Group effects in software evaluation process

Radosław Hofman[1]
Sygnity Research, Poland

Abstract— *This article analyzes experiment results regarding subjective perception issues caused by the group effect. 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 group effect and its influence on software quality perception.

Index terms—.Software, 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 (1). 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 (1). In this article we present the evidence supporting the validity of the discussed model regarding the area of the influence of an observer's mental state and social context on the quality perception. The observer's mental state, 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 (2) and Boehm (3). 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 (4) and is being used for the decomposition of the end users perspective to software components requirements (5).

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 we conclude that users and customers use some model and method to evaluate software.

The valuation of goods has been studied by economic scientists for centuries (6). 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 (7). 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 (6). The first publication drawing the attention to limitations of the *homo economicus* concept was the author of this idea Adam Smith. In (8) the author describes the asymmetric reaction to the increase and decrease of wealth. This observation was investigated in the 20th century by Daniel Kahneman and Amos Tversky (9).

The economists begrudgingly accepted the counterexam-

[1] 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

ples 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 (10). Psychologists start to compare their psychological models with the economics ones. Daniel Kahneman and Amos Tversky had published the research results for the heuristics in decision making (11) and the prospect theory (9) considered to be the two most important milestones of behavioral economics (6).

The works of Herbert A. Simon (12), Garry S. Becker (13), George A. Akerlof (14), A. Michael Spence, Joseph E. Stiglitz, and Daniel Kahneman (9) 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 (15).

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 19th century near Lipsk (Leipzig) (16). 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 (17). 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 for certain products was proposed by Steenkamp for food quality perception (18). 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. Then one should not only describe investigated phenomenon but also prove the time sequence (cause-effect) and describe the future behavior for the independent variable changes (19).

In the summary of background analysis, described broader in (20), we stress 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 (1). 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 on fig. 1.

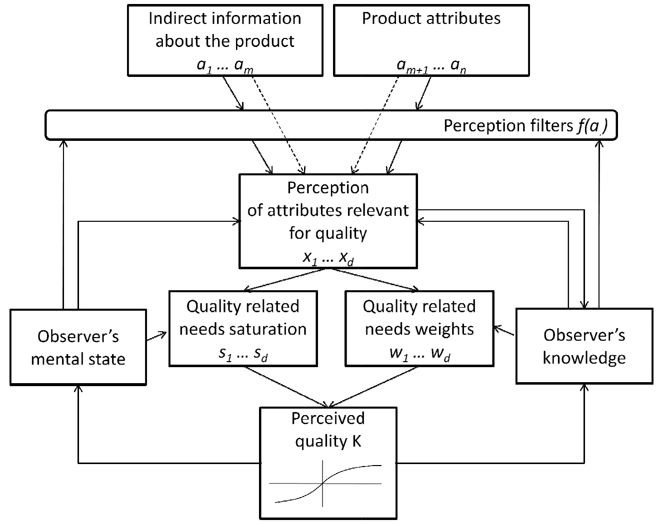

Fig. 1, The software quality perception model

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 we can expect diminishing marginal increases caused by each "quality unit" (Gossen's law). On the other hand if the observed quality is below expectations then we can expect radical dissatisfaction non proportional to the "quality gap" (positive-negative asymmetry) (21).

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 invalid associations etc.). Also both of these structures influence the behavior of the attention filter (perception filter).

The general perceived 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 first: judgments about attributes or the general grade.

II. The Experiment

A. *Description*

In this section we describe the techniques and decisions made for the experiment preparation. 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. The perception may be affected by information seen in media, rumors, previous experience of subjects, infrastructure failures during evaluation and many other. In the above list the most difficult to handle is the problem of differences between IT projects – each project has its own requirements, context of software usage, GUI layout, history of conduction etc. From this perspective it is rather unlikely to have independent IT projects with a controllable list of differences.

The authors have decided to prepare a dedicated, real-like environment for the experiment. To simulate a real project a special application framework has been prepared – TestLab. We will not discuss details of this tool summarizing only 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 applications (called TestApps) with a controllable quality level.

The quality control is designed based on the probability of failure in a TestApp. 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. 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 group effect analysis we have designed an experiment tracing the influence of a group's opinion on an individual's opinion. Subjects were to have assigned tasks in a preset environment and at the end of each task they were to express their individual quality grade. The group was not aware of the final grade expressed by an individual subject, thus this grade may be considered as a subjective opinion.

Subjects were evaluating an internet banking application (TestApp2). Each group was introduced to a task as if it were a real evaluation task. The real goals of the experiment were explained after the experiment's completion. Subjects were given the ability to remove their profiles and all information gathered during the experiment (subjects had randomly assigned numbers and were able to anonymously issue their number on a sheet of paper with the word "disagree").

For the purpose of the experiment, the subjects were randomly divided into three groups located in separate rooms. Each group consisted of subjects and figureheads playing roles of critics with positive or negative attitudes. In one of the groups the figureheads were additionally told to make a disturbance during the evaluation.

In each group the test leader was appointed. Groups had short internal meetings after fixed time periods where each group member was to talk about their impression of the application and accomplished tasks. During these meetings figureheads were expressing positive opinions (in group A) or negative opinions (in group B and C).

The experiment was conducted using a general scenario consisting of two stages. In the first stage subjects completed their profile surveys containing questions about the importance of software quality characteristics (declared quality).

After the evaluation, subjects were assessing the application's quality using an individual survey (an analogical survey as they have used in the profiling stage). Surveys were prepared using Likert-type scales having bipolar terms at the ends, following Osgood's semantic differential (22). Quality characteristics assessed with the survey were: rich functionality, general software quality, compliance with formal rules, efficiency, productivity, satisfaction, learnability, adaptability, reliability, safety and security[2]. The ends of the scale ware 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 was defined as *Application has this quality attribute but not on an exceptional level* [value=6]. The scale was 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 on fig 2.

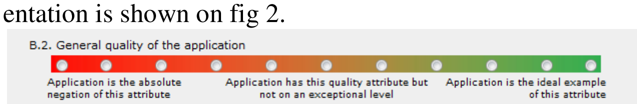

Fig. 2, Scale example for the general quality question

Other instructions and communications were to be identical for all of the groups.

B. Execution

. Subjects were assigned to groups as a result of a randomization procedure. The groups were physically separated. Subjects were Ph.D. seminar participants representing various IT oriented companies. The application TestApp2 was designed to simulate an internet banking system and 100% of the evaluators were users of such applications in the real world. We will present groups equality analysis in section II.C.

The experiment was conducted according to an independent groups plan and the experiment plan was discussed in the previous section.

During the experiment in each group there were three internal meetings where subjects and figureheads loudly expressed their opinions (starting from figureheads). Figureheads were addressing issues regarding general quality not any specific quality characteristic. While submitting the final quality grade to the management system there were no interactions inside groups.

During the experiment one of the subjects from group A declared that she had to leave after couple of minutes. Her answers in the first survey and in the evaluation stage were excluded from the analysis.

C. Validity analysis

To assess validity we have to discuss the internal validity first. The first question regards the quality of the samples. The samples were a result of a randomization procedure and all of the subjects have declared being users of internet banking applications. There are no clues to reject the as-

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

sumption that the groups were equivalent. To analyze the strength of the observed effect we have to show that the groups were equivalent. We will now present test for the homogeneity of the groups.

The test is based on the null hypothesis H_0 (we are testing three equalities in three tests): $M_A=M_B$, $M_A=M_C$ and $M_B=M_C$. Groups' means are based on the declared quality importance in the first survey. The declared importance is presented in table 1.

A	B	C
9	9	10
10	10	11
10	10	11
10	11	10
6	9	11
na*		
9		

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

For these four groups we apply the ANOVA method: the F-test (19), to compare inter group differences. We will use the method for the two-groups comparison having analysis analogical to Student's test for two groups (22).

We have decided to exclude one of the subjects from the analysis (marked with asterisk). This was the subject who had not finished the evaluation. She had not completed the first survey and assigned the lowest mark for each quality characteristic after the unfinished evaluation.

Homogeneity of groups is calculated using an ANOVA table as shown in table 2. We analyze homogeneity in pairs.

M_A	M_B	M_C	SS	SSE	F	p	F_{crit} (5%)	*d*
8,00	9,80		9,45	56,8	1,66	11%	4,96	
8,00		10,6	19,7	55,2	3,57	23%	4,96	
	9,80	10,6	1,60	4,00	3,20	11%	5,32	

Tab. 2, F-test for the groups homogeneity

In each test we have no statistical clue to reject the null hypothesis on the preset confidence level ($\alpha=5\%$).

Homogeneity of the groups was also tested by the analysis of the safety assessment from the evaluation phase. The safety of the application is un-testable for standard users, thus the assessed quality level is a result from subjects' background confidence, not the evaluation itself. All of the subjects declared themselves as internet banking users which means that their personal confidence in internet banking applications should be above average. in the pairwise ANOVA analysis of null hypothesis assuming the means to be equal is presented in table 3.

M_A	M_B	M_C	SS	SSE	F	p	F_{crit} (5%)	*d*
4,33	3,80		0,78	52,1	0,13	72%	5,12	
4,33		2,80	6,41	44,1	1,31	28%	5,12	
	3,80	2,80	2,50	33,6	0,60	46%	5,32	

Tab. 3, Safety of the application assessment

As presented above we have no reasons to reject the null hypothesis for all of the homogeneity tests, on the preset confidence level. The analysis shows then that we should assume the groups were equivalent.

Confounding threads were mitigated by deception of the subjects. The invitation for the meeting and information given at the beginning of the evaluation described the task as an evaluation of a new technology prototype. These circumstances comply with typical modern behavioral economics research (23).

The external validity is a general measure of the likelihood that the observed reaction will take place in the future. According to Mook (24) if one is testing a theory based on psychological studies then external validity is not of key importance (19). This observation uses a corollary that behavior patterns are rather constant even in different situations (23). In the case of the discussed experiment the research is focusing on psychological theories. Although the statistical significance of the results is presented in section II.D.

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 (19).

D. Results analysis

The results of the experiments will be analyzed with the F test and ANOVA method (19). Below we will present a selected characteristics analysis (omitting those where no effects were observed). The analysis of the data shows that during the experiment the floor effect could have taken place. Frustrated subjects expressed their absolute negation by assigning the lowest possible value for the quality. Thus we may not observe proper differences between groups B and C. In our opinion the survey-based approach has its limits in the measurement of the attitude of a frustrated subject, as they will always select the lowest possible grade. An approach to such analysis could employ neuro-analysis showing differences in the intensity of emotions (26).

For each analysis we assume the null hypothesis $\mathbf{H_0}$ about the means equality between pairs: $M_A=M_B$, $M_A=M_C$ and $M_B=M_C$. The preset confidence level is $\alpha=5\%$. In the following tables we present the probability of an observation for the assumption of the null hypothesis being true. We also present Cohen's *d* factor for the effect strength analysis (where an effect is observed).

a) The general quality

The general quality assessment analysis is presented in table 4.

M_A	M_B	M_C	SS	SSE	F	p	F_{crit} (5%)	*d*
6,50	2,20		50,4	32,3	141	0%	5,18	1,50
6,50		1,80	60,2	32,3	16,8	0%	5,18	1,54
	2,20	1,80	0,40	13,6	0,24	64%	5,32	0,32

Tab. 4, F-test for general quality assessment

We reject the null hypothesis for the assumptions about the equality of $M_A=M_B$ and $M_A=M_C$. Cohen's *d* used for the effect strength estimation predicts a large effect strength when $d>0.8$ (23). In the analyzed case *d* exceeds 1.5, thus we conclude that the effect strength is large.

We have noted in the previous section that dissatisfied users had expressed the maximum negative grade. Due to this we could not observe any statistical difference between groups B and C.

b) Productivity

Software productivity assessment analysis is presented in table 5.

M_A	M_B	M_C	SS	SSE	F	p	F_{crit} (5%)	*d*
7,67	4,40		29,1	28,5	9,18	1%	5,12	1,36
7,67		5,20	16,6	54,1	2,76	13%	5,12	0,93
	4,40	5,20	1,60	60,0	0,21	66%	5,32	0,31

Tab. 5, F-test for software productivity assessment

We are able to reject the null hypothesis only for the assumptions about the equality of $M_A=M_B$. The difference between groups A and C consists of two independent influences: in group C figureheads were expressing negative opinions about the software and making a disturbance during the evaluation task. The productivity assessment is the grade of how fast the accomplishment of tasks was done. In other words the measure is converse to the awareness of the time spent on the task (a user builds their feeling about productivity having in mind their list of tasks to be done and the feeling of waiting for the application).

The effect size of the difference between M_A and M_B is large.

c) Satisfaction

The users' satisfaction assessment analysis is presented in table 6.

M_A	M_B	M_C	SS	SSE	F	p	F_{crit} (5%)	*d*
6,50	2,00		55,2	23,5	21,1	0%	5,12	1,60
6,50		4,20	14,4	52,3	2,48	15%	5,12	0,89
	2,00	4,20	12,1	36,8	2,63	14%	5,32	0,94

Tab. 6, F-test for users' satisfaction assessment

The users' satisfaction assessment corresponds to the previous characteristic. We reject the null hypothesis only for the assumption that $M_A=M_B$. The effect size is large.

d) Reliability

The reliability assessment analysis is presented in table 7.

M_A	M_B	M_C	SS	SSE	F	p	F_{crit} (5%)	*d*
4,40	1,60		19,6	18,4	8,52	2%	5,32	1,36
4,40		1,80	16,9	28,0	4,83	6%	5,32	1,16
	1,60	1,80	0,10	16,0	0,05	83%	5,32	0,15

Tab. 7, F-test for reliability assessment

Reliability assessment corresponds to the productivity and users' satisfaction assessment. We reject the null hypothesis only for the assumption that $M_A=M_B$. The effect size is large.

Although the first analysis prevents us from rejecting the null hypothesis for the assumptions that $M_A=M_C$ we have observed, that it is unclear what the decision should be. In group C the reliability was assessed by all but one user at the level 1 (the lowest grade). The other subject has assessed reliability at the level 5. The log analysis shows that the application was clicked by this subject only 36 times experiencing only 5 errors. If this observation was rejected the $M_A=M_C$ assumption has to be rejected (removal of this subject from other analysis does not change the general conclusions). Table 8 presents the reliability analysis after one outstanding observation was removed.

M_A	M_B	M_C	SS	SSE	F	p	F_{crit} (5%)	*d*
4,40	1,60		19,6	18,4	8,52	2%	5,32	1,36
4,40		1,00	25,7	15,2	11,8	1%	5,59	1,50
	1,60	1,00	0,80	3,20	1,75	23%	5,59	0,85

Tab. 8, F-test for reliability assessment after removal of one observation

E. Interpretation of the results

Results presented in the previous section support the thesis that a user's state of mind matters in the software quality assessment process. The detailed interpretation of the results is provided in the section below. For the floor effect we conclude that when evaluators are frustrated with product quality (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 general quality assessment grade test was designed to verify the influence of suggestions onto individual assessment. It is important to notice, that despite of opinions expressed loudly by the subjects, their final assessment was hidden from the rest of the group. The interpretation is using observations known from cognitive psychology research – people tend to stand for their own opinions, thus publicly expressed opinion is a kind of boundary for the private opinion (11).

The null hypothesis for the second independent variable (disturbing factor during the evaluation) could not be rejected on pre-set confidence level. The main problem is the floor effect caused by the frustration of subjects discussed at the beginning of this section. Inability to reject the null hypothesis should not be interpreted as a proof for effect not taking place during the evaluation (see analysis of certain characteristics).

The productivity and satisfaction analysis show the same general effect caused by an influence of authorities inside the evaluation team. An interesting observation is associated with the disappearance of this effect in third group where disturbing factor was set (the effect with large size occurred for the general quality assessment). An interpretation of this observation regards the character of disturbing factor: subjects were involved in off-topic, pleasant discussions. Their feelings about time spent on tasks was then significantly lower than in other groups. The same change was taking place in the area of personal satisfaction which was on a higher level due to pleasant time spent on discussions. Both mechanisms resulted in higher assessment of these characteristics. It seems that negative influence from the authorities was balanced with the subjects feelings.

The reliability analysis show that the effect was observed for both groups. One of the observations has made the effect violating slightly confidence level, but as there are clues to reject this observation as an error (this subject assessed quality on the level near to group's mean in all other categories) we conclude that the effect exist for both groups. Analogically to the general quality analysis we could not observe effect resulting from the second independent variable (disturbing factor) caused by the floor effect. In this point we conclude that disturbing factor could not compensate negative context built by figureheads because of strong associations with fatal crashes of software. Experienced users have reliability related associations with a "blue screen" thus when they have seen such a screen couple of times during relatively short time of evaluation they have regarded a low reliability of the application. This connotation was strengthen by the negative comments from the figureheads. This result may be also regarded as an example of exploitive bias, where people tend to pay more attention to more spectacular occurrences (crash with a "blue screen").

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.

This new research area regarding fallacies, cognitive processes etc. in the area of software quality assessment is being defined. 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 suggests that authoritative figures may affect personal judgments about software quality. Also we conclude that the escalation of emotions leads to frustration and negative attitude thus a proper customer's emotion management should be applied. If authoritative figures criticize the system or evaluators emotions fall beyond control it will cause the customer to be negatively prejudiced about product's quality, which would affect the final acceptance of the product.

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

IV. References

1. **Hofman, R.** Software quality perception. [book auth.] CISSE2008. *Innovations and Advanced Techniques in Systems, Computing Sciences and Software Engineering.* s.l. : Springer (w przygotowaniu), 2009.
2. **McCall, J., Richards, P. and Walters, G.** *Factors In software quality.* s.l. : Griffiths Air Force Base, NY, Rome Air Development Center Air Force Systems Command, 1977.
3. **Boehm, B., et al.** *Characteristics of software quality.* New York : American Elsevier, 1978.
4. *ISO/IEC SQuaRE. The seconod generation of standards for software product quality.* **Suryn, W. and Abran, A.** 2003. IASTED2003.
5. *SQuaRE based Web Services Quality Model.* **Abramowicz, W., et al.** Hong Kong : International Association of Engineers, 2008. International Conference on Internet Computing and Web Services. ISBN: 978-988-98671-8-8.
6. **Camerer, C. and Loewenstein, G.** *Behavioral Economics: Past, Present, Future (introduction for Advances in Behavioral Economics).* Mimeo : Carnegie Mellon University, 2003.
7. **Smith, A.** *An Inquiry into the Nature and Causes of the Wealth of Nations.* 1776. Chapter IV: Of the Origin and Use of Money.
8. —. *The theory of moral sentiments.* London : A. Millar , 1759.
9. *"Prospect" theory: an analysis of decision under risk.* **Kahneman, D. and Tversky, A.** 47, 1979, Econometrica.
10. **Nęcka, E., Orzechowski, J. and Szymura, B.** *Psychologia poznawcza.* Warszawa : Wydawnictwo Naukowe PWN, 2008.
11. **Tversky, A. and Kahneman, D.** Judgment under Uncertainty: Heuristics and Biases. *Science.* 1974, 185.
12. *Rational choice and structure of environments.* **Simon, H.** 63, 1956, Psychological review.
13. **Becker, G.** Crime and punishment: An Economic Approach. *Journal of Political Economy.* 1968.
14. **Akerlof, G.** The Market for 'Lemons': Quality Uncertainty and the Market Mechanism. *Quarterly Journal of Economics.* 1970, 84.
15. **Nobel Foundation.** Nobelprize.org. *All Laureates in Economics.* [Online] [Cited: 07 27, 2009.] http://nobelprize.org/nobel_prizes/economics/laureates/.
16. **Boring, E.** *A History of Experimental Psychology.* Second Edition. s.l. : Prentice-Hall, 1950.
17. *Who Is the Founder of Psychophysics and Experimental Psychology?* **Khaleefa, O.** 16, 1999, American Journal of Islamic Social Sciences.
18. **Steenkamp, J., Wierenga, B. and Meulenberg, M.** *Kwali-teits-perceptie van voedingsmiddelen deel 1. Swoka.* Den Haag : s.n., 1986.
19. **Shaughnessy, J., Zechmeister, E. and Zechmeister, J.** *Research Methods in Psychology.* Seventh edition. s.l. : McGraw-Hill, 2005.
20. **Tversky, A. and Kahneman, D.** Judgement under uncertainty: heuristics and biases. [book auth.] D. Kahneman, A. Tversky and P. Slovic. *Judgement under uncertainty: heuristics and biases.* Cambridge : Cambridge University Press, 1982.
21. **Osgood, C., Suci, G. and Tannenbaum, P.** *The measurement of meaning.* Urbana, IL : University of Illinois Press, 1957.
22. **Cohen, J.** *Statistical power analysis for the behavioral sciences.* Second Edition. Hillsdale, NJ : Erlbaum, 1988.
23. **Underwood, B. and Shaughnessy, J.** *Experimentation in psychology.* New York : Wiley, 1975.
24. *In defense of external invalidity.* **Mook, D.** 38, 1983, American Psychologist.
25. *The forgotten variable in conformity research: Impact of task importance on social influence.* **Baron, R., Vandello, J. and Brunsman, B.** 5, 1996, Journal of Personality and Social Psychology, Vol. 71.