

International Journal of Advance Research in Computer Science and Management Studies

Research Article / Survey Paper / Case Study

Available online at: www.ijarcsms.com

Effectiveness Estimation of Object Oriented Software: Design Phase Perspective

Pooja Gupta¹Computer Science and Engineering
Goel Institute of Technology and Management
Lucknow – India**Dr. Namrata Dhanda²**Computer Science and Engineering
Goel Institute of Technology and Management
Lucknow – India

Abstract: Effectiveness is an essential software quality factor that is useless if it is not available at an initial stage in the software development life cycle. It becomes more important in the case of object oriented design. Estimating effectiveness of object oriented design near the beginning in the development cycle, mainly at design phase; significantly reduce the development cost and rework, and as well as assists the software designers and developers for delivering high quality maintainable software within time and budget. This paper illustrates the need and significance of effectiveness at design phase and build up a Effectiveness Estimation framework and multivariate linear Effectiveness Estimation Model for Object-Oriented Design. Developed model estimates the effectiveness of class diagrams in respect of their Effectiveness, Effectiveness. Lastly the developed models have been validated using experimental tryout.

Keywords: OOD, Effectiveness, Effectiveness Estimation, Software Design, Software Quality.

I. INTRODUCTION

Software is going away to be changed several times for different reasons while being developed and particularly after it has been delivered. Commonly the term maintenance is used when referring to those changes made to software products after they have been delivered. Depending on the reasons for alteration and the wider organizational perspective, a variety of approaches to maintenance such as corrective or adaptive maintenance are or relatively should be applied [1]. Despite the truth that software maintenance is a costly and difficult task; it is not correctly managed and often unnoticed. One cause for this poor management is the lack of established measures for software effectiveness [3].

An exhaustive survey of the relevant literature reveals the fact that there is no standard methodology and /or structured guideline available to quantify effectiveness at design phase [5]. Practitioners emphasize on the need of having an organized and efficient approach for effectiveness estimation. For this reason there is a need to develop a more logical solution for effectiveness estimation at design phase [2, 21]. Formentioned facts motivated the study/researcher to make an effort in this direction and to develop a comprehensive effectiveness estimation framework and model to measure object oriented software effectiveness at design phase of development life cycle [6].

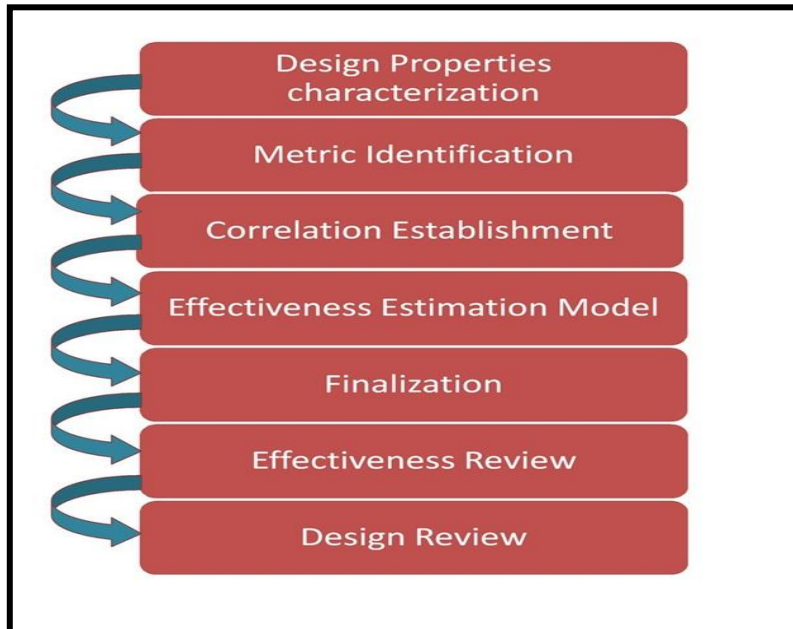


Fig 1: Effectiveness Estimation Framework: Design Phase Perspective

II. OBJECT ORIENTED DESIGN PROPERTIES

Object oriented design is the most popular concept in today's software development environment [4]. Object oriented system consider object as the primary agent involved in a computation process [7]. It requires more significant effort at the early phase in the software development life cycle to recognized objects, classes, and the relationships among them. Object oriented programming is a basic knowledge that supports quality objectives [8]. The necessity to deal with the effectiveness of software design is the essential issue that influenced the overall development cost and quality. A good object oriented design needs design procedures and practices that must be used in development cycle [10]. Their violation will ultimately have a strong impact on the quality attributes [20]. Object oriented principles direct the designers what to hold up and what to keep away from. A number of measures have been defined so far to measure object oriented design. There are several important themes of object orientation that are known to be the basis of internal quality of object oriented design and support in the perspective of estimation [9]. These themes significantly include inheritance, encapsulation, cohesion and coupling [19].

III. OBJECT ORIENTED DESIGN METRICS

The most central aim of metric selection is to pick such metrics which are statistically important and must be applicable. Studies have been conducted and found that there exists powerful relation among Object Oriented software metrics and its effectiveness. Software metrics offer an effortless and inexpensive way to identify and correct probable reasons for low software quality according to the effectiveness sub -factor as this will be supposed by the programmers [12]. Set up Estimation programs and design metric standards will support in preventing failures before the maintenance process and decrease the essential effort during that phase. Internal metrics are extremely associated with the programmers' view of effectiveness [13]. However, unhappiness with internal quality standards may not necessarily outcome in low rank of effectiveness although it is generally expected [22-24]. In that case, it is likeable that, regardless of what internal Estimations designate, the concluding judge for the effectiveness of the delivered software is the programmer [14, 16].

IV. MODEL DEVELOPMENT

Estimation of class diagram's Effectiveness is prerequisite for the accurate effectiveness Estimation. For this reason prior to developing EEM^{OOD}, the study has developed models for Effectiveness. In order to set up the models subsequent multivariate linear model (1) has selected.

$$Y = \mu + \beta_1 * X_1 + \beta_2 * X_2 + \dots + \beta_n * X_n + \epsilon \quad (1)$$

Eq. (1)

Where

Y is dependent variables.

X1, X2... Xn are independent variables.

β1, β 2... β n are the coefficients.

€ is error term

μ is the intercept.

V. EFFECTIVENESS ESTIMATION MODEL

In order to set up an Effectiveness estimation model of object oriented class diagram, metrics listed in [15] will play the role of independent variables while Effectiveness will be taken as dependent variable. The data used for developing Effectiveness model is taken from [17]. The correlation among Effectiveness Factors and Object Oriented Characteristics has been established as depicted in equation2. Using SPSS, values of coefficient are calculated and Effectiveness model is originated as below.

$$\text{Effectiveness} = -4.081 + 4.645 \times \text{Encapsulation} + 11.996 \times \text{Inheritance} + 2.701 \times \text{Coupling} - .506 \times \text{Hierarchies}$$

Eq. (2)

The Coefficients part of the output gives us the values that we need in order to write the regression equation (4). The Standardized Beta Coefficients give a measure of the contribution of each variable to the Effectiveness model. A big value designates that a unit change in this predictor variable has a large effect on the criterion variable. The t and Sig (p) values give a rough indication of the impact of each predictor variable – a big absolute T value and small p value suggests that a predictor variable is having a large impact on the criterion variable. The experimental evaluation of Effectiveness is very encouraging to obtain effectiveness index of software design for low cost testing and maintenance.

Table 1: Coefficients for Effectiveness Estimation Model

Coefficients ^a						
Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error			
1	(Constant)	-4.081	4.229		-.965	.511
	Encapsulation	4.645	2.966	.228	1.566	.362
	Inheritance	11.996	1.982	.492	6.053	.104
	Coupling	2.701	.738	1.310	3.659	.170
	Hierarchies	-.506	.068	-1.827	-7.391	.086

a. Dependent Variable: Effectiveness

The descriptive statistics of the output gives the mean, standard deviation, and observation count (N) for each of the dependent and independent variables and is shown in Table 4.2.

Table 2: Descriptive Statistics for Effectiveness Estimation Model

Descriptive Statistics			
	Mean	Std. Deviation	N
Effectiveness	8.1357	3.12306	6
Encapsulation	.8867	.15319	6
Inheritance	.5417	.12813	6

Coupling	1.7167	1.51449	6
Hierarchies	6.0000	11.27830	6

The Model Summary table of the output is most useful when performing multiple regression. Capital R is the multiple correlation coefficients that tell us how strongly the multiple independent variables are related to the dependent variable. R square is very supportive as it gives us the coefficient of determination. The Model Summary is shown in Table 4.3.

Table 3: Model Summary for Effectiveness Estimation Model

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.999 ^a	.997	.987	.34954

VI. EMPIRICAL VALIDATION

Empirical validation is a vital phase of proposed research. Empirical validation is the standard approach to justify the model approval. Taking view of this truth, practical validation of the effectiveness model has been performed using sample tryouts. In order to validate developed effectiveness model the data has been taken from [11].

Table 4: Computed Ranking, Actual Ranking and their Relation

Projects ↓	Effectiveness Ranking		$\sum d^2$	r_s	$r_s > \pm .781$
	Computed Rank	Known Rank			
P1	7	4	9	0.945455	✓
P2	10	8	4	0.975758	✓
P3	9	9	0	1	✓
P4	5	1	16	0.90303	✓
P5	6	2	16	0.90303	✓
P6	8	3	25	0.848485	✓
P7	3	7	16	0.90303	✓
P8	1	5	16	0.90303	✓
P9	4	10	36	0.781818	✓
P10	2	6	16	0.90303	✓

Sperman's Coefficient of Correlation r_s was used to check the significance of correlation among calculated values of effectiveness using model and it's 'Known Values'. The ' r_s ' was estimated using the method given as under: Sperman's Coefficient of Correlation.

$$r_s = 1 - \frac{6 \sum d^2}{n(n^2 - 1)} \quad -1.0 \leq r_s \leq +1.0$$

'd' = difference between 'Calculated ranking' and 'Known ranking' of effectiveness.

n = number of projects used in the experiment.

The correlation values between effectiveness through model and known ranking are shown in table (4) above. Pairs of these values with correlation values r_s above $[\pm .781]$ are checked in table. The correlations are up to standard with high degree of confidence, i.e. up to 99%. Therefore we can conclude without any loss of generality that effectiveness Estimation model measures are really reliable and significant and applicable.

VII. CONCLUSION

The study has developed model to compute effectiveness of the class diagrams. Effectiveness model measures the effectiveness of class diagrams in terms of their design properties. This paper developed Effectiveness Estimation framework and multivariate linear Effectiveness Estimation Model for Object-Oriented Design. Effectiveness model have been developed using the method of multiple linear regressions. The study moreover validates the quantifying ability of effectiveness model. The applied validation on the effectiveness model concludes that proposed model is highly consistent, acceptable and considerable. The values of effectiveness are of instant use in the software development process. These values help software designers to review the design and take proper corrective measures, early in the development cycle, in order to control or at least reduce future maintenance/testing cost.

References

1. IEEE Press, "IEEE Standard Glossary of Software Engineering Technology," ANSI/IEEE Standard 610.12-1990, 1990.
2. ISO, "International standard ISO/IEC 9126. Information technology: Software product evaluation: Quality characteristics and guidelines for their use." 1991.
3. Abdullah, Dr, Reena Srivastava, and M. H. Khan. "Testability Estimation of Object Oriented Design: A Revisit". International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 8, pages 3086-3090, August 2013.
4. Zhao, L.: A New Approach for Software Testability Analysis. In: Proceeding of the 28th International Conference on Software Engineering, Shanghai, pp. 985-988 (2006)
5. Huda, M., Arya, Y.D.S. and Khan, M.H. (2014) Measuring Testability of Object Oriented Design: A Systematic Review. International Journal of Scientific Engineering and Technology (IJSET), 3, 1313-1319.
6. Abdullah, Dr, Reena Srivastava, and M. H. Khan. "Testability Measurement Framework: Design Phase Perspective". International Journal of Advanced Research in Computer and Communication Engineering, Vol. 3, Issue 11, Pages 8573-8576, November 2014.
7. Dromey, R.G.: A Model for Software Product Quality. IEEE Transaction on Software Engineering 21(2), 146-162 (1995).
8. Huda, M., Arya, Y.D.S. and Khan, M.H. (2015) Quantifying Reusability of Object Oriented Design: A Testability Perspective. Journal of Software Engineering and Applications, 8, 175-183. <http://dx.doi.org/10.4236/jsea.2015.84018>
9. Abdullah, Dr, Reena Srivastava, and M. H. Khan. "Modifiability: A Key Factor To Testability". International Journal of Advanced Information Science and Technology, Vol. 26, No.26, Pages 62-71, June 2014.
10. Gao, J., Shih, M.-C.: A Component Testability Model for Verification and Measurement. In: Proc. of the 29th Annual International Computer Software and Applications Conference, pp. 211-218. IEEE Comp. Society (2005)
11. Bansiya, Jagdish, and Carl G. Davis. "A hierarchical model for object-oriented design quality assessment." Software Engineering, IEEE Transactions on 28.1 (2002): 4-17.
12. Abdullah, Dr, M. H. Khan, and Reena Srivastava. "Testability Measurement Model for Object Oriented Design (TMM^{OOD})". International Journal of Computer Science & Information Technology (IJCSIT), Vol. 7, No 1, February 2015, DOI: 10.5121/ijcsit.2015.7115.
13. Huda, M., Arya, Y.D.S. and Khan, M.H. (2015) Testability Quantification Framework of Object Oriented Software: A New Perspective. International Journal of Advanced Research in Computer and Communication Engineering, 4, 298-302. <http://dx.doi.org/10.17148/IJARCCE.2015.4168>
14. Fu, J.P. and Lu, M.Y. (2009) Request-Oriented Method of Software Testability Measurement. Proceedings of the ITCS 2009 International Conference on Information Technology and Computer Science, Kiev, 25-26 July 2009, 77-80.
15. Huda, M., Arya, Y.D.S. and Khan, M.H. (2015) Evaluating Effectiveness Factor of Object Oriented Design: A Testability Perspective. International Journal of Software Engineering & Applications (IJSEA), 6, 4149. <http://dx.doi.org/10.5121/ijsea.2015.6104>
16. Lee, Ming-Chang. "Software Quality Factors and Software Quality Metrics to Enhance Software Quality Assurance." British Journal of Applied Science & Technology 4.21 (2014).
17. Huda, M., Arya, Y.D.S. and Khan, M.H. (2015) Metric Based Testability Estimation Model for Object Oriented Design: Quality Perspective. Journal of Software Engineering and Applications, 8, 234-243. <http://dx.doi.org/10.4236/jsea.2015.84024>
18. Badri, M. and Toure, F. (2012) Empirical Analysis of Object-Oriented Design Metrics for Predicting Unit Testing Effort of Classes. Journal of Software Engineering and Applications, 5, 513-526. <http://dx.doi.org/10.4236/jsea.2012.57060>
19. Abdullah, Dr, M. H. Khan, and Reena Srivastava. "Flexibility: A Key Factor To Testability", International Journal of Software Engineering & Applications (IJSEA), Vol.6, No.1, January 2015. DOI: 10.5121/ijsea.2015.6108
20. Mouchawrab, S., Briand, L.C. and Labiche, Y. (2005) A Measurement Framework for Object-Oriented Software Testability. Information and Software Technology, 47, 979-997. <http://dx.doi.org/10.1016/j.infsof.2005.09.003>
21. Jungmayr, S. (2002) Testability during Design, Softwaretechnik-Trends. Proceedings of the GI Working Group Test, Analysis and Verification of Software, Potsdam, 20-21 June 2002, 10-11.
22. Bruntink, M. and Van Deursen, A. (2004) Predicting Class Testability Using Object-Oriented Metrics. Proceedings of the Fourth IEEE International Workshop on Source Code Analysis and Manipulation, Chicago, 15-16 September 2004, 136-145.
23. Amin, A. and Moradi, S. (2013) A Hybrid Evaluation Framework of CMM and COBIT for Improving the Software Development Quality.
24. Zheng, W.Q. and Bundell, G. (2008) Contract-Based Software Component Testing with UML Models. International Symposium on Computer Science and Its Applications (CSA '08), 978-0-7695, 13-15 October 2008, 83-102.