Understandability Measurement in an Early Usability Evaluation for Model-Driven Development: An Empirical Study

Jose Ignacio Panach, Nelly Condori-Fernández, Francisco Valverde,
Nathalie Aquino, Óscar Pastor
Centro de Investigación en Métodos de Producción de Software
Universidad Politécnica de Valencia
Camino de Vera s/n, 46022 Valencia, Spain
Phone: +34 96 387 7000, Fax: +34 96 3877359
{jpanach, nelly, fvalverde, naquino, opastor}@pros.upv.es

ABSTRACT
Traditionally, usability has been evaluated taking into account the user's satisfaction when interacting with the software system. However, in a Model-Driven Development (MDD) process, where conceptual models are the main resource for software system generation, the usability can potentially be evaluated at earlier stages. This work goes one step further proposing that certain usability attributes, specifically internal understandability attributes, can be measured from Conceptual Models. This work presents an empirical study carried out to evaluate the proposal. The goal of this study is to evaluate whether the value measured using our proposal is related to the understandability value perceived by the end user. From the analysis of the empirical results obtained, several weaknesses of the proposal are stated.

Categories and Subject Descriptors
D.2.2 [**Software Engineering**]: Design Tools and Techniques – *Computer-aided software engineering (CASE)*. D.2.4 [**Software Engineering**]: Software/Program Verification – *Validation*. D.2.8 [**Software Engineering**]: Measures – *Process metrics*

General Terms
Measurement, Experimentation, Verification.

Keywords
Usability, metrics, indicators, understandability, automatic code generation, empirical evaluation, conceptual modeling.

1. INTRODUCTION
According to ISO/IEC 9126-1 [7], usability is composed of six sub-characteristics that can be measured by attributes. From these sub-characteristics, this paper focuses on understandability, which is defined as the *capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use*. For each sub-characteristic and specifically for understandability there are two measurable attribute types: a) *External attributes*, which are measurable during the interaction between the user and an implemented system, and b) *Internal attributes*, which can be measured before implementing and executing the system.

ESEM'08, October 9–10, 2008, Kaiserslautern, Germany.

Historically, the Human-Computer Interaction (HCI) community has been working to measure external attributes. In this context a number of measurement techniques based on surveys have been proposed, such as WAMMI [8]. A disadvantage of these techniques is the huge amount of resources that they require: task definitions, surveys, devices for recording user interaction, a group of users and the final system. Some authors such as Fraternali [5] have proposed evaluating usability at a more abstract level than surveys. Following this trend, we propose measuring understandability using internal attributes for conceptual models. We focus on understandability because, when comparing it to other usability sub-characteristics, such as attractiveness and operability, usability can be measured by means of more internal attributes. Early understandability evaluation, which is performed on conceptual models, has the advantage of diminishing the cost. Conceptual models are used as an input to the MDD process for automatic code generation, which implies that understandability evaluation can be carried out automatically before implementing the system. However, it is important to note that internal understandability is only a portion of the measurable understandability since there are many subjective attributes that can only be measured in the final system. Internal understandability could be seen as a prediction of system usability before the system is generated.

As an example of the MDD process and in order to evaluate our proposal we have chosen OOWS [4] and the software development method called OO-Method [11]. The combination of these methods generates a fully functional Web application. OO-Method and OOWS are UML-compliant Model-Based Code Generation Methods, which assures the applicability of the results to other similar methods. Section 2, below, explains our proposal to measure internal attributes and Section 3 shows how the empirical evaluation was carried out and analyzes the results.

2. A PROPOSAL FOR MEASURING INTERNAL UNDERSTANDABILITY
In an MDD method, the Conceptual Model represents the code in an abstract way; therefore, measuring internal understandability on Conceptual Models implies predicting the understandability of

the system. The advantages of this approach are the following: (1) the system does not have to be generated to carry out the evaluation; (2) the evaluation can be carried out automatically. From all the understandability attributes, the evaluation is focused only on internal attributes, in other words, external attributes that depend on the end user cannot be measured with our proposal.

The method consists of three steps: (1) Application of Measures; (2) Application of Indicators; (3) Results Grouping. This measurement method can be used in any software development method based on conceptual models, although Measures and Indicators depend on a specific software development method. As example of software development method, we have used OOWS. As follows we briefly explain this measurement method, and more detail can be found in [10].

The first step to measure internal understandability is the Application of Measures. Each internal attribute has one or more Measures based on Conceptual Primitives. Internal understandability attributes extracted from the Usability Model of Abrahao et al [1] are: (1) Brevity (BR); (2) Information Density (ID); (3) Message Conciseness (MC); (4) Navigability (N); (5) Initial Values Completion (IVC). For instance, a measure for IVC using OOWS is the number of default values that have been defined to facilitate the input data in a context. These Measures are Base Measures because they do not depend on other measures. In order to obtain a measurement that includes all the system (all contexts), we use Derived Measures (averages or percentages). For instance, the Derived Measure for IVC is the average of default values defined in all the contexts of the system. This step obtains the value for each Derived Measure.

The second step consists in the assignation of a qualitative value to each numerical value obtained by means of Derived Measures. The qualitative values are: Very Good (VG), Good (G), Medium (M), Bad (B), Very Bad (VB). Each qualitative value has a rank for each Derived Measure. The Ranks used to define the indicators have been built from the usability guidelines and usability heuristics described in the literature, as [9]. For instance, Indicators for IVC Measure are: IVC ≥ 0.20 → VG; $0.20 > IVC \geq 0.15$ → G; $0.15 > IVC \geq 0.10$ → M; $0.10 > IVC \geq 0.05$ → B; IVC < 0.05 → VB

In the last step, called Results Grouping, all Indicators are grouped together in order to provide a value for understandability. To do this, a tree is built with the following characteristics: the root is the understandability sub-characteristic, the leaves are the Measures, and the branches are internal attributes. The method used to group Measures is based on the Chung's work about non-functional requirements [3]. We have used this approach because the author proposes an evaluation based on qualitative values, which is just what we need to group together Indicators.

3. EVALUATING OUR PROPOSAL

The research question of our empirical study is whether there is a significant difference between users' perceptions about the understandability of the final applications and the value obtained with the early evaluation method. To answer this question, we have carried out a comparison between two sets of values: 1) a set of values which was obtained by two experts in measurement and OOWS using OOWS Conceptual Models; 2) a set of values of perceived understandability for end-users using web applications generated from previously evaluated Conceptual Models.

The *subjects* were twenty undergraduate students from the Technical University of Valencia. The *objects* were two web applications that had the same level of complexity: Rent-a-Car and IMDB Lite. To minimize the influence of subjective aspects, both web applications had the same visual appearance

3.1 Variables and Hypotheses

- As response variable that corresponds to the outcomes of the experiment, we have identified *understandability.*
- As variable that affects the response variable, we have identified *Evaluation Methods of understandability*. This variable has two alternatives: 1) evaluation of understandability from conceptual models, without end-users; 2) evaluation of understandability with end-users.
- **As v**ariables that we do not want to influence the experimental results, we have identified *application domain, quality of conceptual models used to generate the web applications, complexity of web applications.*

We have identified the following **hypothesis** related to Research Question: H_1: *There is a significant difference between the understandability internal measures obtained with the early evaluation method and the external measures obtained from users' perceptions*.

3.2 Instrumentation

The instruments used to carry out the experiment were: (1) Tasks: a list of tasks for each Web application that the user must carry out; (2) Survey: A list of twelve closed questions (5-point Likert scale) defined to capture the end-users' impressions. Each question refers to a Measure used to measure understandability internal attributes. Instruments used in the empirical study are available at *http://oomethod.dsic.upv.es/EmpiricalStudy*

3.3 Validity Evaluation

In this section, we discuss the threats identified in our experiment.

Random heterogeneity of subjects: All the subjects had approximately the same knowledge about web applications. This homogeneity reduces the external validity of the experiment.

Selection: the subjects had knowledge of Web Applications. The external validity of the experiment is also reduced.

Inadequate pre-operational explanation of constructs: This threat means that the constructs are not sufficiently defined, and hence the experiment cannot be sufficiently clear. We have used an inter-item correlation analysis to evaluate the construct validity of the response variable proposed by Campbell and Fiske [2]: *Convergent validity (CV)* and *Discriminant validity (DV).* This average DV should be lower than the average CV. The results of the validity analysis for each construct show that the CV value was higher than the DV value.

In addition, the *reliability analysis* on the survey was conducted using the Chronbach alpha. The value obtained was 0.60 that is the minimum acceptable level for exploratory research [6].

Representative material: In the experiment, we tried to use representative web applications generated with OOWS.

3.4 Data Analysis and Interpretation

In order to answer our research question, the scores assigned by subjects to each survey item were averaged and compared with

the value obtained using the early evaluation method. Figure 1 shows the comparison for the Rent a Car system. The values obtained using the two measurement methods (Y-axes) were very similar for ID5 (a measure for Information Density). However, for MQ and IVC the difference was lower. Similar results were obtained in the comparison of IMDB Lite.

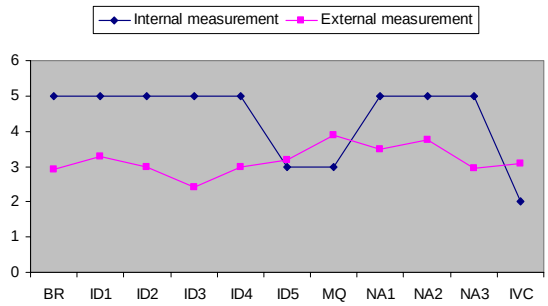

Figure 3. Comparison between internal and external measures for Rent a Car application

However, to confirm these descriptive results, our hypothesis was formally tested by verifying whether the values obtained using our approach (internal attributes of understandability) were significantly different with the mean scores (external measures). The Shapiro Wilk normality test was applied for all values assigned by the subjects. As data distribution was not normal ($p<0.05$), we used a non-parametric technique: Mann-Whitney U-test. The statistical test was applied with a significance level of 5 %, ($\alpha = 0.05$).The results of the test for the Rent a Car and IMDB Lite applications are presented in the Table 1.

Table 1. Test statistic for the Rent a Car and IMDB

Statistics	Rent Car	IMDB
Mann-Whitney U	25.000	11.000
Wilcoxon W	91.000	77.000
Z	-2.396	-3.415
Asym. Sig. (2-tailed)	.017	.001
Exact Sig [2*(1-tailed Sig.)]	.019	.001(a)

As in both applications the 2-tail value is lower than the specified significance level, we can conclude that there is a significant difference between internal understandability measures obtained with the early usability method and external measures obtained from perceptions of subjects. The main reasons for this are:

- Derived Measures are defined as the average understandability value among the Conceptual Primitives that define a Base Measure. This is not a good approach, since a context with limited understandability significantly affects the end-users' perception.
- The Ranks of the Indicators are too strict. A small difference of one unit could establish if the attribute has good or bad understandability. Indicators based on fuzzy logic could provide a more realistic evaluation.

4. CONCLUSIONS

This paper presents and evaluates a method for measuring system understandability using internal attributes represented as Conceptual Primitives in an MDD method. The aim of this empirical evaluation is to verify whether the values obtained by internal attributes correspond with the values perceived by end-users using a Web engineering method called OOWS. Even though, the results have shown that the vast majority of the internal attributes do not represent the understandability perceived by end-users, the method is still promising it. Results obtained with this experiment emphasize the importance of the empirical evaluation to improve measurement methods. It is not enough to provide an apparently adequate set of metrics and indicator: a rigorous empirical evaluation is strictly required to assess that the measurement is working in practice as expected. We were initially surprised by the unexpected, obtained results. However, we have concluded that thanks to that kind of work, we can now analyze how to properly modify both metrics and indicators.

5. ACKNOWLEDGMENTS

We would like to thank professor Joan Fons and his students of the Technical University of Valencia for their collaboration in the empirical evaluation.

This work has been developed with the support of MEC under the project SESAMO TIN2007-62894 and cofinanced by FEDER

6. REFERENCES

[1] Abrahao, S., Insfrán, E. (2006). Early Usability Evaluation in Model Driven Architecture Environments. Sixth Conference on Quality Software (QSIC'06).pp. 287-294.

[2] Campbell D. T. and Fiske D. W., "Convergent and Discriminant Validation by the Multitrait-Multimethod Matrix", in Psychological Bulletin, vol.56, 1959, pp. 81-105.

[3] Chung, L., Nixon, B., Yu, E., Mylopoulos, J. (2000). Non-Functional Requirements in Software Engineering. London, Kluwer Academic Publishing.

[4] Fons J., P. V., Albert M., and Pastor O. (2003). Development of Web Applications from Web Enhanced Conceptual Schemas. ER 2003, LNCS. Springer.pp. 232-245.

[5] Fraternali, P., P. L. Lanzi, et al. (2004). Exploiting Conceptual Modeling for Web Application Quality Evaluation. WWW 2004, New York, USA.

[6] Garson D., Scales and standard measures from statnotes, North Carolina State University, Copyright 1998, last updated March 2008. http://www2.chass.ncsu.edu/garson/pa765/standard.htm

[7] ISO/IEC 9126-1 (2001): Quality model.

[8] Kirakowski, J.; Claridge, N. (1998). Human centered measures of success in web site design. Fourth Conference on Human Factors & the Web (Basking Ridge, NJ, June).

[9] Leavit, M., Shneiderman, B. (2006). Research-Based Web Design & Usability Guidelines, U.S. Government Printing Office. Web: http://www.usability.gov/pdfs/guidelines.html

[10] Panach, I., Condori-Fernández, N., Valverde, F., Aquino, N., Pastor, O. (2007). Towards an Early Usability Evaluation for Web Applications. MENSURA 2007.pp. 67-76.

[11] Pastor, O., Molina, J. (2007). Model-Driven Arquitecture in Practice. Valencia, Springer.