

# Web Applications Quality Perception

Radosław Hofman<sup>1</sup>

**Abstract**— This article analyzes experiment results regarding subjective perception of a web application. Software quality models, since the first publications on this subject, propose a prescriptive approach. Although most of the models are well explained and applicable, they still do not describe the real process taking place in a user's mind.

Behavioral economics, psychology, philosophy and cognitive sciences have developed several theories regarding perception, the valuation of goods and judgments formulation. An application of these theories to software engineering and an intentional management of the user's perception processes can significantly increase their satisfaction level and general quality grade assigned by the user to the software product.

In this article we concentrate on a part of the software quality perception process: the history effect and its influence on software quality perception.

**Index terms**—web application, quality perception, cognitive psychology, behavioral economics.

## I. INTRODUCTION AND MOTIVATION

### A. Motivation

Software engineering aims to deliver high quality software products. Although a similar goal is common for most of the engineering disciplines, software engineering scientists underline that software products are significantly different from all other human crafted products. Intangible software products also seem to be much more complex in the aspect of quality measurement.

On the other hand at every stage of the software production lifecycle, when the software product is presented to individuals (e.g. users), they tend to formulate their own opinion about the quality of the product. Even more, they formulate their opinion in a relatively short time. How is it possible if we consider the fact, that there is no commonly accepted software quality model nor a software evaluation process model? One of the possible answers is a conclusion, that users base their opinion on some other process and different software quality definition than those ones presented in literature.

We have identified the lack of a comprehensive descriptive model explaining the real process of software quality assessment. In consequence we have proposed a theoretical model resulting from cognitive sciences studies [9]. In this article we present the evidence supporting the validity of the discussed model regarding the area of the influence of knowledge on the quality perception. The observer's

knowledge, according to the model presented on fig 1, influences the perception filter (focusing on the most important characteristics), the perception of attributes (perception – combination of observations into a conveyed object) and the weights assigned to the observed characteristics. This article concentrates mainly on the perception part of this influence.

### B. Background

Software quality has been a subject of study since the 1970's when software development techniques started to be perceived as an engineering discipline. The first quality models were published by McCall [12] and Boehm [5]. Successive attempts continue and the most current one is the SQuaRE (Software product QUality Requirements and Evaluation) model developed within the ISO/IEC25000 standards series. This new approach is perceived as the new generation of software quality models [22] and is being used for the decomposition of the end users perspective to software components requirements [1].

The general conclusion about software quality models should observe that there is no commonly accepted model nor is there a commonly accepted evaluation method. On the other hand users and customers are in fact using some model and method to evaluate software.

The valuation of goods has been studied by economic scientists for centuries [7]. Many researchers have also tried to investigate how a personal value grade may be influenced (or fail to be influenced) in the aspect of a cognitive process associated with judgment formulation (compare Lawrence Kohlberg, Max Weber, von Weiser etc.)

The neo-classical economic model of human behavior lays upon assumptions about utility maximization, equilibrium and efficiency. These assumptions correspond with the classical model of human behavior known as *homo economicus*. The concept had appeared in the book considered to be the beginning of the economics science [19]. Although discussed assumptions are widely accepted they are just a simplifications made for the purpose of modeling the decision processes or economic behavior. Publications in the last years have put the above assumptions under critic [7]. The first publication drawing the attention to limitations of the *homo economicus* concept was the author of this idea Adam Smith. In *The theory of moral sentiments* [20] the author describes the asymmetric reaction to the increase and decrease of wealth. This observation was investigated in the 20<sup>th</sup> century by Daniel Kahneman and Amos Tversky [10].

The economists begrudgingly accepted the counterexamples to neo-classical models based on empirical observation results. The new approach in psychology, cognitive psychology, had proposed a new model of human brain using a metaphor of information processing system [14]. Psychologists start to compare their psychological models with the economics ones. Daniel Kahneman and Amos Tversky had

<sup>1</sup> EUR ING, Sygnity Research, Poland  
also PhD Student Department of Information Systems at The Poznań  
University of Economics and Polish Standardization Committee  
Member, email: [rhofman@sygnity.pl](mailto:rhofman@sygnity.pl)

published the research results for the heuristics in decision making [24] and the prospect theory [10] considered to be the two most important milestones of behavioral economics [7].

The works of Herbert A. Simon [18], Garry S. Becker [4], George A. Akerlof [2], A. Michael Spence, Joseph E. Stiglitz, and Daniel Kahneman [10] were awarded with the Bank of Sweden Prize in Memory of Alfred Nobel in 1978, 1992, 2001 and 2002 respectively. The prize for Daniel Kahneman was shared with Vernon L. Smith awarded for the research results in experimental economy.

Modern experimental psychology, understood as a psychological research area, follows ideas proposed by Wilhelm Wundt, who had established the first laboratory for psychological experiments in the 19<sup>th</sup> century near Lipsk (Leipzig) in Germany [6]. Boring concludes, that the psychology scientists were always interested in perception issues which explains the mentioning of this curiosity in literature from the Middle Ages [11]. Modern researchers take advantage of the previous achievements especially in the area of rules for scientific control and the usage of structuralized experimentation plans with known factors of strength.

One of the first quality perception models adopting cognitive approach for certain type of products was proposed by Steenkamp et al. for food quality perception [21]. Their research on the model validity was conducted in psychological research paradigm using an independent groups plan.

Experiments are to trace the cause-effect relations between certain values of an independent variable and the resulting level of a dependant variable(s). Tracing such changes in human's attitude and their judgments is methodologically complex due to a relatively high threat from the factors beyond full control of the experimenter. The researcher should not only describe investigated phenomenon but also prove the time sequence (cause-effect) and describe the future behavior for the independent variable changes [17].

In the summary of background analysis, described broader in [9], it should be stressed, that analyzed areas: software engineering, software quality engineering, behavioral economics and cognitive psychology are in a continuous development stage. Despite this fact, software quality psychology is able to take advantage of those research results focusing on issues related to software quality perception.

### C. The software quality perception model

Software quality psychology is a new research area focusing on the description of cognitive processes associated with the perception of software quality [9]. This research area is still being defined and this article is one of the first presenting the experimental evidence supporting this area. First research concentrates on the software quality perception model presented in fig. 1.

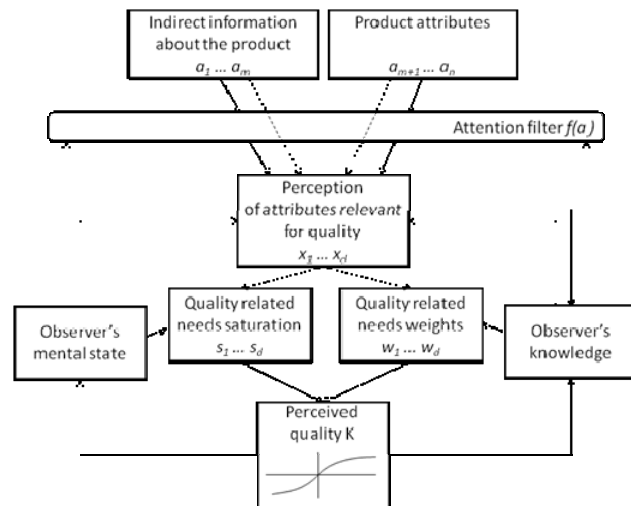


Fig. 1, The software quality perception model

The overall quality grade depends on the knowledge based importance of characteristics and also the current needs saturation. If the observed quality is above expectations then a diminishing marginal increases caused by each “quality unit” may be expected (Gossen’s law). On the other hand if the observed quality is below expectations then a radical dissatisfaction, non proportional to the “quality gap” (positive-negative asymmetry), may be expected [23].

Furthermore, both observer’s knowledge and the observer’s mental state influence the perception of software quality characteristics (e.g. supplies the process with valid or apparent associations etc.). Also both of these structures influence the behavior of the attention filter.

The perceived general quality grade is a non linear combination of perceived attributes  $K(x,w,s)=\sum_i F_i(s_i,w_i,x_i)$  with an open question what is taking place first: judgments about single attributes or the general grade assessment.

## II. THE EXPERIMENT

### A. Description

In this section the techniques and decisions made for the experiment preparation will be described. The first problem to be solved regarding the research area is the problem of preparing a comparable and controllable environment. The number of sources possibly influencing the perception should be considered as high. Especially in the regard to testing a web application task. The perception may be affected by information in media, rumors, previous experience of subjects, infrastructure or access failures during evaluation and many other. In the above list the most difficult to handle is the problem of differences between IT products and projects delivering these products. Each product has its own requirements, context of software usage, GUI layout, history of project conduction etc. From this perspective it is rather unlikely to have independent IT projects with a controllable list of differences.

A decided, real-like environment was designed and prepared for the experiment. To simulate a real project environment, a special application framework has been prepared – TestLab. The details of this will not be discussed here

except the most important factors for the purpose of the described experiment.

TestLab is a framework which allows the handling of subjects profiles, assignments and monitoring of evaluation tasks, gathering feedback from subjects etc. A more important factor is the ability to deploy real-like web applications (called TestApps) with a controllable fault probability level which is negatively correlated with the quality level.

Each screen of TestApp is internally described with the categories of possible (observable) failures. With the assignment of a task to a subject, the experimenter sets the general probability of failure for the individual task with weights between different categories of failures. The failure probability function is negatively correlated with quality level of an application thus with the use of this function the experimenter is able to control the quality level.

TestLab is generating 12 failure types (list based on categorized bugs reports from >100 real projects). This list contains: "Blue screen" (application produces lots of technical information about failure and stops working), Performance error (application hangs for 90 seconds), Data lost error (after a filled form is submitted the application reacts like it was an empty form – there are two types of this error – while writing or reading), Data change error (the application stores different data than submitted by the user – there are also two types of this error – while writing or reading), Calculation error (the application returns an incorrect calculation result), Presentation error (the application presents the screen as a maze or another kind of this error: the screen is presented with scrambled static texts on the screen), Form reset error (every 2-10 seconds the whole form is being reset), Inaccessibility of function (the possibility of performing the next step is inactive), Irritating messages error (the application displays some sequential messages about errors on window presentation but continues normal function).

For the purpose of the described experiment two TestApps were prepared: the web based issue submitting system (TestApp1) and the internet banking system (TestApp2). Each application has complete documentation for evaluation purposes: the requirements, the test scenarios etc.

The general experiment plan is an independent groups plan. There are four groups in the experiment with different treatment: two GFI and two BFI groups evaluating applications with differing fault probability as shown in fig 2. Two HM and LM groups differ in additional motivation existing in HM groups. The Cartesian Product of these two partitions consists of four groups: GFI.HM, GFI.LM, BFI.HM and BFI.LM.

The experiment's scenario composes of the following stages:

- Subjects complete their profile surveys containing questions about the importance of software quality characteristics
- Subjects are evaluating TestApp1 (failure probability FP=0%) on Friday (all groups are starting the experiment at the same time) assessing quality of the evaluated application on completion of the task
- For the following week subjects are evaluating TestApp2 (FP is being changed as shown in fig 2)

assessing quality at the end of each evaluation cycle

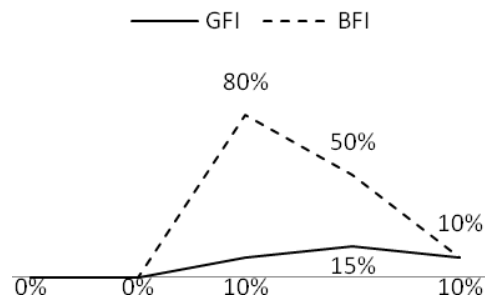


Fig. 2, TestApp2 sequential FP levels for GFI and BFI groups

As it is shown in fig. 2 for first two days all of the groups have the same quality level, then BFI groups gets the version with FP=80%, while for GFI FP=10%. For the last cycle all of the group evaluate the application with the same FP level FP=10%. On the last day the difference between quality assessed by groups may be the consequence of recollections from the previous days but in the technical manner the applications are the same.

After each evaluation, subjects are assessing a version's quality using a survey (an analogical survey as they have used in the profiling stage). Surveys use Likert-type scales having bipolar terms at the ends, following Osgood's semantic differential [16]. Questions are asked about: rich functionality, general software quality, compliance with formal rules, efficiency, productivity, satisfaction, learnability, adaptability, reliability, safety and security<sup>2</sup>. 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]. In the middle point the neutral anchor is defined as *Application has this quality attribute but not on an exceptional level* [value=6]. The scale is intended to look like a continuous scale (using red color on the negative and green at the positive and with gradient change between). The way of presentation is shown in fig 3.

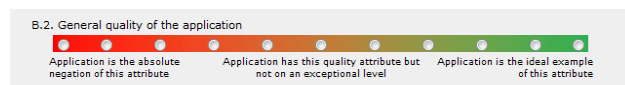


Fig. 3, Scale example for the general quality question

All instructions and communications, expect additional motivation communicates, were identical for all of the groups.

### B. Execution

For the purpose of the a purposive sampling method was chosen. Subjects were to be physically separated (to avoid information exchange). Professional software evaluators, having similar experience backgrounds, were used as subjects. The application TestApp2 was designed to simulate an internet banking system and 100% of the evaluators are users of such applications in the real world. The groups equality

<sup>2</sup> The list is based on Software Product Quality in Use as in ISO/IEC 25010 Commission Draft, 2008

analysis will be presented in section II.C.

The experiment was conducted according to an independent groups plan and the experiment plan discussed in the previous section. There were four independent groups: two following the BFI quality path and two following the GFI (as presented on fig 2). The two groups (one BFI and one GFI group) were told that it is important to assess quality adequately, and two others were left without additional instructions.

After TestApp1 tests, one of the groups had lost one person (due to illness) – as this person did not take part in TestApp2 tests we decided to continue the experiment. The evaluation during the first two days of TestApp2 tests had no remarkable occurrences. On the third day the quality was decreased and the evaluators stopped the tests complaining to their managers about the dramatically low quality level. They were told to continue and do as many test-scenarios as they were able to (the mail message was sent equally to all groups). The following days did not bring any new occurrences not expected in the plan.

**C. Validity analysis**

To assess validity properly, the internal validity should be discussed first. In this area the quality of the samples and the sampling method itself has to be considered. To analyze the strength of the observed effect the groups homogeneity has to be shown. Below there are two tests of groups homogeneity presented.

The first test is based on the null hypothesis  $H_0: M_1=M_2=M_3=M_4$  based on the declared quality importance in the first survey. The declared importance is presented in table 1.

Tab. 1, The declared importance of quality for four groups

GFI LM	GFI HM	BFI LM	BFI HM
10	9	11	11
11	10	10	10
11	8	10	10
11	10	11	11

For these four groups the ANOVA method: the F-test [17] will be applied, to compare inter group differences. Used method for the two-groups comparison is analogical to Student's test for two groups [8].

Homogeneity of joint GFI groups and BFI groups is calculated using an ANOVA table as shown in table 2. The second row of the table presents the same test for homogeneity of joint LM and HM groups.

Tab. 2, F-test for the groups homogeneity

$M_{G1}$	$M_{G2}$	SS	SSE	F	p	$F_{crit 5\%}$
GFI – 10,0	BFI – 10,5	1,0	10,0	1,4	0,27	4,6
LM – 10,6	HM – 9,9	2,3	8,8	3,6	0,08	4,6

Homogeneity of the groups was also tested by a pre-test as described in the previous section. Groups were given the same application with the same quality level. The quality grade assigned to this application is presented in table 3.

Tab. 3, Pre-test quality assessment

GFI LM	GFI HM	BFI LM	BFI HM
7	7	6	8
6	7	9	8
9	8	6	8
8	10	2	6

Again the test is performed with use of pair wise F-test assuming the null hypothesis to be true.

Tab. 4, F-test for the pre-test

$M_{G1}$	$M_{G2}$	SS	SSE	F	p	$F_{crit 5\%}$
GFI – 7,8	BFI – 6,6	5,1	45,4	1,6	0,23	4,6
LM – 6,6	HM – 7,8	5,1	45,4	1,6	0,23	4,6

As presented above, there are no arguments to reject the null hypothesis for all of the homogeneity tests, on the preset confidence level of  $\alpha=5\%$ . The analysis shows then that equivalency of the groups should be assumed.

Confounding threads were mitigated by employment of typical commercial procedures and employees. The evaluation project was described as a commercial one and passed to a professional software evaluation company. The company had prepared four teams in separate locations and four test managers located outside the location, where their teams were performing the evaluation. Communication procedures were identical to those, which the company was using in a typical evaluation project. Subjects in this experiment were professional software evaluators holding international certificates in testing. These circumstances comply with typical modern behavioral economics research [25].

The external validity is a general measure of the likelihood that the observed reaction will take place in the future. According to Mook [13] if one is testing a theory based on psychological studies then external validity should be based on the effect size not the statistical significance of the sample [17]. This observation uses a corollary that behavior patterns are rather constant even in different situations [25]. In the case of the discussed experiment the research is focusing on psychological theories.

External validity is typically lower when the experiment is conducted in pre-set and constant conditions (such experiments are more sensitive), and is typically greater when the balancing methods are employed to control variations of variables [17].

**D. Results analysis**

The results of the experiment will be analyzed with the F test and ANOVA method [17]. For the general effect GFI and BFI will be jointly analyzed as well as HM (high motivation) and LM (low motivation) groups.

Quality of the last version of TestApp2 was graded as shown in table 5.

Tab. 5, Quality grade of the last version for four groups

GFI LM	GFI HM	BFI LM	BFI HM
10	3	2	1
3	3	4	2
3	6	2	2
4	3	1	2

The analysis of data shows that during the experiment the floor effect could have taken place (range between data in BFI HM is the lowest among groups). The low end of the scale was described as “total negation of quality” thus the results should be interpreted as the highest possible negative grade.

The null hypothesis  $H_0$  assuming that there is no influence from the history effect is being verified first. The calculation of the F-test is shown in table 6 in the first row.

Tab. 6, F-test for the last version – the history effect and motivation effect

$M_{G1}$	$M_{G2}$	SS	SSE	F	p	$F_{crit 5\%}$
GFI – 4,4	BFI – 2,0	22,6	49,9	6,3	0,02	4,6
LM – 3,6	HM – 2,8	3,1	69,4	0,6	0,44	4,6

The null hypothesis is rejected on the pre-set confidence level  $\alpha=5\%$ . Estimation of the effect size requires additional statistic to be calculated – Cohen’s  $d$  [17]. The value of  $d$  for GFI to BFI comparison is  $d=1.08$ . According to Cohen [8] this value is interpreted as a large effect.

The second part of the research is the comparison between higher and standard motivation among subjects. The null hypothesis  $H_0$  is assuming that there is no difference between HM and LM groups. The calculation of F-test is shown in table 6 second row.  $H_0$  hypothesis cannot be refuted.

The next part of the experiment was the analysis of feedback information provided to test managers. For each version, the statistics presented to the test manager was calculated and compared with their assessment grade. The comparison of simple estimators is presented in table 7.

Tab. 7, Simple estimators for the opinion of the manager

	Estimator				
	Mean	Geometric mean	Harmonic mean	Harmonic mean rounded	Median
Mean value	4,77	4,53	<b>4,30</b>	4,29	4,69
Mean 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 $r$	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

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 was also tested (test managers have provided answers on a discrete scale).

### E. Interpretation of the results

Results presented in the previous section support the thesis that a user’s knowledge matters in the software quality assessment process. The detailed interpretation of the results is provided in this section below. The floor effect observed

during the experiment seem to be inevitable when evaluators are frustrated with product quality (if they are forced to continue testing even if the application has an unacceptable quality level). The frustration leads to the assignment of the most critical grade without any analysis of its accuracy for the situation.

The first part of the experiment was designed to analyze the problem of perception. Users are unable to verify the technical quality of the applications themselves thus they have to rely on their associations and knowledge. On the other hand people are convinced that objects are unable to change their properties rapidly – this way of perception is used, for example, for observing the movement of objects, although for the quality perception process it influences associations about the product based on previous observations [14].

In the second part of the experiment the motivation effect influence was investigated. In Baron’s experiment [3] the motivation of subjects has affected the results reversing the group effect observed previously by Asch. The experiment presented in this article, has provided no support for the hypothesis that the motivation of subjects influences the software evaluation process. The explanation of this result is based on the professional character of activities undertaken regularly by evaluators. Testers who verify software in large organizations are put under pressure from two sides. If they declare a malfunction then they assume the risk of personal consequences if it was not a malfunction. On the other hand if the malfunction is not noted then they risk even more severe consequences. In such a situation additional motivation is far too weak to have any significant influence on the process.

The experiment regarding the secondary perception has shown that the opinion of the information recipient is worse than the simple average of the analyzed grades. In fact, the most effective estimation, among tested estimators, was made using the harmonic mean of evaluators’ grades. It should be noted that this estimator is the one giving the lowest value among the tested ones.

### III. CONCLUSION

Literature regarding behavioral economics and cognitive psychology presents descriptive models of human cognitive processes. These processes are the base for all judgment formulation processes and decision processes.

Software quality engineering attempts to define the objective measure of software quality. This normative model does not consider the aspects of behaviorism, subjectivism and fallacies which are proven to exist among mental processes. In consequence the normative model will not be able to reflect the real process taking place in the observer’s mind.

A new research area regarding fallacies, cognitive processes etc. within the area of software quality assessment is raising. As the results of the presented experiment have clearly shown, some fallacies strongly affect the perception processes, while the others seem to be negligible in certain circumstances.

This conclusion shows potential benefits resulting from the management of the customers’ perception of the software quality. The application of the results is immediate. For

example, the results of the presented experiment suggest that, despite the lifecycle model, special care should be given to the image of the product. The project management must not allow the delivery of even a single version of a product not compliant with the quality requirements. If such a situation did occur it would cause the customer to be negatively prejudiced about product's quality, which would affect the final acceptance of the product.

The more is known about the cognitive processes associated with the customer's assessment of software quality, the more effective project quality strategies will be able to be build.

## REFERENCES

- [1] Abramowicz W., Hofman R., Suryn W. and Zyskowski D., SQaRE 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.
- [2] Akerlof G. The Market for 'Lemons': Quality Uncertainty and the Market Mechanism, Quarterly Journal of Economics. - 1970. - 84.
- [3] Baron R., Vandello J. and Brunzman B. The forgotten variable in conformity research: Impact of task importance on social influence, Journal of Personality and Social Psychology. - 1996. - 5 : Vol. 71.
- [4] Becker G. Crime and punishment: An Economic Approach, Journal of Political Economy. - 1968.
- [5] Boehm B. et al. Characteristics of software quality - New York : American Elsevier, 1978.
- [6] Boring E. A History of Experimental Psychology - Prentice-Hall, 1950. - Second Edition.
- [7] Camerer C. and Loewenstein G. Behavioral Economics: Past, Present, Future (introduction for Advances in Behavioral Economics) - Mimeo: Carnegie Mellon University, 2003.
- [8] Cohen J. Statistical power analysis for the behavioral sciences - Hillsdale, NJ : Erlbaum, 1988. - Second Edition.
- [9] Hofman R. Software quality perception in Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering, CISSE2008: Springer (in print), 2010.
- [10] Kahneman D. and Tversky A. "Prospect" theory: an analysis of decision under risk, Econometrica. - 1979. - 47.
- [11] Khaleefa O. Who Is the Founder of Psychophysics and Experimental Psychology?, American Journal of Islamic Social Sciences. - 1999. - 16.
- [12] McCall J., Richards P. and Walters G. Factors In software quality, Griffiths Air Force Base, NY, Rome Air Development Center Air Force Systems Command, 1977.
- [13] Mook D. In defense of external invalidity, American Psychologist. - 1983. - 38.
- [14] Nęcka E., Orzechowski J. and Szymura B. Psychologia poznawcza - Warszawa : Wydawnictwo Naukowe PWN, 2008.
- [15] Nobel Foundation Nobelprize.org, All Laureates in Economics. - 07 27, 2009. - [http://nobelprize.org/nobel\\_prizes/economics/laureates/](http://nobelprize.org/nobel_prizes/economics/laureates/)
- [16] Osgood C., Suci G. and Tannenbaum P. The measurement of meaning - Urbana, IL : University of Illinois Press, 1957.
- [17] Shaughnessy J., Zechmeister E. and Zechmeister J. Research Methods in Psychology, McGraw-Hill, 2005. - Seventh edition.
- [18] Simon H. Rational choice and structure of environments, Psychological review. - 1956. - 63.
- [19] Smith A. An Inquiry into the Nature and Causes of the Wealth of Nations - 1776. - Chapter IV: Of the Origin and Use of Money.
- [20] Smith A. The theory of moral sentiments - London, A. Millar, 1759.
- [21] Steenkamp J., Wierenga B. and Meulenbergh M. Kwali-teits-perceptie van voedingsmiddelen deel I. Swoka - Den Haag, 1986.
- [22] Suryn W. and Abran A. ISO/IEC SQaRE. The second generation of standards for software product quality. IASTED2003. - 2003.
- [23] Tversky A. and Kahneman D. Judgement under uncertainty: heuristics and biases in Judgement under uncertainty: heuristics and biases Kahneman D., Tversky A. and Slovic P. (eds) - Cambridge: Cambridge University Press, 1982.
- [24] Tversky A. and Kahneman D. Judgment under Uncertainty: Heuristics and Biases, Science. - 1974. - 185.
- [25] Underwood B. and Shaughnessy J. Experimentation in psychology - New York: Wiley, 1975.