International Journal of Applied Engineering Research ISSN 0973-4562 Volume 12, Number 12 (2017) pp. 3228-3236
© Research India Publications. http://www.ripublication.com

Estimation of Software Development Efforts using Improved Delphi Technique: A Novel Approach

Ashish Rai[1]
Assistant Professor, Department of Computer Science,
Shia P.G. College, University of Lucknow, Uttar Pradesh, India

[*1]ORCID: 0000-0002-1708-4170

Prof. G. P. Gupta[2]
Professor, Department of Computer Science,
Shia P.G. College, University of Lucknow, Uttar Pradesh, India

Dr. Pankaj Kumar[3]
Associate Professor, Department of Computer Science & Engineering,
Shri Ramswaroop College of Engineering and Management, Uttar Pradesh, India

(*Corresponding Author)

Abstract

Software Estimation is done often because it is expected to help in predicting how much will the project cost and when will the project get completed. Proper analysis and Effort Estimation is necessary for successfully planning for a testing project. Any flaw in critical estimation phase, results in missing the project deadlines, reduces Return of Investment and loses of customer's faith, and in my view “Bad estimation can lead to poor distribution of work”.

There are different standard and no standard method is for testing the estimation. In many project based companies, people utilize non-standardized but conventional estimation methods to make things work. These methods might have developed over a continuous period accommodating hidden factors like nature of application under test, environment, and risk factors for that specific organization. But these methods can’t be adopted as a generalized organization standard for a mature operation model.

Keywords: Software Estimation, Project Management, Effort Estimation Delphi Technique

INTRODUCTION

Software development effort estimation, calculate the effort to develop software system development based on the input parameter. Efficient cost estimation is difficult and significant activities. Effort development Estimation is supposed to be fateful because of infrequency, underestimation bias and other goals. It may be useful as input to development repatriation and bidding rounds.

In this paper we discuss overview and improved approach of software effort estimation technique. Software metric and especially software estimation is based on measuring of software attributes which are typically related to the product, the process and the resources of software development. This kind of measuring can be used a parameters in project management models which provide assessments to software project managers in managing software projects to avoid problems such as cost overrun and behind the schedule. Underestimating the costs may result in management approving proposed systems which can exceed their budgets, with underdeveloped functions and poor quality, and failure to complete in time.

Overestimation may result in too many resources committed to the project, or, during contract bidding, result in not winning the contract, which can lead to loss of jobs. So accurate cost estimation is important and Software cost estimation involves the determination of effort in person-months, project duration in calendar time and cost in rupees. Most of the cost estimation models attempt to generate an effort estimate, which can then be converted into the project duration and cost. Accurate and reliable effort estimation is still one of the most challenging processes in software engineering. There have been number of attempts to develop cost estimation models.

There are different techniques to classify estimation methods as Expert estimation, Formal estimation model, Hybrid estimation Model. The most finding, in many predicting areas, is that grouping of estimates from autonomous sources, preferable applying dissimilar approaches, will on average increase the estimation accuracy. There are a many factors potentially representing over-optimistic effort estimates that increases accuracy of effort estimation.

Effort Estimation Techniques

There are the following approaches for Software effort estimation.

- Traditional Estimation Approach
- Heuristic or Expert judgments Approach
- Parametric or Algorithmic Approach
- Regression Approach
- Analogy Approach
- Soft Computing Approach

LITERATURE SURVEY

Most of the traditional Approach, such as regression models, function points, COCOMO and others which require effort estimation. Effort schedule are serious problems in the software development [5]. Several approaches for the cost estimation techniques are developed till date [7, 11, 23, 31].

The review from different o authors clarifies that an approach is not so successful for the estimation as hybrid or combination [16 , 18] of more than one approach can provides the accurate estimation of cost effort.

The Delphi Method [1, 2, 18] is Looping process which is used to collect and filter the judgments of experts by using a series of questionnaires with feedback. The questionnaires [15, 28] are prepared to discuss the problems, opportunities, solutions or predictions. Each questionnaire is prepared on the output and the results of the previous questionnaire. This process continued for number of times until the research question is appropriately answered. The Delphi method is being widely used throughout the world in many Industrial Sectors [11,32] including Information Technology, Transportation, Engineering, Education, Business, Health-Care etc. There is immense flexibility as how it can be used? The Delphi Method can be used whenever there is incomplete knowledge about a problem [or phenomena to focus the intelligentsia or expertise on the problems in hand. Formal software development effort estimation models have been around for more than 40 years. [16, 17] They're the subject to more than a thousand research studies and experience reports. In spite of massive effort and promotion, available formal estimation models are not in much use. It is now time to move on and focus industrial estimation process improvement work and scientific research on Judgment-based effort estimation methods [3, 6]. There are very good reasons to claim that future estimation process improvement and research initiatives should aim at better judgment-based effort estimation processes and not at better formal models. The relation between effort [8, 9, 10] and size in software development contexts is not stable. Magne Jorgensen in his "Software Development Effort Estimation : Formal Models or Expert Judgment" viewpoints, has suggested on many places that, organizations, process improvement work and research initiatives should focus on better judgment-based effort estimation processes, not on introducing or improving formal estimation models [12, 13, 14]. The essential difference between formal-model-based and expert judgment-based effort estimation is the quantification step such as a formula, which can be taken into consideration from Putnam Model, at the time of analyzing the data and presentation of result.

Wideband Delphi approach was introduced at Rand Corporation, the technique can help you to estimate, plan and schedule almost anything [19, 20]. It originates from the Delphi Method that was developed in the 1940s as a forecasting tool. It has since been adopted many areas [24,25] to estimate many kinds of software.

[29, 30] Boehm's original coordinator provides each specialist with a specification and an estimating form.

Planning Poker [21, 22] is a consensus-based method for estimating, commonly used to predict effort or approximate size of tasks in project development. The technique was described by James Grenning and later popularized by Mike Cohn in the book Agile Estimating and Planning. Planning Poker [26,27] is based on a list of features to be delivered and several copies of a deck of numbered cards. Work Breakdown Structure is a decomposition of the total work to be carried out on software into smaller units that can be estimated, evaluated, and tracked.

THE DELPHI METHOD

Delphi Technique is a consensus-based effort development estimation technique that involves experts from estimation, software development Approach and application areas. A team of 4 to 8 members constituted with a moderator. The estimation process conducts first kickoff meeting, the estimation team creates a wbs, discusses list of assumptions and after the meeting, every team member initiates effort estimation for each task, then, in an estimation session, the team revises the estimates as a group & achieve consensus.

The Delphi method has been used in research to develop, identify, forecast and to validate in a wide variety of research areas. Three round Delphi is typical, single and double round Delphi studies have also been completed. The number of experts vary from 4 to 17. The method can be modified to suit the circumstances and research question.

Analysis shown in following table reveals the flexibility of the method. Their focus, number of rounds and participants are varied from software to software.

Table 1: Delphi Method Diversity

Non IS/IT Study	**Delphi Focus**	**Rounds**	**No. of Experts**
Gustafson, Shukla, Delbeeq & Walster (1973)	Estimate almanac events to investigate Delphi accuracy	2	4
Kuo & Yu (1999)	Identify national park selection criteria	1	28
Nambisan et al. (1999)	Develop a taxonomy of organizational mechanisms	3	6
Lam, Petri & Smit (2000)	Develop rules for a ceramic casting process	3	3
Roberson, Collins & Oreg (2005)	Examine and explain how recruitment message specificity influences job seeker attraction to organizations.	2	171
IS/IT Study	**Delphi Focus**	**Rounds**	**No. of Experts**
Niederman, Brancheau & Wetherbe (1991)	Survey senior IS executives to determine the most critical IS issues for the 1990s.	3	114, 126 & 104
Brancheau, Janz & Wetherbe (1996)	Survey SIM members to determine the most critical IS issues for the near future	3	78, 87 & 76
Scott (2000)	Rank technology management issues in new product development projects	3	20
Brungs & Jamieson (2005)	Identify and rank computer forensics legal issues	3	11

EXISTING DELPHI METHOD

The following Model Describe the existing Delphi Methods

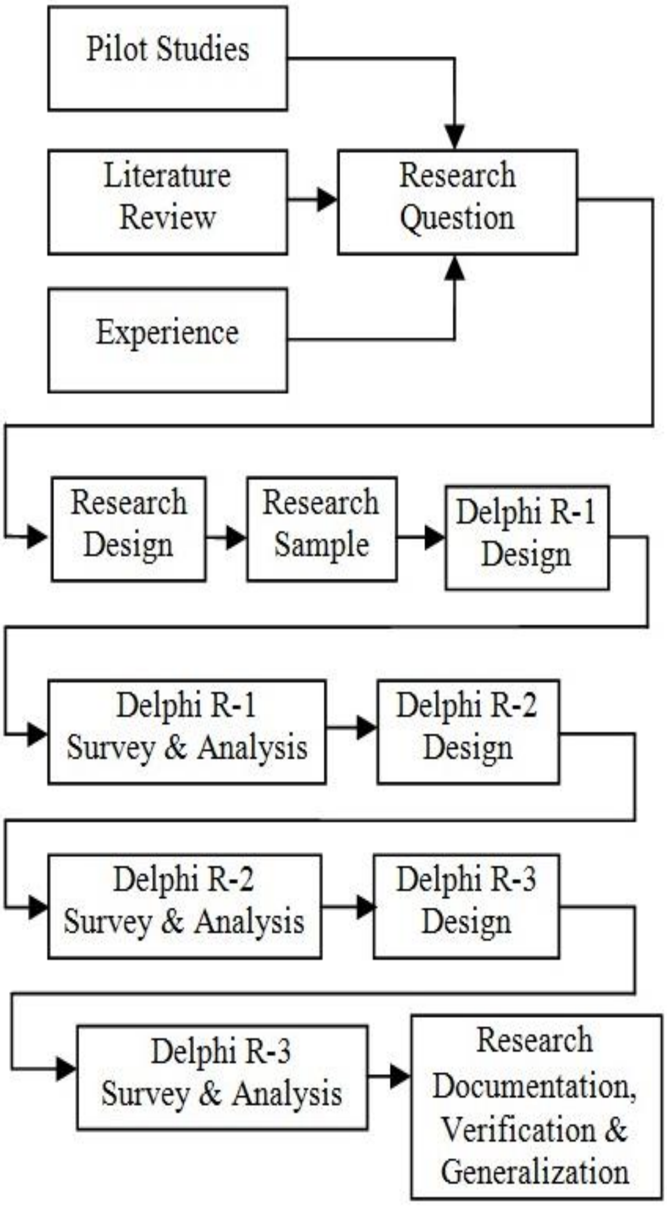

Figure 1. Typical Three Round Delphi Method

The classical Delphi method by four key features

Anonymity of Delphi participants

It allows the participants to freely express their opinions without undue social pressures to conform from others in the group. Decisions are evaluated on their merit, rather than who has proposed the idea.

Iteration

It allows the participants to refine their views in light of the progress of the group's work from round to round.

Controlled feedback

It informs the participants about other participant's perspectives, and provides the opportunity for Delphi participants to clarify or change their views.

Statistical aggregation of group response

It allows for a quantitative analysis and interpretation of data.

IMPROVED DELPHI METHOD

There is the following suggested improved Delphi approach for efforts estimation.

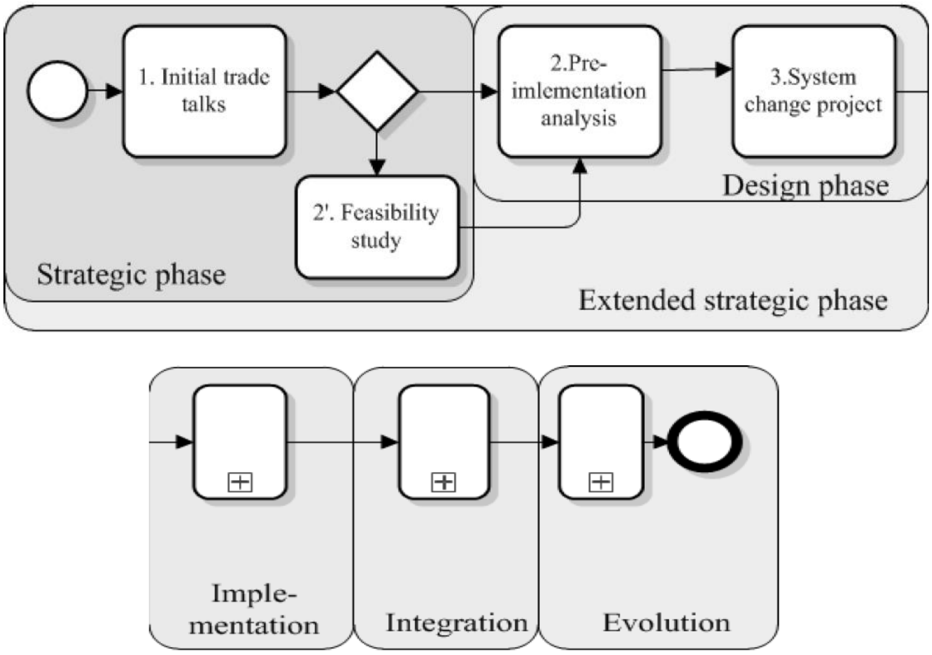

Figure 2. The stages of strategic phase and other phases during Software Development

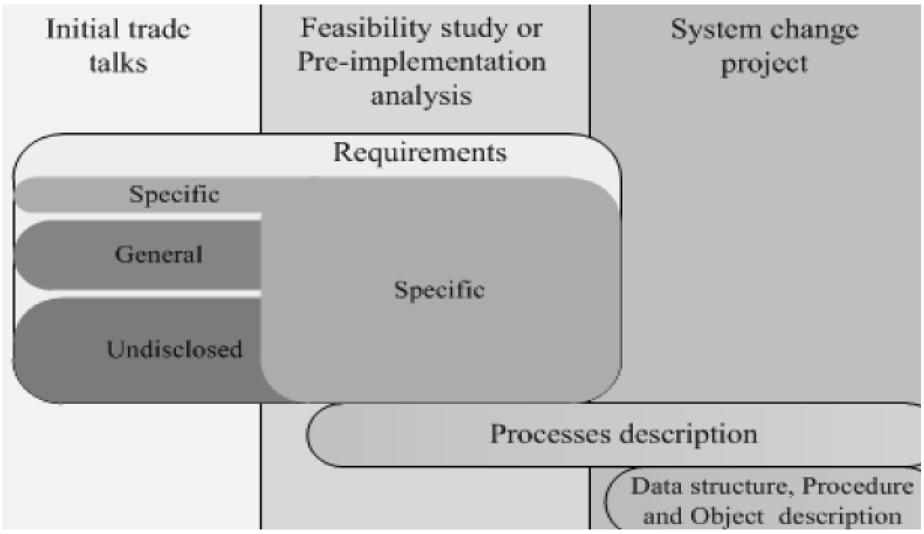

Figure 3. Input information for Effort Estimation process

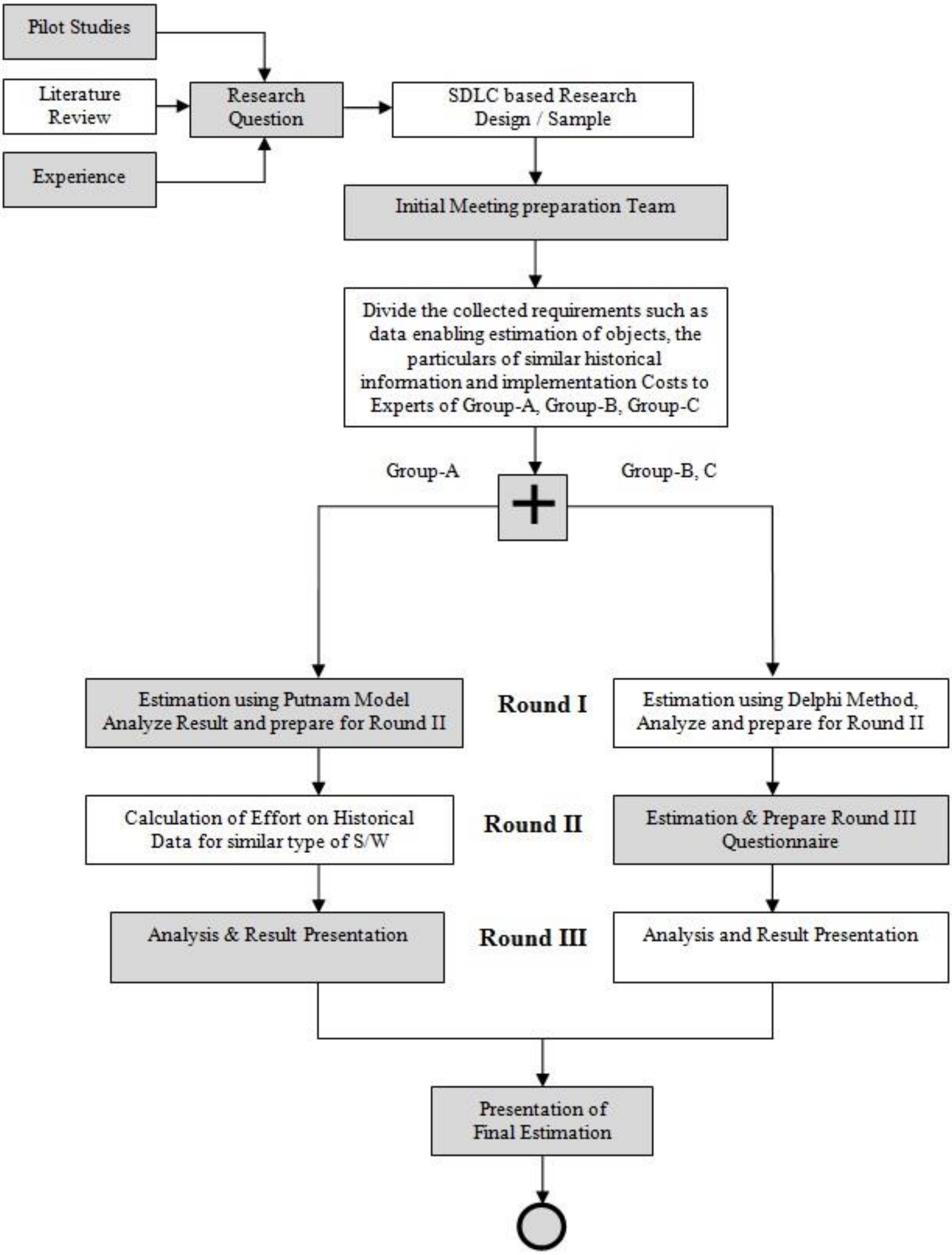
Pilot Studies
Literature Review
Experience
Research Question
SDLC based Research Design / Sample
Initial Meeting preparation Team
Divide the collected requirements such as data enabling estimation of objects, the particulars of similar historical information and implementation Costs to Experts of Group-A, Group-B, Group-C
Group-A
Group-B, C
Estimation using Putnam Model Analyze Result and prepare for Round II
Round I
Estimation using Delphi Method, Analyze and prepare for Round II
Calculation of Effort on Historical Data for similar type of S/W
Round II
Estimation & Prepare Round III Questionnaire
Analysis & Result Presentation
Round III
Analysis and Result Presentation
Presentation of Final Estimation

Figure 4. Improved Effort Estimation process model

DESIGN FRAMEWOK

Delphi suggests three distinct characteristics useful for study and will be included anonymity, Interaction with controlled feedback and Statistical group response.

Apart from above three, judgmental combination has been taken into consideration by dividing the panel of experts into three groups and their effort estimation will be taken into consideration at the time of final presentation of Result.

All the above mentioned characteristics will be administered using a database website, e-mail over the Internet.

METHODOLOGY

Participants Selection and Anonymity

The software experts, who will participants in the study, should have an experience of software development of atleast three to five years. According to their experience in the different fields and the length of the software in terms of LOC, they will be required to participate. As reported by Clark, in order to establish any novel or revised course of study, past curricula must continuously be examined to bring about further reform. The Delphi method should be used whenever a consensus is needed from persons who have knowledgeable about a particular subject.

All expert panel members will be contacted via e-mail, speed post or personally requesting them for voluntarily participate in further study.

The criteria for each expert panelist are based on the following qualifications:-

- Panelist should have at least 3 to 5 years of field experience.
- Participants should have graduated from the University.
- Panelist should have experience of software development of simple and complex computer software projects.
- Panelist should have at least teaching experience for more than 5 years if he belongs to teaching community.

The ‘expert panel’ and the ‘review panel’ will be drawn from the same pool of participants. Regarding the selection of the subjects for a Delphi study, choosing the appropriate subjects is the most important step in the entire process because it directly related to the quality of the results generated. Referencing the total group of panelists, the participants have to be split up into groups. The panelists will be assigned to one of the three aforesaid groups, and the review panelist will also be from each group and will be selected randomly, as the software experts employed in this study, will be of higher level of credibility in relation to the forthcoming results.

The duty of the review panelists will be to review the information compiled from each round, comment on the direction of the contents and give suggestions for the next round.

Size of Panel in Improved Methods

The experts of the various software fields, associated with a review panel, will enhance the qualitative area of research. Key aspects include panel selection, including experts’ qualification, size and participant commitment.

It has been suggested that reasonable results can be obtained from small panels of 10-15 individuals. However Turnoff recommended that a Delphi sample size should be atleast between 10 to 50 participants. Atleast 30 experts, combined with three review panel experts would be a desirable panel size for this study as well as to eliminate of any chance of being biased one, if any. Kennedy recommended that the researcher specified the specializations and qualifications of each panelist which will be done through survey.

The software experts will be gathered and finalized by taking following aspects into consideration:-

- Highest Degree obtained.
- Major area for each degree which software expert holds.
- Current position in the relevant field.
- The class is being taught.
- Period of experience in the field.
- Any experience in course designing and development.

Instrument and Execution

The ‘Call for Participation’ for interested parties will be completed through online survey or through form based survey. The response after survey will be analyzed and after that the survey for each round will be created, so that the review panelists can comment on the clarity and consistency of the instructions and data received. After getting the opportunity to respond, recommend changes and the desired modifications the instructions would be sent to the expert panelists by including the purpose of study.

After getting all the consent forms, the instructions for the first round will be sent to the review panel. Report following four individual phases in the web-based Delphi process:

The first phase explores the subject being researched.

The second phase moves to determine an understanding of how the entire group views the issue.

The third phase used to explore disagreement and determine reasons for differences.

ROUND I

The instrument for Round I of the web-based Delphi have to be created based on the core topics.

Round I consisted of an unstructured, open-ended instrument that should have a space for additional suggestions. After completing the first round instrument, the panelists have to submit their responses, within the survey. Results will be then summarized and fed to the participants, allowing reeasement and new recommendations.

ROUND II

The instrument for Round II will be developed on the basis of the information compiled from Round I and will be sent to the review panel for authentication before being sent to the expert. Once it is got approved by the review panel, the expert panel will be notified of the availability of the Round II instrument.

Participants will be requested to rate the importance of the competencies as represented from Round I, using a point Likert scale with a score of '5' representing the highest degree of importance and '1' representing the lowest.

Expanding further, the Likert scale will exist as:

1 = Strongly Disagree

2 = Disagree

3 = neither Disagree nor Agree

4 = Agree

5 = Strongly Agree

Once the results will be collected, the like responses will be combined and analyzed in preparation for Round III.

ROUND III

Within this round, the expert panel ranked the objectives from most important competency to the least important competency. Once the expert panel completed Round III by ranking the competencies and objectives in order from 1 to n and submit their responses, the results will be collected.

There are five levels of correlation:

1. Perfect
2. High
3. Moderate
4. Low
5. Absence of Correlation

The resulting correlations gathered from the data will enable further to observe favored competencies from the expert panelists. Competencies in the overall correlation range moderate to perfect correlations, in order to be ranked in final Time. The mean and median values will assist in showing whether or not consensus reached. Round III will be the final round, to determine the consensus of the competencies and objectives. The competencies that receive higher consensus from the expert panel will be summarized. Once the review panel approves the final instrument, it will be made available to the expert panel for final responses. The expert panelists will be asked only to 'accept' or 'reject' any remaining competencies from Round Three and no modifications will be allowed at this final stage.

After the results will be submitted, they will be collected and analyzed by utilizing the Chi-Square test. This analysis of choice will be displayed in contingency tables to indicate the relationship and significance between the variables. Chi Square will also give the ability to measure the degree of stability among the answers collected in the Delphi study, adding validity to the responses garnered.

MERITS OF DELPHI METHOD

There are many merits for Improved Delphi technique such as its usefulness when the organization does not have any in-house expert with the domain knowledge or the development platform experience to come out with a quick estimate, very quick to derive an estimate, simple to administer use, and If appropriate, experts are chosen carefully, the results can be surprisingly accurate

CONCLUSION AND FUTURE SCOPE

Software development efforts era is at demanding phase and cost estimation effort always remains an open challenge, as complex task.

The review of Software development has their significance presence in the estimation. In this Research paper we describe a novel approach of Estimation of Software Development Efforts over Improved Delphi Technique. The stages of strategic phase and other phases during Software Development provide a improved developed accommodating hidden factors like application under test, environment, and cost factors for that specific organization. The purpose of such approach, provide a estimated efforts for software development.

The study clearly confirms that judgment decision inconsistencies and overconfidence are an issue, means that expert judgment should not be used as an estimation method but rather point to the need to supplement the judgment processes that account for the biased observed.

We proposed to collect and summarize the collected information by improved judgment process. We implement in new phases such as Preparation of invitation letters/ e-mail for experts to attend the questionnaire meeting as planned in the design framework and also Questionnaire preparation for various improved Rounds of customer satisfaction questionnaire implementation with analysis and paper writing.

ACKNOWLEDGMENTS

I acknowledge my utmost gratitude and immense respect to Prof. G. P. Gupta [Director-Department of Computer Science, Shia P.G. College, University of Lucknow, Uttar Pradesh, INDIA] and Dr. Pankaj Kumar [Associate Professor-Department of CSE, Shri Ramswaroop College of Engineering and Management, Uttar Pradesh, INDIA] for their encouragement, inspiration and insightful suggestions.

REFERENCES

[1] A. B. Nassif, L. F. Capretz and D. Ho, "Enhancing Use Case Points Estimation Method using Soft Computing Techniques," *Journal of Global Research in Computer Science,* vol. 1, no. 4, pp. 12-21, November, 2010.

[2] D. D. Galorath and M. W. Evans, *Software Sizing, Estimation, and Risk Management.* Boston, MA, USA: Auerbach Publications, 2006.

[3] C.Francalanci and M. Francesco, "The impact of complexity on software design quality and costs," in *European Conference on Information Systems,* 2008, pp. 1-13.

[4] A. B. Nassif, L. F. Capretz and D. Ho, "Software estimation in the early stages of the software life cycle," in *International Conference on Emerging Trends in Computer Science, Communication and Information Technology,*2010, pp. 5-13.

[5] D. S. Kushwaha and A. K. Misra, "Software Test Effort Estimation," ACM SIGSOFT Software Engineering Notes, Vol. 33, No. 3, May 2008.

[6] P.S. Sandhu, P. Bassi and A. S. Brar, "Software Effort Estimation Using Soft Computing Techniques," World Academy of Science, Engineering and Technology, 2008, pp. 488-491.

[7] M. Chemuturi, "Software Estimation Best Practices, Tools & Techniques: A Complete Guide for Software Project Estimators," J. Ross Publishing, Lauderdale, July 2009.

[8] Free Software Foundation, "Neuroph Framework," Version 3, June 2007.

[9] M.Braz and S Vergilio, "Software Effort Estimation Based on Use Cases," 30th Annual International Com puter Software and Applications Conference (COMP- SAC'06), Chicago, 17-21 September 2006, Vol. 1, pp. 221-228.

[10] J. Kaur, S. Singh and K. S. Kahlon, "Comparative Analysis of the Software Effort Estimation Models," World Academy of Science, Engineering and Technology, Vol. 46, 2008, pp. 485-487.

[11] K. Molokken-Ostvold and M. Jørgensen, "A Review of Surveys on Software Effort Estimation," in 2003 ACM-IEEE International Symposium on Empirical Software Engineering (ISESE 2003), Frascati, Monte Porzio Catone (RM), ITALY, 2003, pp. 220-230.

[12] J. M. Verner, W. M. Evanco, and N. Cerpa, "State of the practice: An exploratory analysis of schedule estimation and software project success prediction," Information and Software Technology, vol. 49, pp. 181-193, 2007.

[13] A. L. Lederer and J. Prasad, "A causal model for software cost estimating error," IEEE Transactions on Software Engineering, vol. 24, pp. 137-148, 1998.

[14] A. L. Lederer and J. Prasad, "Nine management guidelines for better cost estimating," Communications of the ACM, vol. 35, pp. 51-59, 1992.

[15] J. D. Procaccino and J. M. Verner, "Software project managers and project success: An exploratory study," Journal of Systems and Software, vol. 79, pp. 1541-1551, 2006.

[16] N. Agarwal and U. Rathod, "Defining `success' for software projects: An exploratory revelation," International Journal of Project Management, vol. 24, pp. 358-370, 2006.

[17] R. Berntsson-Svensson and A. Aurum, "Successful software project and products: An empirical investigation," in Proceedings of the 2006 ACM/IEEE International symposium on empirical software engineering Rio de Janeiro, Brazil: ACM Press, 2006.

[18] J. Wateridge, "How can IS/IT projects be measured for success?," International Journal of Project Management, vol. 16, pp. 59-63, 1998.

[19] S.-W. Lin and V. M. Bier," A study of expert overconfidence," Reliability Engineering & System Safety, vol. 93, no. 5, pp. 711-721, May 2008.

[20] Jørgensen, M., "Estimation of Software Development

Work Effort: Evidence on Expert Judgment and Formal Models", International Journal of Forecasting, Vol. 23, Issue 3, pp.449-462, 2007.

[21] Sheppard, M. Cartwright, M. Kadoka, G. "On Building Prediction Systems for Software Engineers", Empirical Software Engineering, Vol. 5, pp. 175–182, 2000.

[22] Foss, T. Stensrud, E. Kitchen ham, B. Myrtveit, I. "A Simulation Study of the Model Evaluation Criterion MMRE". IEEE Transactions on Software Engineering, Vol. 29, No.11, 2003

[23] James Grenning Planning, Poker Renaissance Software Consulting April 2002

[24] Mike Cohn, "Agile Estimating and Planning". Mountain Goat Software November 2005.

[25] Galorath D., Evans M., "Software Sizing, Estimation, and Risk Management", Boca Raton, FL: Auerbach, 2006.

[26] Wei Lin Du, Danny Ho, Luiz Fernando Capretz, ―Improving Software Effort Estimation Using Neuro-Fuzzy Model with SEER-SEM‖, Global Journal of Computer Science and Technology, Vol. 10 Issue 12, pp.52-64, October 2010.

[27] Engineering Function Points and Tracking System, Software Technology Support Centre, Retrieved on May 14, 2008.

[29] Capers Jones, "Software Engineering Best Practices": pp. 318–320, October 2009.

[30] K. Vinay Kumar, V. Ravi, Mahil Carr and N. Raj Kiran, (2008), "Software development cost estimation using wavelet neural networks", Journal of Systems and Software, Volume 81, pp 1853-1867. Sinhal et al., International Journal of Advanced Research in Computer Science and Software Engineering 3(6), June - 2013, pp. 1120-1135 2013

[31] Prasad Reddy P.V.G.D, Sudha K.R, Rama Sree P and Ramesh S.N.S.V.S.C, ―Software Effort Estimation using Radial Basis and Generalized Regression Neural Networks‖, Journal of Computing, Volume 2, Issue 5, pp 87-92, 2010.

[32] C. J. Burgess, M. Lefley, "Can Genetic Programming improves Software Effort Estimation? A Comparative Evaluation", Machine Learning Applications in Software Engineering: Series on Software Engineering and Knowledge Engineering, pp. 95–105, 2005. [42]. A.P. Engel Brecht, Fundamentals of Computational Swarm Intelligence, John Wiley & Sons, New Jersy, 2006.