

# Use Case Point Method of Software Effort Estimation: A Review

Mudasir Manzoor Kirmani  
Research Scholar  
School of CS & IT  
MANUU, Hyderabad, India

Abdul Wahid  
Dean and Head  
School of CS & IT  
MANUU, Hyderabad, India

## ABSTRACT

Software project management is one of the critical activities in software development process. Many software development projects fail due to poor management of projects. The main aim of software project management team is to count what is countable, measure what is measurable and prepare measurable parameters for continuously monitor and manage a software development project. One of the very important factors in proper management of a software project is software estimation. At present most of software projects developed are using object oriented technology for the development of software projects. Researchers across the globe have used object oriented technology based method known as use case point method for estimating efforts for a software project development. This research work is an effort to review different invariants of use case point method of effort estimation. The research work also highlights the importance of continuously refining the UCP method in order to reduce the gap between the estimated and actual efforts.

## Keywords

Use Case Point, Extended Use Case Point, Software Effort Estimation, Use Cases.

## 1. INTRODUCTION

For any software development project it is very important to measure and control the processes that need to be taken care of in a development project. In order to measure a process a good software estimation methodology ensures the process predictability and clarity on timelines. Accurate estimates are not only beneficial to software engineers but to other stake holders as well. A good software estimation method leads to optimal utilization of resources and improves the quality of the end product which results in higher levels of customer satisfaction levels. This paper attempts to look at the different software effort estimation methods based on UCP method which are in use and an effort has been made to find out the limitations of different effort estimation methods which will result in establishing a platform for researchers to enhance the existing models for refinement.

Cost models like COCOMO and sizing methods like Function Point Analysis (FPA) are well known and are widely used parametric estimation models in software industry and academia. The limitation of Function Point Analysis method is that experiences people are required for counting the function points. In order to minimize the limitations of FPA method, Gustav Karner in 1993 [9] developed a method for sizing and estimating projects developed with the object-oriented method and named it as “use case point” method for estimation software efforts in the early stages of software project development. An early estimate of effort based on use cases can be made when there is some understanding of the problem domain, system size and architecture at the stage at which the estimate is made. The use case points method is a

software sizing and estimation method based on use case counts called use case points.

To gain useful information early in the software development process, the focus is mainly on two UML diagrams viz. use case diagram and class diagram. While requirement elicitation is captured through a use case diagram, detail design information is provided in class diagrams. The early work starts from use case points, which was first given by Karner (1993) [9] as a software project effort estimation model. His work has been inspiration for researchers since then and many have elaborated his approach in different contexts. Use case point approach has gained significant popularity among industry professionals as well as research scholars. There are many approaches available that exploits knowledge from use case diagrams and class diagrams in the process of predicting efforts for a system. Use case diagrams are not the only source of early estimation. Analysis class diagrams can also provide handful of information pertaining to effort required to build the software. Another approach of UML points provides simple calculation, easy to implement, and provides reasonable cost estimation in the upper stage of software development as it exploits use case diagram and class diagrams.

## 2. LITERATURE REVIEW

Damodaran, M., (2001) research work highlighted the disadvantages of using function points as it requires detailed requirements in advance and other issues in the modern object oriented systems which are designed using Use Cases methodology. The research work further mentioned that use case points (USP) offers significant practical advantage over the available estimation technique currently in use. The research work was focussed on mitigating the impact of limitations of function points method used for effort estimation which required detailed requirements in advance and related issues that the modern object oriented system encounters while using use cases.

Schneider et al. (2001) [19] mentioned that use case point estimation technique performs better than other existing techniques for software effort estimation. The study further concluded that regardless of its better performance the UCP technique still lacks standard for describing proper value for man-hour to be used for overall calculation.

Damodaran M. and Washington A, (2002) [6] conducted research intended to look at the potential of successful application of the use case point method for estimating the size and effort of software development projects. The research work highlighted that there is no proper standard for calculation of UCP moreover; criteria for counting the number of transactions in UCP is still not clear. The research work indicates that UCP method has the potential to be reliable for software effort estimation if these ambiguities are rectified

Anda, B. et al. (2002) [2] in their research work stressed on the issues that more knowledge is required about the contexts wherein UCPs can be incorporated to be adaptable to local environment in order to improve the software effort estimation process. The research work was carried out with a centralized objective of developing an estimation model based on use case model which is simple to use. The researchers applied UCPs on several projects of the Scandinavian software development companies and conducted interviews with management and people involved in development to understand the use of UCPs and the efficiency of UCP. The researchers highlighted the need for establishing a scheme for measuring improvements to the estimation process as the most obvious success criterion is the accuracy of the estimate.

Carroll, E. R. (2005) [4] conducted research work at Agilis solution and FPT software engineering teams to demonstrate the accuracy of estimates for software projects early in software development life cycle (SDLC), before the requirements are fully defined and the system is fully designed. The basis for their estimation were use cases, given point factor and modified for technical and environmental factor as per the use case method. After applying the process across more than 200 projects, the demonstrated metrics proved that the estimation was less than 9% for actual and estimated cost on 95% of the projects. The researchers categorised software projects as simple & complex, similarly proposed 28 & 20 man-hours for complex and simple category respectively. In addition to this introduced a new counting strategy for environmental factors. The research work concluded by proposing an estimation tool using use case points method and emphasized the importance of constantly modifying the model to achieve consistency in effort estimation.

Clemmons R. K., (2006) [5] carried out research work which was an attempt to establish a base line for productivity factor (PF) for web development project using Use Case Points. The productivity factor was used to determine the number of man-hours needed to complete the project. The study indicates that UCP method can produce an early estimate within 20% of the actual effort than expert and other estimate methodologies. In UCPs the factors like technical and environmental factor are precisely considered to generate precise estimation. The research work further concluded that technical and environmental factors should be given enough consideration in UCP method for software effort estimation in early stages of development. However the author encouraged researchers to use UCP method for more projects to help produce software on time and under budget.

Kusumoto, S. et al. (2007) [12] have made an effort to describe the importance of using use case point (UCP) method of effort estimation method in early stages of software development. The research work proposed an automated measurement tool U-EST which calculated use case points from use case model's written in UML files. The automated tool was applied on five different software projects and the results showed the UCP calculated using U-EST were considerably adequate.

Uzal, R. et al. (2007) [23] in their research work gave a detailed list of "use case point" method supporting different areas of PMBOK (Project Management Body of Knowledge). The research work concluded that "use case point" method is perhaps the best available tool for effort estimation of software development projects and highlighted that generalized standardization of "use case point" method will be an important contribution to software industry.

Robiolo, G. and Orosco R. (2008) [18] had given more stress on the versatility of use case point method which can be employed to estimate the effort in early stages of development. The research work was intended to find out an alternative method for estimating software effort which can give better performance when compared to function-point method of software effort estimation. The research work proposed two alternative variables which express a notion of size as transactions and entity objects and a technique for estimating effort mean productivity value. The estimation results showed good improvisation by implementing UCP but with the limitation that this can be deployed to the projects carried out in similar environmental circumstances.

Periyasamy, K. and Ghode A., (2009) [17] research work focused on the internal details of the use cases. The researchers proposed a modified software effort estimation technique based on UCP and named the same as Extended Use Case Point (e-UCP) Model. The model proposed did include use case narrative with six parameters as input parameter, output parameter, a predict in precondition, a predict in post condition, an action in successful scenario and an exception with 0.1, 0.1, 0.1, 0.1, 0.2 and 0.1 weight respectively. The second significant change proposed in the model was in classification of actor weight classification wherein the categorization was changed from 03 in UCP model to 07 in proposed model. The seven actor weight classifications proposed were very-simple, simple, less-average, average, complex, very-complex and most-complex with 0.5, 1.0, 1.5, 2.0, 2.5, 3.0 and 3.5 weight respectively. The third modification proposed was by classifying use case weight into 04 categories in comparison to 03 in UCP model. The use case weight classification in the proposed model was simple, average, complex and most complex with 0.5, 1.0, 2.0 and 3.0 weights respectively. The proposed model used 20 man-hours as suggested by Karner (1993) to estimate efforts for software development projects using e-UCP. The research work concluded by showing the improvement in the performance of e-UCP model of software effort estimation in comparison with UCP model. However, the researchers recommended to increase the size of dataset used for evaluating the performance of the proposed model.

Nassif, A. B. (2010) [14] in the research work had made an effort to enhance the use case point method of estimating effort for a software project at an early stage of development in order to improve the performance of UCP method. The research work also highlighted the scope of automating the use case method due to its simplicity, accurateness, and ease of use up to a certain degree. The research work concluded that UCP based methods performance is better when compared with other software effort estimation methods used in early stages of development process and is not the ultimate model in its present form hence needs to be further refined to remove ambiguity in the transaction description. Therefore, there is scope of improvement for better evolution of effort estimation methods.

Ochodek M., et al. (2011) [16] in their research work which was intended to investigate the construction of UCP to find out possible ways of simplifying it. The evaluation method used cross-validation procedure to compare accuracy of different variables of UCP. The analysis was based on details derived from 14 projects for which effort ranged from 277 to 3593 man-hours. The research study results indicate that the number of adjustment factors can be reduced to 06 (2 environmental factors and 4 technical complexity factors). This ambiguous nature in selecting the adjustment factors

again lands the UCP technique in confusion hence; there is wide scope to form the proper baseline for adjustment factors.

Nagar C. (2011) [13] in his study of software effort estimation indicated that UCP estimation is much better and popular method for effort calculation in early stages of development. UCP based method are more effective than FPA and COCOMO variants up-to a certain degree. The research work justified the improvements in UCP method by increasing the number of adjustment factors from 21 to 29 is contradictory with Ochodek, M, et al (2011). Hence this behaviour of increasing the adjustment factors increases the uncertainty in estimation. The need of the hour is, improvisation in UCP to form concrete basis for incorporation of adjustment factors for estimation.

Nassif, A. B. et al. (2011) [15] proposed a model for software effort estimation in early phases of software development using use case points, the proposed model takes into account, the non-linear relationship between software size and software effort, and the impact of project complexity and productivity factor. The research work adopted regression analysis on several projects that have similar project complexity and team productivity. The results showed that software effort estimation improved up to 16-25% by using UCP with regression model. However, the results of the proposed method did not show significant improvement for large projects but it was proposed that further refinements to the proposed method may improve the estimates for large-sized software projects as well.

Alves, R. et al. (2012) [1] made an effort to bring human-centric models into the software development lifecycle in order to enhance development practices. The research work carried out statistical analysis of the software effort estimation result of seven real world projects using UCP and iUCP modified HCI based method. The designed model incorporated HCI, revised use case & actor assessment criteria and had technical complexity factor (TCF), Environmental Complexity factor (ECF) and productivity factor (PF). The authors introduced new classification of actor and use case like simple-system-actor, average-system-actor, complex-system-actor, simple-human-actor, average-human-actor and complex-human-actor with 1, 2, 4, 3, 4 and 6 weight factors respectively. The research work concluded by highlighting the three major factors impacting the software effort estimation process as requirement volatility, software framework and project type which were included in the revised model iUCP model. The research work further recommended that a revised formula for software effort estimation method using UCP or iUCP needs to designed and evaluated by using data set from more than 07 projects.

Kumari, S. and Pushkar, S. (2013) [11] conducted a detailed review of popular software cost estimation methods to highlight the strengths and weaknesses of various software cost estimation methods. The comparative analysis of existing popular models based on performance which is evaluated using MMRE (Mean Magnitude of Relative Error) and PRED (Prediction concluded that it is difficult to judge the better model because all effort estimation models have varying nature and suggested that a new model should be put in practice. Therefore, one of the potential solutions to this problem can be software effort estimation method based on UCP method with refined evaluation criteria, including parameters like actor, use cases, TCF and ECF which might help in bridging the gap between the effort estimation models and the software industry.

Azzeh, M. (2013) [3] analyzed the potential use case point estimation model for global projects. The research work suggested that use case points can prove to be a better alternative to effort estimation in the early stages of development process for global software development projects. The study suggested modification on the environmental factors of the generic Use Case Point method of software effort estimation as there is still not a well established base for enlisting the adjustment factors clearly.

Tanuu and Kumar, Y., (2014) [22] gave a detailed comparative analysis of different cost estimation methods used in research domain in general and software industry in particular. The research highlighted that COCOMO-II 2000 produced better estimation results than other traditional methodologies. However UCP can be accepted as best technique for estimation after subjecting it to further refinement in its build-up.

Subruadi, A. et al. (2014) [23] reviewed UCP method proposed by Karner (1993 to evaluate the effectiveness of effort rate on the estimation of the software project development using UCP. The study describes that UCP is a better approach for the effort estimation but there is still ambiguity in clearly defining certain variables or factors for effort rates like complexity of the project and quality of human resource involved. The study further gave an insight into the fact that there are no proper criteria for deciding the ER (Effort Rate Value) across different domains.

Sholiq, et al. (2014) [22] focused on to improve ER (Effort Rate Value) on the use case point (UCP) method for estimation effort for website development. The UCP estimation was adopted as a method of choice to calculate the effort estimation rate and it was concluded with emphasizing on the significance of refinement of UCP method of effort estimation.

Jena P. P. and Mishra S. (2014) [8] in their study concluded that the effectiveness of the UCP method largely depends on the way use case transactions are counted. Therefore, the need for further planning is required to simplify the process of counting transactions by redefining the process. The research study also emphasized the need of revisiting number of environmental factors in order to obtain more precise and accurate results.

Wang et al. (2009) [24] extended the UCP model by constructing a probabilistic cost model by integrating a fuzzy set theory with Bayesian Belief Networks with the UCP model. The evaluation of the extended UCP shows slim improvement in comparison with the original UCP.

### 3. FINDINGS AND RECOMMENDATIONS

The comparative list of finding and recommendations from different research publication selected in this review are given in table (1).

<b>Research Work</b>	<b>Findings</b>	<b>Future research recommendations</b>
Schneider et al. (2001) [19]	Better performance of UCP	Standardization of man-hours to calculate efforts
Damodaran, M. et al.	No proper standard for calculation of	Removal of ambiguities to

(2002) [6]	UCP and counting number transactions	improve performance
Anda, B., et al. (2002) [2]	Developed an estimation model based on UCP	Technique for measuring performance of UCP
Carroll, E. R. (2005) [4]	UCP as best method for estimation; Proposed an estimation tool	Continuous refinement to achieve consistency in effort estimation.
Clemmons, R. K. (2006) [5]	Baseline productivity for web application	Increase the size of data-set for testing performance
Kusumoto, S., et al. (2007) [12]	Proposed U-EST automated measurement tool	Continuous improvement and refinement of parameters in UCP.
Robiolo, G., et al. (2008) [12]	An alternative to UCP and proposed two alternative variables	More evaluation needs to be done; To be tested for different projects
Periyasamy, K., et al. (2009) [17]	Proposed e-UCP; Performance improved.	Increase the size for data set for testing performance
Nassif, A. B., (2010) [14]	UCP performs better in comparison to other models	UCP needs to be refined.
Ochodek, M., et al. (2011) [16]	Tried to simplify UCP method Reduced TCF to 04 parameters  Reduced ECF to 02 parameters	Overall refinement of parameters in UCP.
Nagar C., (2011) [13]	UCP performs better	Refinement is needed for better results
Nassif, A. B., et al. (2011) [15]	Proposed model based on non-linear relationship between software size & effort;  Productivity improved from 16-25% using UCP model.	Needs to be tested for large sized software projects.
Alves, R., et al. (2012) [1]	Proposed Human-centric iUCP model with Technical, Environmental and Productivity Factors	Revised formulae using UCP & iUCP needs to be designed
Kumari, S., et al. (2013) [11]	Evaluated performance of UCP using MMRE and PRED(x) to compare with other models	To revisit number of parameters in technical and environmental factors.
Azzeh, M., (2013) [3]	Analyzed UCP for global software	modification in environmental

	projects	factors for generic UCP model
Tanuu, et al. (2014) [22]	Comparative analysis of effort estimation models	UCP needs to be refined for acceptance
Subriadi A. P., et al. (2014) [21]	Evaluated effectiveness of effort rate on project management	Revision of UCP is needed for better estimation of efforts.
Jena, P. P., et al. (2014) [8]	Effectiveness of UCP was justified	Revisiting number of parameters in environmental and technical factors for precise estimates.

#### 4. SUMMARY

The review of the research papers in this research study emphasize on the importance of use cases in all the models discussed in different research works. The use cases are to be treated as building block for any software effort estimation method based on use case point model. Therefore, the onus lies on how well an estimator has understood the system which will ensure correctness, completeness and accuracy of use cases. Correct, complete and accurate use cases will help an estimator to count the number of transaction which will result in correctly categorizing software development project which will further strengthen the estimated efforts for the software development project at early stages of development.

A roadmap for successful implementation of any effort estimation method should start with building the foundations correctly followed by strategic planning and establishment of the processes. However, it is very important to continuously verify and validate estimates and compare the same with previously completed projects in order to evaluate performance of the effort estimation method which will result in improving the effort estimation model for software development projects.

#### 5. CONCLUSION

The review study highlights the importance of software effort estimation in the early stages of software development process in general and significance of use case point method of software effort estimation in particular. The research papers reviewed in this study have advocated for use case point method as an effective method for software effort estimation in early stages of software development process. The research papers from different authors reviewed emphasized on refining use case point method of estimation for software development projects. However, in order to have effective UCP method of estimation it needs to be review and refine the way use case transactions are counted. Some of the researchers in the research papers studied have also highlighted the importance of revisiting and refining the parameters used in environmental and technical factors which are directly linked to the estimates calculated using UCP method of estimation. Therefore, the need of the hour is to further refine the parameters used in environmental and technical factors of UCP method with improved and easy-to-use method for counting the number of use case transactions which may lead to more effective UCP model of estimation of efforts for software development projects. In order to obtain more precise and accurate results the UCP method of software effort estimation should be continuously refined for improvements.

Practices need time to mature towards what we perceive as best practices and they are continuously challenged by new theories and practices. Therefore, this research work is just a platform for the research community to improve the existing models for benefit of the software industry in general and academia in particular.

## 6. REFERENCES

- [1] Alves, R., Valente, P., Nunes, N. J., (2012) Improving software effort estimation with human-centric models: a comparison of UCP and iUCP accuracy, In Proceedings of the 5th ACM SIGCHI symposium on engineering interactive computing systems, ACM, 2012, (pp. 287-296).
- [2] Anda, B., Angelvik, E., & Ribu, K. (2002). Improving estimation practices by applying use case models. In Product Focused Software Process Improvement (pp. 383-397). Springer Berlin Heidelberg.
- [3] Azzeh, M., (2013) Software cost estimation based on use case points for global software development, 5th International Conference on Computer Science and Information Technology (CSIT) IEEE, 2013, (pp 214-218).
- [4] Carroll, E. R. (2005). Estimating software based on use case points. In Companion to the 20th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications (pp. 257-265). ACM.
- [5] Clemmons, R. K. (2006). Project estimation with use case points. The Journal of Defence Software Engineering, 18-22.
- [6] Damodaran, M., and Washington, A. (2002). Estimation using use case points. Computer Science Program. Texas-Victoria: University of Houston. Sd.
- [7] Ibarra, G and Villain, P. (2010) "Software Estimation Based on Use Case Size, Brazilian Symposium on Software Engineering, 2010.
- [8] Jena, P. P., & Mishra, S. (2014) Survey Report on Software Cost Estimation using Use Case Point Method, International Journal of Computer Science & Engineering Technology Vol. 5 (4), 2014, (pp 280-287).
- [9] Karner, G., (1993) "Metrics for Objectory", Diploma thesis, University of Linkoping, Sweden. No. LiTHIDA-Ex- 9344:21.December 1993.
- [10] Kitchenham, B, Kansala, K (1993) Inter-item correlations among function points. International conference on software engineering proceedings of the 15th international conference on software engineering, Baltimore, pp 477-480
- [11] Kumari, S., and Pushkar, S., (2013) Performance Analysis of the Software Cost Estimation Methods: A Review, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 3(7), 2013 (pp. 229-238).
- [12] Kusumoto, S., Tsuda, M., and Inoue, K. (2007) On a Use Case Points Measurement Tool for Effective Project Management, In Workshop on Accountability and Traceability in Global Software Engineering ATGSE, 2007, (pp. 47-48).
- [13] Nagar C., (2011) Software effort estimation using use case point approach by increasing technical factor and experience factor, International Journal on computer science and engineering, Vol. 3 (10), (pp 3337-3345).
- [14] Nassif, A. B., (2010) Enhancing Use Case Points Estimation Method using Soft Computing Techniques, Journal of Global Research in Computer Science, Vol.1 (4), 2010, ( pp 12-21).
- [15] Nassif, A. B., Ho, D. and Capretz, L. F., (2011) Regression model for software effort estimation based on the use case point method, International Conference on Computer and Software Modelling, 2011, (pp. 106-110).
- [16] Ochodek, M., Nawrocki, J., & Kwarciak, K., (2011) Simplifying effort estimation based on Use Case Points. Information and Software Technology, Vol.53 (3), 2011, (pp 200-213).
- [17] Periyasamy, K. and Ghode, A. (2009) "Cost Estimation Using Extended Use Case Point (e-UCP) Model", International Conference on Computational Intelligence and Software Engineering, 2009.
- [18] Robiolo, G., and Orosco, R. (2008). Employing use cases to early estimate effort with simpler metrics. Innovations in Systems and Software Engineering, Vol. 4 (1), 31-43.
- [19] Schneider, G. and Winters, J. P. (2001) Applied use Cases, Second Edition, A Practical Guide. Addison-Wesley, 2001.
- [20] Sholih, Sutanto T., Widodo A. P., and Kurniawan W., (2014) Effort rate on use case point methods for efforts estimation of website of development, Journal of Theoretical and Applied Information Technology, Vol. 63 (1), 2014, (pp 209 – 218).
- [21] Subriadi A. P., Sholih, and Ningrum P. A., (2014) Critical review of the Effort rate value in use case point method for estimating software development effort, Journal of Theoretical and Applied Information Technology, Vol. 59 (3), 2014, (pp 735 – 744).
- [22] Tanuu, Kumar Y., (2014) Comparative Analysis of Different Software Cost Estimation Methods, International Journal of Computer Science and Mobile Computing, Vol. 3 (6), 2014, (pp 547 – 557)
- [23] Uzal R., Montejano G., Riesco D., and Debnath N. C., (2014) Software projects effort estimation using "use case points" method in context of Project Management Body of Knowledge, XIII Congreso Argentino de Ciencias de la Computacion CACIC, 2007, (pp 530-536).
- [24] Wang, F., Yang, X., Zhu, X. and Chen, L., (2009) "Extended use case points method for software cost estimation," in International Conference on Computational Intelligence and Software Engineering, 2009.
- [25] Y. Yavari. (2011) "Software complexity level determination using software effort estimation use case points metrics", Malaysian Conference in Software Engineering, 12/2011.