0.2 - B.H	0%	404	100%	2	150	3	5	9	9	10	10	11	11	11	4	3	9	10
Subject-B.H-4	H,S:A.A	Evaluation App2v0.2 - B.H	0%	325	100%	180	247	5	13	3	4	5	9	9	6	11	6	2	3	3
Subject-B.H-M	H	Evaluation App2v0.2 - B.H	0%	n/a	n/a	40	n/a	8	13	4	6	6	8	9	9	11	6	3	6	7
Subject-A.L-1	H,S:A.C	Evaluation App2v0.3 - A.L	10%	85	96%	30	100	0	1	9	8	9	10	10	10	10	7	7	9	9

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-A.L-2	H,S:A.C	Evaluation App2v0.3 - A.L	10%	247	91%	60	35	2	2	3	3	6	9	6	6	9	2	2	5	6
Subject-A.L-3	H,S:A.C	Evaluation App2v0.3 - A.L	10%	224	95%	90	100	3	10	6	8	8	4	9	9	11	4	2	7	4
Subject-A.L-4	H,S:A.C	Evaluation App2v0.3 - A.L	10%	47	92%	20	40	4	2	5	6	1	7	5	5	9	4	1	1	1
Subject-A.L-M	H	Evaluation App2v0.3 - A.L	10%	n/a	n/a	240	n/a	8	11	4	5	1	8	8	7	9	3	2	1	1
Subject-A.H-1	H,S:A.A	Evaluation App2v0.3 - A.H	10%	71	85%	60	50	2	3	1	9	9	6	9	2	9	1	9	6	6
Subject-A.H-2	H,S:A.A	Evaluation App2v0.3 - A.H	10%	215	92%	90	80	6	5	3	4	6	8	8	3	9	1	3	6	6
Subject-A.H-3	H,S:A.A	Evaluation App2v0.3 - A.H	10%	374	92%	2	200	2	10	5	9	8	7	8	4	8	8	2	5	5
Subject-A.H-4	H,S:A.A	Evaluation App2v0.3 - A.H	10%	269	92%	90	150	3	5	2	7	2	6	9	2	9	9	1	6	3
Subject-A.H-M	H	Evaluation App2v0.3 - A.H	10%	n/a	n/a	35	n/a	3	12	3	7	6	6	8	2	9	5	3	6	6
Subject-B.L-1	H,S:A.C	Evaluation App2v0.3 - B.L	80%	22	26%															
Subject-B.L-2	H,S:A.C	Evaluation App2v0.3 - B.L	80%	18	25%	60	30	5	10	1	8	4	2	4	1	6	6	2	3	3
Subject-B.L-3	H,S:A.C	Evaluation	80%	12	21%	30	50	30	30	1	6	1	7	3	1	8	3	1	3	5

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
		App2v0.3 - B.L																		
Subject-B.L-4	H,S:A.C	Evaluation App2v0.3 - B.L	80%	30	27%	30	15	5		1	5	5	1	1	1	4	4	1	2	2
Subject-B.L-M	H	Evaluation App2v0.3 - B.L	80%	n/a	n/a		n/a			1	4	3	2	3	2	5	4	2	2	2
Subject-B.H-1	H,S:A.A	Evaluation App2v0.3 - B.H	80%	17	24%	30	10		0	1	1	1	1	1	1	1	1	1	1	1
Subject-B.H-2	H,S:A.A	Evaluation App2v0.3 - B.H	80%	6	32%	45	10		0	1	1	1	1	1	1	1	1	1	1	1
Subject-B.H-3	H,S:A.A	Evaluation App2v0.3 - B.H	80%	13	19%	1	100	1	3	1	8	9	2	10	1	11	4	1	8	10
Subject-B.H-4	H,S:A.A	Evaluation App2v0.3 - B.H	80%	39	22%	45	40	1	0	1	1	1	1	1	1	1	1	1	1	1
Subject-B.H-M	H	Evaluation App2v0.3 - B.H	80%	n/a	n/a	10	n/a	1	2	1	1	1	1	1	1	1	1	1	1	1
Subject-A.L-1	H,S:A.C	Evaluation App2v0.4 - A.L	15%	33	94%	30	50	0	0	9	9	9	10	10	9	10	6	10	9	9
Subject-A.L-2	H,S:A.C	Evaluation App2v0.4 - A.L	15%	265	85%	50	30	3	3	3	3	5	3	3	2	9	2		4	6
Subject-A.L-3	H,S:A.C	Evaluation App2v0.4 - A.L	15%	162	85%	75	60	6	12	4	8	6	3	9	9	10	3	2	3	3
Subject-A.L-4	H,S:A.C	Evaluation App2v0.4 - A.L	15%	35	88%	30	60	2	2	5	6	6	8	8	6	9	3	4	1	1

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-A.L-M	H	Evaluation App2v0.4 - A.L	15%	n/a	n/a	220	n/a	10	18	3	4	1	8	8	4	10	2	6	1	1
Subject-A.H-1	H,S:A.A	Evaluation App2v0.4 - A.H	15%	120	88%	90	80	1	3	1	6	6	9	6	6	9	1	1	6	6
Subject-A.H-2	H,S:A.A	Evaluation App2v0.4 - A.H	15%	176	86%	70	70	15	4	3	2	6	10	9	2	9	1	4	5	5
Subject-A.H-3	H,S:A.A	Evaluation App2v0.4 - A.H	15%	524	87%	150	250	3	10	7	8	9	7	8	6	9	8	3	4	2
Subject-A.H-4	H,S:A.A	Evaluation App2v0.4 - A.H	15%	548	85%	120	300	3	3	3	6	3	6	7	4	7	7	2	3	2
Subject-A.H-M	H	Evaluation App2v0.4 - A.H	15%	n/a	n/a	30	n/a	3	10	3	6	6	6	7	4	8	5	3	5	3
Subject-B.L-1	H,S:A.C	Evaluation App2v0.4 - B.L	50%	93	56%	50	20	4	6	1	4	1	1	1	1	1	1	1	1	1
Subject-B.L-2	H,S:A.C	Evaluation App2v0.4 - B.L	50%	46	48%	60	60	3	2	2	9	6	2	4	1	6	6	2	3	2
Subject-B.L-3	H,S:A.C	Evaluation App2v0.4 - B.L	50%	40	55%	45	100	30	8	2	5	3	9	3	1	9	3	1	2	2
Subject-B.L-4	H,S:A.C	Evaluation App2v0.4 - B.L	50%	91	52%	60	30	5	5	1	3	2	2	2	1	4	3	1	2	2
Subject-B.L-M	H	Evaluation App2v0.4 - B.L	50%	n/a	n/a	60	n/a	20	6	2	6	3	4	3	2	5	3	2	2	2
Subject-B.H-1	H,S:A.A	Evaluation	50%	81	52%	120	100			1	1	1	1	1	1		1	1	1	1



Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
		App2v0.4 - B.H																		
Subject-B.H-2	H,S:A.A	Evaluation App2v0.4 - B.H	50%	27	50%	120	60			1	2	2	2	1	2	3	2	1	1	1
Subject-B.H-3	H,S:A.A	Evaluation App2v0.4 - B.H	50%	78	52%	2	150	8	6	1	9	8	2	10	1	11	4	2	4	8
Subject-B.H-4	H,S:A.A	Evaluation App2v0.4 - B.H	50%	66	56%	120	150	13	4	1	1	1	1	1	1	1	1	1	1	1
Subject-B.H-M	H	Evaluation App2v0.4 - B.H	50%	n/a	n/a	25	n/a	21	4	1	1	1	1	1	1	1	1	1	1	1
Subject-A.L-1	H,S:A.C	Evaluation App2v0.5 - A.L	10%	78	94%	30	100	1	1	10	9	10	11	11	10	10	6	9	9	9
Subject-A.L-2	H,S:A.C	Evaluation App2v0.5 - A.L	10%	128	90%	45	30	2	5	3	3	4	4	6	4	9	3	3	6	6
Subject-A.L-3	H,S:A.C	Evaluation App2v0.5 - A.L	10%	345	92%	120	100	7	15	3	7	5	3	9	4	10	3	2	3	4
Subject-A.L-4	H,S:A.C	Evaluation App2v0.5 - A.L	10%	90	95%	30	50	2	4	4	6	6	8	6	4	8	3	3	1	1
Subject-A.L-M	H	Evaluation App2v0.5 - A.L	10%	n/a	n/a	250	n/a	10	20	4	5	3	10	10	7	10	4	4	1	1
Subject-A.H-1	H,S:A.A	Evaluation App2v0.5 - A.H	10%	138	93%	90	80	1	3	3	6	6	9	6	6	9	1	9	6	6
Subject-A.H-2	H,S:A.A	Evaluation App2v0.5 - A.H	10%	371	91%	100	90	14	7	3	4	5	9	9	5	9	1	3	6	6

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-A.H-3	H,S:A.A	Evaluation App2v0.5 - A.H	10%	786	90%	240	400	2	15	6	8	8	8	9	7	9	8	4	4	4
Subject-A.H-4	H,S:A.A	Evaluation App2v0.5 - A.H	10%	416	92%	120	300	2	3	3	6	3	6	8	3	8	8	2	6	2
Subject-A.H-M	H	Evaluation App2v0.5 - A.H	10%	n/a	n/a	50	n/a	3	15	3	7	5	7	7	5	8	4	4	5	4
Subject-B.L-1	H,S:A.C	Evaluation App2v0.5 - B.L	10%	213	95%	80	50	3	8	2	4	3	8	8	3	3	3	3	8	4
Subject-B.L-2	H,S:A.C	Evaluation App2v0.5 - B.L	10%	111	91%	60	60	2	3	4	7	7	5	7	7	8	7	3	4	4
Subject-B.L-3	H,S:A.C	Evaluation App2v0.5 - B.L	10%	175	90%	60	130	20	10	2	5	5	9	4	2	9	3	1	3	3
Subject-B.L-4	H,S:A.C	Evaluation App2v0.5 - B.L	10%	255	93%	60	80	7	3	1	3	3	3	5	2	4	2	1	2	3
Subject-B.L-M	H	Evaluation App2v0.5 - B.L	10%	n/a	n/a	60	n/a	10	5	2	5	4	7	7	5	5	3	2	3	3
Subject-B.H-1	H,S:A.A	Evaluation App2v0.5 - B.H	10%	146	91%	120	100	8		1	1	1	2	2	1	4	1	1	1	1
Subject-B.H-2	H,S:A.A	Evaluation App2v0.5 - B.H	10%	229	92%	120	150	6	3	2	5	3	3	3	4	5	4	2	2	2
Subject-B.H-3	H,S:A.A	Evaluation App2v0.5 - B.H	10%	141	90%	1	100	7	10	2	6	9	3	4	3	11	2	1	9	9
Subject-B.H-4	H,S:A.A	Evaluation	10%	248	91%	120	150	10	5	2	3	2	5	8	4	9	4	1	1	1

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
		App2v0.5 - B.H																		
Subject-B.H-M	H	Evaluation App2v0.5 - B.H	10%	n/a	n/a	20	n/a	10	5	1	2	1	2	2	1	5	2	1	1	1
Subject-A.L-1	H,S:A.C	Evaluation App3/4 - A.C	10%	109	88%	30	100	1	2	6	5	3	5	9	8	10	7	8	9	9
Subject-A.L-2	H,S:A.C	Evaluation App3/4 - A.C	10%	221	91%	45	40	3	2	3	2	2	6	6	4	10	2	2	6	6
Subject-A.L-3	H,S:A.C	Evaluation App3/4 - A.C	10%	239	90%	100	70	7	9	3	6	6	8	9	9	9	2	2	4	4
Subject-A.L-4	H,S:A.C	Evaluation App3/4 - A.C	10%	28	100%	15	30	2	4	6	3	1	6	8	6	8	1	9	1	1
Subject-A.L-M	H	Evaluation App3/4 - A.C	10%	n/a	n/a	230	n/a	11		3	4	1	4	4	6	10	3	3	1	1
Subject-A.H-1	H,S:A.A	Evaluation App3/4 - A.A	10%	33	97%	60	30	1	2	6	3	6	6	6	6	9	1	3		6
Subject-A.H-2	H,S:A.A	Evaluation App3/4 - A.A	10%	9	69%	5	5	1	0	3	2	6	6	6	6	9	1	5	6	6
Subject-A.H-3	H,S:A.A	Evaluation App3/4 - A.A	10%	214	93%	200	250	2	5	8	8	8	8	9	7	9	8	4	4	2
Subject-A.H-4	H,S:A.A	Evaluation App3/4 - A.A	10%	394	91%	90	150	1	5	5	4	6	6	6	5	7	7	5	6	3
Subject-A.H-M	H	Evaluation App3/4 - A.A	10%	n/a	n/a	50	n/a	2	5	6	4	6	6	6	6	7	6	4	5	5

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-B.L-1	H,S:A.C	Evaluation App3/4 - A.C	10%	92	92%	40	30	3	6	3	4	3	10	8	4	5	4	4	5	1
Subject-B.L-2	H,S:A.C	Evaluation App3/4 - A.C	10%	96	93%	30	40	1	3	4	8	6	5	7	7	7	6	5		1
Subject-B.L-3	H,S:A.C	Evaluation App3/4 - A.C	10%	174	91%	60	100	5	5	6	6	7	9	8	5	9	2	5	6	5
Subject-B.L-4	H,S:A.C	Evaluation App3/4 - A.C	10%	241	92%	60	50	3	2	2	2	2	4	4	2	4	1	1	3	2
Subject-B.L-M	H	Evaluation App3/4 - A.C	10%	n/a	n/a	60	n/a	5	4	4	4	5	7	6	5	6	3	4	3	2
Subject-B.H-1	H,S:A.A	Evaluation App3/4 - A.A	10%	145	93%	60	80	1	5	8	8	9	6	6	6	10	6	6	4	4
Subject-B.H-2	H,S:A.A	Evaluation App3/4 - A.A	10%	74	86%	150	50	0	3	6	6	7	7	7	7	9	8	6	6	6
Subject-B.H-3	H,S:A.A	Evaluation App3/4 - A.A	10%	159	91%	1	100	4	6	2	6	9	3	11	6	11	2	3	10	9
Subject-B.H-4	H,S:A.A	Evaluation App3/4 - A.A	10%	239	92%	120	150	10	5	3	4	4	6	6	6	11	6	2	3	2
Subject-B.H-M	H	Evaluation App3/4 - A.A	10%	n/a	n/a	25	n/a	12	7	6	7	7	6	7	7	10	6	4	4	2

Table A-2 Experiment 1 – Raw data of evaluation tasks (source: own study)

## 2. EXPERIMENT 2

Remark: complete data for Experiment 2 consist also from data regarding subjects other than Managers from Experiment 1.

Subject	Group	Task	A.1: Age	A.2: Working experience in years	A.3: Software tester experience in years	A.4: ISTQB Testing Foundation certificate	A.5: Have you ever evaluated banking related software?	A.6: Are you Internet banking user?	A.7: For how many years?	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-A.L-5	S:A.C	PersonalSurvey - A.L	23	5	1	0	1	1	4	8	6	8	10	10	10	9	4	10	10	10
Subject-A.H-5	S:A.A	PersonalSurvey - A.H	47	20	9	1	1	1	3	7	7	11	9	8	4	6	8	9	11	11
Subject-B.L-5	S:A.C	PersonalSurvey - B.L	25	1	1	0	0	1	4	11	8	11	9	11	10	11	9		11	11
Subject-C.C-1	S:C.C	PersonalSurvey - C.C	50	29	5	1	1	1	6	11	10	11	11	11	11	10	10	11	11	11
Subject-C.C-2	S:C.C	PersonalSurvey - C.C	45	20	10	0	1	1	13	10	10	11	8	11	7	8	6	11	11	11
Subject-C.C-3	S:C.C	PersonalSurvey - C.C	43	15	1	0	0	1	6	11	8	11	10	11	8	10	9	11	11	11
Subject-C.C-4	S:C.C	PersonalSurvey - C.C	42	20	4	0	0	1	7	11	9	11	10	9	8	2	9	11	11	11
Subject-C.C-5	S:C.C	PersonalSurvey - C.C	26	0	0	0	1	1	4	10	9	9	11	11	10	6	7	11	9	11
Subject-C.C-6	S:C.C	PersonalSurvey - C.C	36	20	8	0	1	1	10	11	7	11	8	9	9	9	7	11	11	11
Subject-C.C-7	S:C.C	PersonalSurvey - C.C	54	20	15	1	1	1	10	9	8	11	10	11	9	10	9	11	11	11

Subject	Group	Task	A.1: Age	A.2: Working experience in years	A.3: Software tester experience in years	A.4: ISTQB Testing Foundation certificate	A.5: Have you ever evaluated banking related software?	A.6: Are you Internet banking user?	A.7: For how many years?	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-C.A-1	S:C.A	PersonalSurvey - C.A	26	5	2	0	1	1	7	11	7	11	11	10	7	9	9	11	11	11
Subject-C.A-2	S:C.A	PersonalSurvey - C.A	45	20	15	1	1	1	10	11	9	11	10	10	11	11	10	11	11	11
Subject-C.A-3	S:C.A	PersonalSurvey - C.A	28	5	2	1	1	1	8	10	8	11	11	11	8	8	6	8	11	10
Subject-C.A-4	S:C.A	PersonalSurvey - C.A	40	16	5	0	1	1	9	11	8	9	10	10	10	11	9	11	11	11
Subject-C.A-5	S:C.A	PersonalSurvey - C.A	36	12	1	0	1	1	6	11	10	11	11	11	9	9	6	11	11	11

Table A-3 Experiment 2 – Personal surveys raw data (source: own study)

Subject	Group	Task	$f_p$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-A.L-5	S:A.C	Evaluation App1 - A.L	0%	139	100%	45	100	0	1	6	3	7	11	8	6	10	2	8	6	7

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-A.H-5	S:A.A	Evaluation App1 - A.H	0%	151	100%	70	30	2	3	5	5		4	5	5	8	8	6	6	6
Subject-B.L-5	S:A.C	Evaluation App1 - B.L	0%	182	100%	40	40	0	4	9	9	9	9	10	10	10	7	9	9	9
Subject-A.L-5	S:A.C	Evaluation App2v0.1 - A.L	0%	92	100%	45	80	0	1	5	4	6	10	9	8	10	1	8	7	4
Subject-A.H-5	S:A.A	Evaluation App2v0.1 - A.H	0%	222	100%	80	40	0	3	5	7		8	9	4	8	6	7	7	7
Subject-B.L-5	S:A.C	Evaluation App2v0.1 - B.L	0%	268	100%	70	40	0	2										1	1
Subject-A.L-5	S:A.C	Evaluation App2v0.2 - A.L	0%	31	100%	45	100	0	1	6	6	8	11	8	7	8	2	9	3	3
Subject-A.H-5	S:A.A	Evaluation App2v0.2 - A.H	0%	291	100%	70	50	0	0	8	8	8	8	8	7	8	6	7	8	8
Subject-B.L-5	S:A.C	Evaluation App2v0.2 - B.L	0%	103	100%	60	30	0	1	10	10	10	10	10	10	10	4	10	6	6
Subject-A.L-5	S:A.C	Evaluation App2v0.3 - A.L	10%	50	94%	45	100	0	2	4	4	6	11	8	5	9	1	4	5	3
Subject-A.H-5	S:A.A	Evaluation App2v0.3 - A.H	10%	235	92%	60	40	1	4	3	3	3	3	3	3	3		3	3	6
Subject-B.L-5	S:A.C	Evaluation App2v0.3 - B.L	80%	31	27%	30	20	5	10	2	2	2	2	2	2	10	4	2	2	2
Subject-A.L-5	S:A.C	Evaluation	15%	54	89%	45	100	0	4	4	4	7	11	8	4	8	1	3	4	1

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
		App2v0.4 - A.L																		
Subject-A.H-5	S:A.A	Evaluation App2v0.4 - A.H	15%	211	85%	75	50	1	1	3	3	3	4	4	2	6	3	3	3	3
Subject-B.L-5	S:A.C	Evaluation App2v0.4 - B.L	50%	71	58%	40	25	3	5	2	7	10	10	10	4	10	1	2	2	2
Subject-A.L-5	S:A.C	Evaluation App2v0.5 - A.L	10%	47	90%	45	100	0	4	4	3	7	11	8	5	8	1	3	2	2
Subject-A.H-5	S:A.A	Evaluation App2v0.5 - A.H	10%	482	91%	70	70	2	2	2	3	3	2	4	1	6	3	1	1	1
Subject-B.L-5	S:A.C	Evaluation App2v0.5 - B.L	10%	80	91%	40	25	2	2		8	3	10	10	9	9	2	2	2	2
Subject-A.L-5	S:A.C	Evaluation App3/4 - A.C	10%	81	95%	45	60	0	2	6	4	8	11	9	8	10	1	8	4	4
Subject-A.H-5	S:A.A	Evaluation App3/4 - A.A	10%	200	92%	70	50	2	2	3	4	5	4	4	3	6	5	2	5	4
Subject-B.L-5	S:A.C	Evaluation App3/4 - A.C	10%	230	91%	60	30	2	1	7	7	10	10	10	7	10	2	4	2	2
Subject-C.C-1	S:C.C	Evaluation App3/4 - C.C	10%	187	91%	120	50	1	10	4	7	4	9	10	5	10	2	4	6	6
Subject-C.C-2	S:C.C	Evaluation App3/4 - C.C	10%	237	92%		40	2	3	7	3	6	10	10	3	10	1	10	6	3
Subject-C.C-3	S:C.C	Evaluation App3/4 - C.C	10%	149	91%	30	30	0	4	5	5		8	8	5	7		9	5	5



Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Subject-C.C-4	S:C.C	Evaluation App3/4 - C.C	10%	146	92%	60	30	0	1	6	5	6			8	10	5		7	7
Subject-C.C-5	S:C.C	Evaluation App3/4 - C.C	10%	317	93%	90	40	2	5	4	2	8	10	10	11	11	2	6	9	5
Subject-C.C-6	S:C.C	Evaluation App3/4 - C.C	10%	106	90%	40	20	3	3	6	5	9	9	9	7	10	6	4	8	6
Subject-C.C-7	S:C.C	Evaluation App3/4 - C.C	10%	584	92%	120	300	5	3	6	4	8	8	9	6	9	4	3	4	1
Subject-C.A-1	S:C.A	Evaluation App3/4 - C.A	10%	158	93%	50	100	2	1	9	4	10	11	11	10	11	6	9	4	6
Subject-C.A-2	S:C.A	Evaluation App3/4 - C.A	10%	297	90%	80	70	3	3	8	8	10	10	10	11	11	7	6	9	9
Subject-C.A-3	S:C.A	Evaluation App3/4 - C.A	10%	305	91%	50	550	1	4	9	8	6	10	10	9	10	3	8	5	6
Subject-C.A-4	S:C.A	Evaluation App3/4 - C.A	10%	74	93%	20	10	0	0	6	5	7	9	9	8	9	6	9	7	7
Subject-C.A-5	S:C.A	Evaluation App3/4 - C.A	10%	204	91%	50	65	3	6	9	6	9	8	8	9	11	10	7	10	10

Table A-4 Experiment 2 – Raw data of evaluation task (source: own study)

### 3. EXPERIMENT 3

Subject	Group	Task	A.1: Age	A.2: Working experience in years	A.3: Software tester experience in years	A.4: ISTQB Testing Foundation certificate	A.5: Have you ever evaluated banking related software?	A.6: Are you Internet banking user?	A.7: For how many years?	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Figurehead-POS-1	G	PersonalSurvey - POS	29	7	0	0	0	1	8	11	6	11	10	11	8	10	9	11	11	11
Figurehead-POS-2	G	PersonalSurvey - POS	23	0	0	1	1	1	4	11	9	11	10	11	10	10	11	11	11	11
Subject-POS-1	G	PersonalSurvey - POS	32	11	0	0	1	1	9	9	11	11	8	10	9	8	4	10	11	10
Figurehead-POS-3	G	PersonalSurvey - POS	22	1	0	0	0	1	2	11	9	10	8	8	7	7	7	10	11	11
Subject-POS-2	G	PersonalSurvey - POS	59	35	0	0	0	1	10	10	10	11	11	11	11	11	11	11	11	11
Figurehead-POS-4	G	PersonalSurvey - POS	27	5	0	0	0	1	7	10	9	11	9	9	10	8	7	8	11	11
Figurehead-POS-5	G	PersonalSurvey - POS	23	1	0	0	1	1	6	10	9	11	9	9	9	10	8	11	11	11
Subject-POS-3	G	PersonalSurvey - POS	45	25	0	0	0	1	10	10	10	11	10	11	10	11	11	11	11	11
Subject-POS-4	G	PersonalSurvey - POS	23	3	0	0	0	1	4	10	8	11	10	11	8	9	4	10	11	11
Subject-POS-5	G	PersonalSurvey - POS	22	1	0	0	0	1	2	6	8	9	10	11	8	4	7	10	11	11
Figurehead-	G	PersonalSurvey - POS	34	10	0	0	1	1	8	1	4	1	1	1	1	1	1	1	1	1

Subject	Group	Task	A.1: Age	A.2: Working experience in years	A.3: Software tester experience in years	A.4: ISTQB Testing Foundation certificate	A.5: Have you ever evaluated banking related software?	A.6: Are you Internet banking user?	A.7: For how many years?	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
POS-6		rvey - POS																		
Subject-POS-6	G	PersonalSurvey - POS	24	2		0	0	1	4	9	11	10	11	11	9	6	10	11	11	11
Subject-NEG-1	G	PersonalSurvey - NEG	29	5	0	0	0	1	8	9	8	7	10	11	10	7	7	9	10	8
Figurehead-NEG-1	G	PersonalSurvey - NEG	23	2	0	0	1	1	6	10	4	10	10	10	8	8	4	11	11	11
Figurehead-NEG-2	G	PersonalSurvey - NEG	30	6	0	0	0	1	10	8	6	6	10	11	11	10	2	9	9	11
Subject-NEG-2	G	PersonalSurvey - NEG	34	10	0	1	1	1	6	10	7	11	10	11	11	11	8	11	11	11
Figurehead-NEG-3	G	PersonalSurvey - NEG	25	2	0	0	1	1	6	11	10	11	11	11	10	7	11	11	11	11
Subject-NEG-3	G	PersonalSurvey - NEG	34	9	0	0	0	1	8	10	10	9	8	8	7	7	10	10	11	11
Figurehead-NEG-4	G	PersonalSurvey - NEG	24	5	0	0	0	1	7	9	7	11	9	9	3	6	2	11	11	11
Subject-NEG-4	G	PersonalSurvey - NEG	31	5	0	0	0	1	3	11	7	11	9		6	9	6	11	11	11
Subject-NEG-5	G	PersonalSurvey - NEG	29	4	0	0	1	1	4	9	7	10	6	10	5	6	3	9	11	10

Subject	Group	Task	A.1: Age	A.2: Working experience in years	A.3: Software tester experience in years	A.4: ISTQB Testing Foundation certificate	A.5: Have you ever evaluated banking related software?	A.6: Are you Internet banking user?	A.7: For how many years?	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Figurehead-NEG-5	G	Personal Survey - NEG	24	2	0	0	1	1	5	9	7	10	10	9	7	8	9	9	11	11

Table A-5 Experiment 3 – Personal surveys raw data (source: own study)

Subject	Group	Task	$f_p$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Figurehead-POS-1	G	Evaluation - POS	20%	113	88%	40	90	10	20	7	7	11	10	10	8	10	3	2	4	4
Figurehead-POS-2	G	Evaluation - POS	20%	73	83%	30	30	2	8	6	4	8	11	9	10	9	5	4	3	3
Subject-POS-1	G	Evaluation - POS	20%	45	85%	30	45	3	7	4	1	2	8	8	5	4	2	3	3	1
Figurehead-POS-3	G	Evaluation - POS	20%	52	85%	25	40	4	4	4	8	5	5	4	8	9	4	3	9	6
Subject-POS-2	G	Evaluation - POS	20%	34	92%	20	5	5	3	6	7	7	5	6	5	6	6	6	5	5
Figurehead-POS-4	G	Evaluation - POS	20%	89	91%	42	40	3	8	7	8	10	8	7	6	10	9	5	4	2

Subject	Group	Task	$f_b$	Correct actions	% Correct action	A.1: Declared evaluation time in minutes	A.2: Declared number of operations performed	A.3: Number of critical failures observed	A.4: No of other failures observed	B.1: General product's quality	B.2: Rich functionality	B.3: Compliance with formal rules	B.4: Efficiency (performance indicator)	B.5: Productivity (performance of a user working with the product)	B.6: Pleasentness	B.7: Learnability	B.8: Ability to customize to one's needs	B.9: Robustness	B.10: Safety	B.11: Security
Figurehead-POS-5	G	Evaluation - POS	20%	109	89%	30	20	3	10	4	3	8	5	5	2	9	1	2	2	2
Subject-POS-3	G	Evaluation - POS	20%	63	81%	30	20	3	2	9	7	9	2	6	6	6	3		1	1
Subject-POS-4	G	Evaluation - POS	20%	96	80%	40	28	4	3	9	8	4	6	10	9	11	10	7	8	7
Subject-POS-5	G	Evaluation - POS	20%	46	82%	40	30	5	10	7	5	5	3	8	5	8	5	3	6	4
Figurehead-POS-6	G	Evaluation - POS	20%	85	85%		60	1	20	7	7	3	4	5	5	1	6	8	11	8
Subject-POS-6	G	Evaluation - POS	20%	82	85%	30	18	0	2	4	7	8	7	8	9	11	9	3	3	5
Subject-NEG-1	G	Evaluation - NEG	20%	54	83%	30	20	20	20	2	3	6	1	3	1	4	3	1	3	
Figurehead-NEG-1	G	Evaluation - NEG	20%	117	85%	35	100	5	20	3	3	4	1	2	1	3	3	4	4	1
Figurehead-NEG-2	G	Evaluation - NEG	20%	74	83%	30	50	30	20	1	7		4	6		1	1	1	2	
Subject-NEG-2	G	Evaluation - NEG	20%	53	80%	30	30	20	3	1	3	11	2	4	1	5	1	1	3	4
Figurehead-NEG-3	G	Evaluation - NEG	20%	30	83%	30	50	40	20	1	3	1	1	1	1	4	3	1	2	2
Subject-NEG-3	G	Evaluation - NEG	20%	44	81%	30	40	20	10	1	2	2	2	2	2	3	3	1	1	1
Figurehead-NEG-4	G	Evaluation - NEG	20%	251	81%	30	50	25	10	3	2	4	4	3	3	3	3	2	2	2
Subject-NEG-4	G	Evaluation - NEG	20%	57	86%	30	30	10	5	4	5	7	5	7	3	4	6	2	5	3
Subject-NEG-5	G	Evaluation - NEG	20%	45	80%	35	50	10	15	3	4	5	4	6	3	7	3	3	7	4
Figurehead-	G	Evaluation - NEG	20%	90	87%	30	50	10	10	3	4	6	5	5	3	5	2	2	6	6

