

Usability Evaluation of Object Oriented Software System using Fuzzy Logic Approach

Sanjay Kumar Dubey
Amity University
Sec-125, NOIDA, India

Ajay Rana
Amity University
Sec-125, NOIDA, India

Arun Sharma
KIET, Ghaziabad
(U.P.), India

ABSTRACT

The growth in demand for interactive software system has increased greatly in recent years. But, most of the developed systems are failing due to not providing suitable interface. User interface is the only way by which user can interact with software system. The problem lying in the interface is related to the usability. Usability is regarded as important quality factor for developing the successful interactive software system. It is also a key quality factor in the development of successful software applications. These days mostly software systems are developed using object-oriented methodology. Object-oriented approach enhances the usability of software system when software engineering process combined with usability engineering. Incorporating object-oriented concepts and techniques into system development processes, systems related to human computer interaction are more usable. In spite of the importance of usability, there is no well defined criteria to evaluate it due the fact that many factors influence the usability of software system. This paper identifies the most important factors that impact on usability of object-oriented system and then proposes a model for evaluating the usability of software system using soft computing technique.

General Terms

Software Engineering

Keywords

Usability, Software, System, Model, Soft computing

1. INTRODUCTION

The demand for interactive software system is increasing in exponential way by users. But several systems do not serve the specific needs of different users. After getting the system users don't want to devote much effort in learning to use them. They only want easily to interact with that system. If systems are difficult in use then, in spite of large investment on their design, implementation and modification, they are not accepted by the users. User interface plays very important role in the success/failure of interactive software system. Some empirical studies have identified, that about 50% to 80% of all source code is concerned with the user interface [3].

The field of evaluation of user interface is also known as usability evaluation and it has become increasingly important with computer users [8]. Usability is defined as "the ease with which a user can learn to operate, prepare inputs for, and interpret outputs of a system or component" [12]. Research shows that usability is key component in the overall quality of a software product [24]. Usability also became the key issue in Human-Computer Interaction (HCI) because not only it provides descriptive background and guidelines for development of a quality user interface but it is also concerned with supporting users during their interactions with computers [6]. The major reason for the failure of a system is

unusability [4]. According to Lu LIANG [21], usability problem indicates the potential problem lying in interface design, operating process, or product structure, which results in the difficulty of use for end users. There are various models which incorporate the usability quality factor in wider way and described the impact of usability on system. In spite of best efforts of usability experts, systems continue to have usability flaws that are due to lacking of measuring or evaluating methods of usability. According to E. Chang and T. S Dillon [9], the literature has provided different approaches for usability evaluation which are:

- a) Empirical testing [29]
- b) Inspection [16]
- c) Comparative usability measures [2]
- d) Formal complexity-based measures [10]
- e) Metrics for usability standards in computing [25]

So as to obtain a higher usability level in software product we need to apply the set of technique belonging to usability engineering and object-oriented methods offers bridge between usability engineering and software engineering [34]. According to Xavier Ferre [34], object-oriented software development can be combined with usability technique and among software development techniques, object-oriented approaches are the closest to usability perspective. Usability related activities are embedded in the object-oriented methods/approach by using use-case. According to R. Agarwal et al. [28], paying close attention to the tasks of object-oriented is best suited for rather than assuming it's the best option every time- may be one way to maximize the usability of system. They described that usability of object-oriented approach assumes greater significance. Object-oriented representation increases the usability of software system because they are easy to learn and understand. These learning will not help only users but also help to naive developers.

Research studies determined that many factors influence the usability of the software and these factors contribute into making user performance, user preference, and user interface [29, 2, 25, 15]. To get quality user interface i.e. usability, there is need to characterizing each factors of usability that impact on it individually and then after coalesce them in such a manner that we acquire overall usability of system. But each one of these factors is itself an imprecisely defined concept and has fuzzy aspects. Usability can also not be readily quantified because it within itself has a number of fuzzy aspects. Therefore, we choose to evaluate usability by using soft computing techniques as the key focus areas of soft computing techniques are fuzzy logic, neural computing, evolutionary computing, machine learning and probabilistic reasoning. Soft computing concept was introduced by L.

Zadeh [19]. Soft computing approach suggests the utilization of the given tolerance of imprecision, partial truth, uncertainty and approximation for a particular problem to achieve tractability robustness and low solution cost. Khoshgoftaar [32] established that if we have incomplete information in the beginning phases of software development, a software quality model having fuzzy characteristics will perform better as fuzzy deals with vagueness and imprecision. According to E. Chang and T. S. Dillon [9], quantifying usability is a difficult because it has several dimensions and several characteristics seem to impact upon it. Putting all dimensions and characteristics together to obtain usability is a very difficult issue, because each of them involves linguistic terms and fuzzy concepts. Since the past few years have witness of growing recognition of soft computing technology for intelligent systems, this paper has chosen fuzzy mechanism for characterizing each of the considered individual sub-characteristics of usability for object-oriented system viz. effectiveness, efficiency, satisfaction and learnability and also for assessing usability itself.

2. MATERIALS AND METHODS

All material in the categorizations and definitions, ISO 9126 [13] and J. Nielsen [16] mainly focused on the sub-characteristics that constitute usability while other definitions [14, 26] put emphasis on how usability should be measured. This paper focuses on both the approach. To finalize the main sub-characteristics on which usability depends, we conducted a survey on experts of the domain. The experts were software engineers from different IT companies and Ph. D. students who have good knowledge of the system. From the collected data and by thoroughly study of standards and models [31], the four sub-characteristics namely effectiveness, efficiency, satisfaction and learnability were found as the main contributors in the assessment of usability of the software system.

Out of these four sub-characteristics, the three sub-characteristics viz. effectiveness, efficiency and satisfaction are also adopted as the main dimensions of usability in ISO 9241-11 [14] standard. According to this standard, usability is a three-dimensional quality of software product, defined as “the effectiveness, efficiency and satisfaction with which specified users can achieve specified goals in particular environments”. In this standard, “efficiency” is defined as the resources consumed so as to achieve accuracy and wholeness in accomplishment of users’ objectives, “effectiveness” as the accuracy and wholeness with which goals are achieved in a particular system and “satisfaction” as a subjective measure which concerns the comfort in use and acceptance of the software by the users. Bevan and Macleod [25] also suggested that usability measure is better reflected into effectiveness, efficiency and satisfaction. Satisfaction always relies on subjective judgment, but on some degree, the measurement of effectiveness and efficiency is able to be objectively calculated in some ways [21].

The fourth sub-characteristics i.e. learnability is considered in the study based on the expert’s opinion. The survey [31] also shows that learnability has greater impact among all the other sub-characteristics on the user-interface (usability) of the software system. As per experts, remaining sub-characteristics described in survey [31] play negligible role. So, those sub-characteristics discarded from study. According to E. Chang and T. S. Dillon [9], when one is carrying out the development of the user interface utilizing a particular design methodology, it is difficult to evaluate usability characteristics that the user interface implies on the users.

2.1 Effectiveness

Effectiveness is defined as “the capability of the software product to enable users to achieve specified goals with accuracy and completeness in a specified context of use” [11]. Wang Changqing *et al.* [33] objectively defined that “effectiveness” indicates whether the system is able to support user working in an effective way, namely, whether the user can carry out tasks by least steps in least time. According to them, it is computed by the taking the ratio of “the number of steps in the most effective path of a task” to “the number of steps experienced in practice”. The most common measure of effectiveness taken by usability practitioners is:

- the completion rate;
- the percentage of users who successfully complete the tasks;
- amount of the tasks completed successfully;
- number of errors;
- percentage of relevant functions used;
- percentage of task completed; and
- percentage number of errors.

2.2 Efficiency

Efficiency is “the capability of the software product to provide appropriate performance, relative to the amount of resources used, under stated conditions” [11]. According to Software Usability Measurement Inventory (SUMI) [24] “efficiency” is a measure of the user’s perspicacity of chronological expertise and intellectual workload caused by the interaction. Wang Changqing *et al.* [33] objectively defined efficiency as: “efficiency” indicates whether the system can achieve the objectives of users. According to them, it is calculated by ratio of “the number of finished tasks in an evaluation” to “the number of sample tasks in an evaluation”. E. Chang and T. S. Dillon [9] described efficiency as a quality of the user interface, which characterizes how efficiently the user can complete his task. They defined efficiency in number of components as:

- number of goals/task not achieved;
- time taken for task completion;
- unproductive period; and
- percentage of task not completed.

J. Nielsen [16] described efficiency in form that if users once have learned the design, then they can swiftly perform tasks. Nielsen, proposed that the system should be proficient and valuable to employ, so that after proper learning of the system design by the user, productivity is achieved up to a great extent.

2.3 Satisfaction

ISO 9126-I [11] defined satisfaction as “the capability of the software product to satisfy users in specified context of use”. Satisfaction also refers to users’ subjective assessment of the system concerning how pleasant it is to use [16]. Questionnaires [24] are developed for satisfaction measurement that can be specified on a subjective rating scale such as:

- discomfort practiced;
- fondness for the product;

- pleasure with product use;
- acceptability of the workload when carrying out diverse tasks; and
- the extent to which particular usability objectives have been met.

In SUMI (Software Usability Measurement Inventory), additional measures of satisfaction might consist of the number of positive and negative comments recorded during use [24]. J. Nielsen [16], and A. Seffah *et al.* [4] described satisfaction in a broader meaning by considering other subjective terms such as, for example, the absence of discomfort when using the system and the ability of the system to fulfill the goal of the user.

2.4 Learn ability

Learnability is the capability of the software product to facilitate the user to learn its application [13]. A learnability measure should be able to assess the time and effort required by system developers to learn how to use particular functions (interfaces, operations, etc.), or the effectiveness of the documentation (manuals, help system, demos). For example, the user documentation and the help system should be thorough; it should be complete, precise and must solve all of the user's concerns so as to how to accomplish common tasks. B. Shackel [7] described that learnability is the relation of performance to training and frequency of use, i.e. the novice user's learning time with specified training and retention on the part of casual users. According to J. Nielsen [16] learnability refers to the novices' ability to reach a reasonable level of performance rapidly. Nielsen considered learnability to be an important criterion because a complex and a new system have to be learnt for the efficient results. According to A. Anthony [1], "learnability requires attention to the needs of the novice and uninitiated users. The uninitiated user is one that has no previous experience with the software or similar software. Novice user has either had some experience with similar software or has limited experience with the software".

3. Fuzzy Logic Approach to Usability Model

Fuzzy logic is a mathematical tool which provides an easy way to reach your goal at a specific conclusion based upon vague, indefinite, inaccurate, or mislaid input information [18]. Fuzzy logic also offers a particular convenient way to produce a mapping between input and output spaces by using natural expression [20]. Major advantage of fuzzy logic approach is that the fuzzy logic models can be constructed with no data or with little data [17, 30]. Additionally, fuzzy logic models can be accustomed to new environment when data become available [23]. Fuzzy logic is based on if then rules, which are designed by considering the opinion of experts from the domain. Expert-based estimation was found to be better than all regression-based models [27]. According to Lather *et al.* [5], fuzzy model is best choice for managing ambiguous, doubtful, contradicting and divergent opinions and it is all the way a better choice when complexity and non-linearity are very high. L. Lin *et al.* [22] presented a new assessment method to obtain the integrated software quality for evaluating user satisfaction by using the fuzzy set theory based on the ISO 9126 quality model with a single evaluator. D. Gupta *et al.* [8] has provided a case study of different software quality estimation techniques to build a software quality model. Their paper suggests that the fuzzy and rule based system techniques are better for designing and

evaluating a software quality model. Based on above context, Fig. 1 shows the proposed fuzzy model to evaluate usability.

3.1 Working of Fuzzy Model

a) Fuzzification: To determine the degree to which the input data match the condition of the fuzzy rules.

b) Inference: Fuzzy inference system is the method of formulating the mapping from a given input to an output using fuzzy logic. It calculates the rule's conclusion based on its matching degree.

c) Composition: To combine the conclusion inferred by all fuzzy rules into a final conclusion.

d) Defuzzification: To convert a fuzzy conclusion into a crisp one. The input for the defuzzification process is a fuzzy set and the output is a single number.

There are total 11 membership functions available in MatLab. For fuzzification we considered Triangular Membership Functions (TMF), because of its simplicity and heavy use by researchers for prediction models. It is a three-point function, defined by minimum α , maximum β and modal value m i.e. TMF (α, m, β) where ($\alpha \leq m \leq \beta$). The domain of membership function is taken in the closed interval $[0, 1]$.

4. MEMBERSHIP FUNCTIONS FOR INPUT PARAMETERS

To evaluate usability, we have four main linguistic variables i.e. Effectiveness (EFFEC), Efficiency (EFFIC), Satisfaction (SATIS) and Learnability (LEARN). All these variables are different in nature and contribution in the usability of software system. To get usability, we have used fuzzy logic and defined the linguistic values of these variables into Low, Medium and High categories. To fuzzify, the intervals for these variables are normalized within the range $[0, 1]$ by dividing the base numerical values by the corresponding maximum magnitudes. The fuzzifications of input variables are shown in Fig. 2 (EFFEC), Fig. 3 (EFFIC), Fig. 4 (SATIS) and Fig 5. (LEARN).

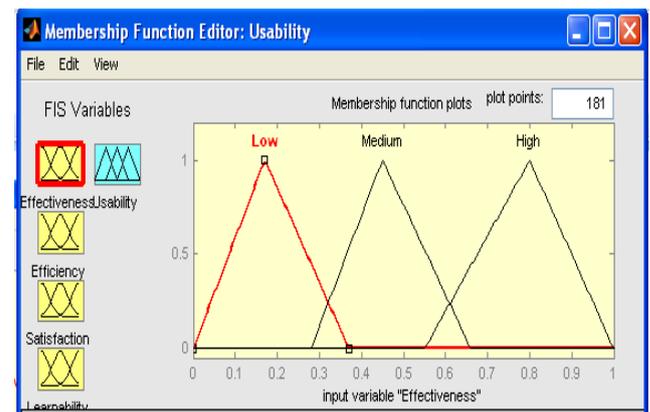


Fig. 2: Membership function for effectiveness

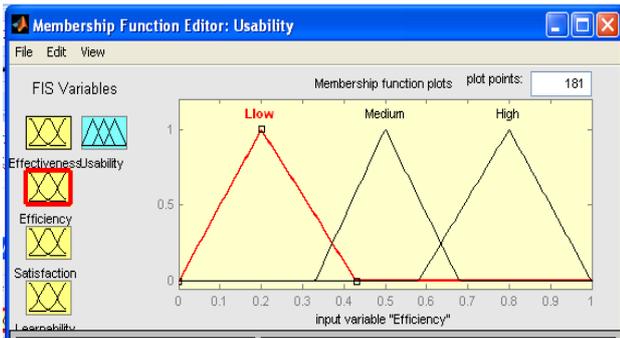


Fig. 3: Membership function for efficiency

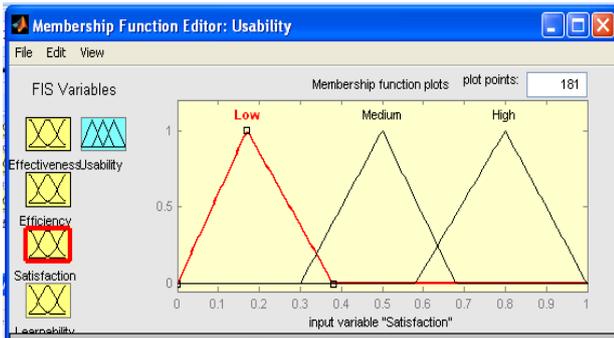


Fig. 4: Membership function for satisfaction

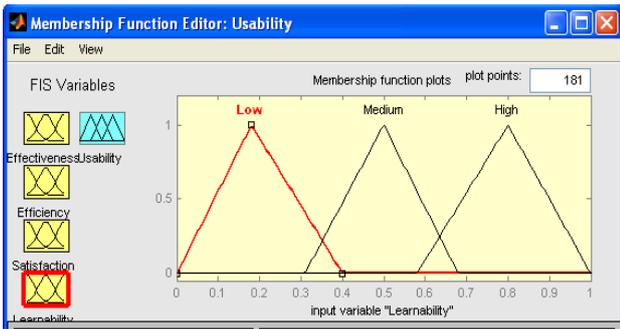


Fig. 5: Membership function for learnability

The output variable usability is classified as Very Low, Low, Medium, High and Very High. Figure 6 represents member function for output variable usability

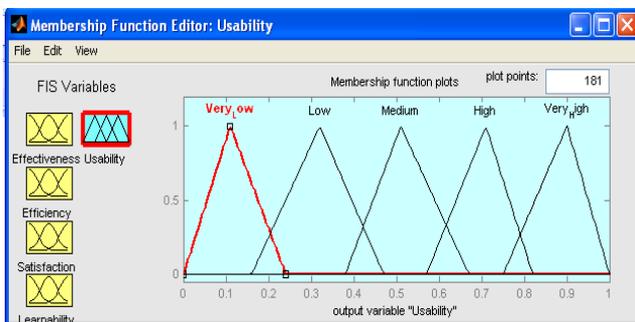


Fig. 6: Membership function for usability

5. KNOWLEDGBASE AND EVALUATION PROCESS

As soon as input data is fuzzified, processing is carried out in fuzzy domain. The model integrates the effect of EFFEC,

EFFIC, SATIS and LEARN into single predictable parameter i.e. software usability, which is based on knowledge base. In the proposed fuzzy model, we are considering four inputs, each consisting of three terms, therefore our knowledge base consists 81 rules after considering all the possible combinations of inputs. Suppose if a fuzzy model is having A terms associated with B inputs, then possible number of rules (say R) for this model can be calculated by considering the cartesian product of all input states.

$$R = A * A * A \dots * B \text{ times}$$

$$R = AB$$

For our model $R = 3 * 3 = 81$

In this paper, Mamdani method is used for defining fuzzy rule due to its simplicity and widely used in research applications, which is used for nonlinear equations. These rules are designed on the basis of experience and expertise knowledge of the field that's why this is called knowledge base. Some of rules are listed in Table 2.

Table 2: Sample Rules

RuleNo.	Input Variables				Output Variables
	Effective-ness	Efficiency	Satis-faction	Learn-ability	
1	Low	Low	Low	Low	Very Low
2	Low	Low	Low	Medium	Low
3	Low	Low	Low	High	Low
.					
.					
.					
81	High	High	High	High	Very High

Using proposed methodology and model, usability of the software system can be assessed. The usability value will be between 0 and 1 and will be in any of the ranges, Very Low, Low, Medium, High and Very High. With the help of this value, we can specify empirical usability level of software system. Rule viewer is shown in Fig. 7. In this figure, Effectiveness = .491, Efficiency = .609, and Satisfaction = .649 and Learnability = .634. Using these inputs and defined fuzzy rules, usability has been evaluated as .634, which is very high.

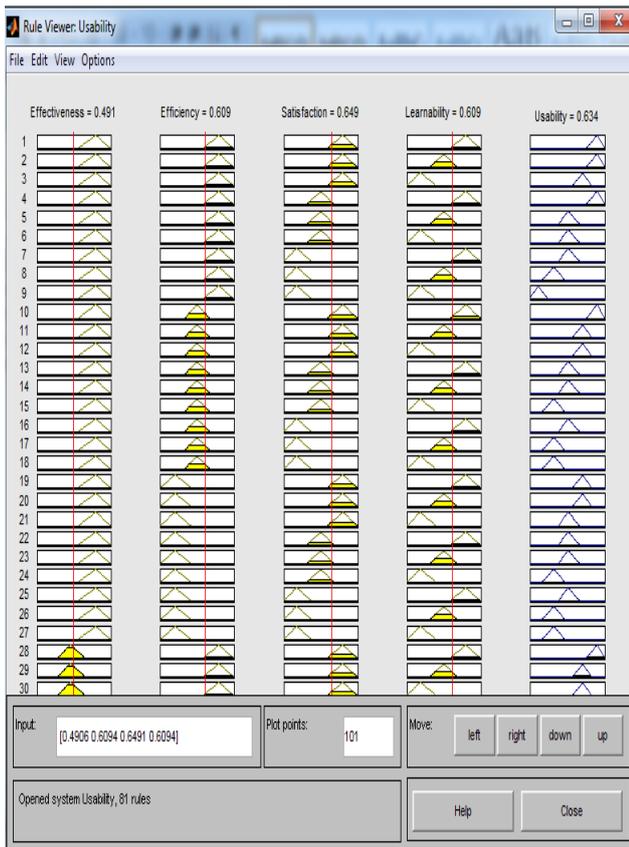


Fig. 7: Rule-Base

6. CONCLUSION

This paper proposed a fuzzy model to determine the usability of object-oriented software system. In this model, input variables are the usability sub-characteristics viz. effectiveness, efficiency, satisfaction, learnability, on which usability of software depends. The sub-characteristics have been determined through extensive survey conducted on the experts of specified domains and were normalized on 0-1 scale. Basis on expert's knowledge and experience, rule base is generated with 81 rules for evaluating usability of object-oriented system. The usability of object-oriented system can be improve by considering the defined characteristics and for this purpose, the developed fuzzy model will help the researchers, usability practitioners and software developers to select best usable software system.

7. REFERENCES

[1] A. Anthony, Software: a fine art, Jan 2004.
 [2] A. Dillon, and M. Maquire, "Usability measurement—Its practical value to the computer industry," in *Proc. ACM/IFIP Human Factors Computer System (INTERCHI)*, Amsterdam, The Netherlands, 1993, pp. 145–148.
 [3] A. M. Brad, and M. B. Rosson, "Survey on user interface programming", Proceedings of the SIGCHI conference on Human factors in computing systems, California, USA, 1992, pp. 195-202.
 [4] A. Seffah, M. Donyae, R. B. Kline, and H. K. Padda: Usability measurement and metrics: A consolidated model, *Software Quality Control*, Vol. 14, No. 2, 2006, pp. 159–178.

[5] A. S. Lather, S. Kumar, Y. Singh, "suitability assessment of software developers: a fuzzy approach", *ACM SIGSOFT, SEN*, Vol. 25 No.3 2000 pp.30-31
 [6] B. Schneiderman, *Designing the User Interface: Strategies for Effective Human Computer Interaction* (2nd ed.), 1992, Reading, Addison-Wesley, MA.
 [7] B. Shackel, Usability—Context, framework, definition, design and evaluation, in B. Shackel and S. Richardson (Eds.), *Human Factors for Informatics Usability*, Cambridge, MA: University Press, 1991, pp. 21–38.
 [8] D. Gupta, H. K. Mittal, and V. Goyal, and "Comparative Study of Soft Computing Techniques for Software Quality Model," *International Journal of Software Engineering Research & Practices*, Vol.1, No.1, Jan, 2011, pp.33-37.
 [9] E. Chang, and T. S. Dillon, "A Usability-Evaluation Metric Based on a Soft-Computing Approach", *IEEE transaction on Systems, man, and Cybernetics—Part A Systems and Humans*, Vol. 36, No. 2, March 2006.
 [10] H. Thimbleby, "Formulating usability," *SIGCHI Bull.*, Vol. 26, No. 2, Apr. 1994, pp. 59–64.
 [11] International Standard. ISO/IEC 9126-1. *Institute of Electrical and Electronics Engineers*, Part 1,2,3: Quality model, 2001.
 [12] IEEE Std. 1061, IEEE standard for a software quality metrics methodology, New York, IEEE Computer Society Press, 1992.
 [13] ISO 9126, "Information Technology-Software Product Evaluation-Quality Characteristics and Guidelines for their Use", Geneva, 1991.
 [14] ISO 9241-11, "Ergonomic requirements for office work with visual display terminals (VDTs) Part 11: Guidance on usability", 1998.
 [15] J. Levy, and J. Nielsen, "Measuring usability preference vs. performance," *Commun. ACM*, Vol. 37, No. 4, Apr. 1994, pp. 66–75.
 [16] J. Nielsen, *Usability Engineering*, Academic Press, 1993.
 [17] J. Ryder, *Fuzzy Modeling of Software Effort Prediction*. In: *Proceeding of IEEE Information Technology Conference*, Syracuse, New York, 1998, pp. 53-56.
 [18] L. A. Zadeh, *Fuzzy Logic, Neural Networks, and Soft Computing*. *Communication of ACM*. 37(3), 1994, pp. 77-84.
 [19] L. A. Zadeh, "Fuzzy sets, *Information and Control*", Vol. 8, 1965, pp. 338-353.
 [20] L. A. Zadeh, From Computing with numbers to computing with words—from manipulation of measurements to manipulation of perceptions. *International Journal of Applied Mathematics and Computer Science* 12(3), 2002, pp. 307-324.
 [21] L. Liang, X. Deng , and Y. Wang , "Usability Measurement Using a Fuzzy Simulation Approach", *International Conference on Computer Modeling and Simulation*, DOI 10.1109/ICCMS, 2009, pp. 76
 [22] L. Lin, and H. M. Lee, "A Fuzzy Software Quality Assessment Model to Evaluate User Satisfaction," *Proceedings of the Second International Conference on Innovative Computing, Information and Control*, Washington DC, USA, September, 2007, pp.438-442.

- [23] M. O. Sailu, M. Ahmed, and J. AlGhamdi, Towards Adaptive Soft Computing Based Software Effort Prediction. In: Fuzzy Information, Processing NAFIPS, 2004, pp. 16-21.
- [24] M. Porteous, J. Kirakowsky, and M. Corbett, SUMI user handbook, Human Factors Research Group, University College Cork, 1993.
- [25] N. Bevan, and M. Macleod, "Usability measurement in context," *Behav. Inf. Technol.*, vol. 13, No. 1/2, 1994, pp. 132–145.
- [26] N. Bevan, J. Kirakowaski, and J. Maisal, (1991): Proceeding of 4th International Conference on HCI, Stuttgart.
- [27] P. Musilek, W. Pedryez, G. Succi, and M. Reformat, Software Cost Estimation with Fuzzy Models. ACM SIGAPP Applied Computing Review 8, 2000, pp. 24-29.
- [28] R. Agarwal, P. De, A. P. Sinha, and M. Tanniru, "On the Usability of Object Oriented Representations," *Communications of the ACM*, Vol. 43, No. 10, October 2000.
- [29] R. Molich, and J. Nielsen, "Heuristic evaluation of user interfaces," in *Proc. ACM Human Factors Computing Systems (CHI)*, New York, 1990, pp. 249–256.
- [30] S. G. MacDonell, A. R. Gray, and J. M. Calvert, "Fuzzy Logic for Software Metric Practitioners and Researchers". In: The Proceedings of the 6th International Conference on Neural Information Processing ICONIP, Perth, 1999, pp. 308-313.
- [31] S. K. Dubey, and A. Rana, "Analytical Roadmap to Usability Definitions and Decompositions", *International Journal of Engineering Science and Technology*, Vol. 2(9), 2010, pp. 4723-4729
- [32] T. M. Khoshgoftaar, and E. Allen, "Prediction of Software Faults Using Fuzzy Nonlinear Regression Modeling", Boca Raton, Florida, 2000.
- [33] W. Changqing, W. Xugang, Ma Cuixia, D. Changzhi, and D. Guozhong, "Probabilistic regular grammars for the evaluation of Human-Computer interface usability", *Journal of Computer Aided Design & Computer Graphics*, vol.17(12), 2005, pp. 2709-2715.
- [34] X. Ferre, "Incorporating usability into an Object Oriented Development Process", pp 1-7.
Accessed: September, 2011.
[Http://is.ls.fi.upm.es/Xavier/papers/usability_oo_process.pdf](http://is.ls.fi.upm.es/Xavier/papers/usability_oo_process.pdf)