INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 4, Issue 3)
Available online at: www.ijariit.com

Spectrum of software cost estimation approaches for successful project management

Kartiki Bhamre
kartikinikam@gmail.com
Vidya Pratishsthan's Institute of Technology, Baramati, Maharashtra

Dr. Amol C. Goje
director@viitindia.org
Vidya Pratishsthan's Institute of Technology, Baramati, Maharashtra

ABSTRACT

At the present day, the business state is dynamic as well as complex. The entire organization, especially management comes across multiple issues while managing the software projects like changing requirements, escalation in new services/demands, tight budget plans and fast turnaround demands. On the other hand, organizations are trying to perform outstanding with fewer resources and finance. Hence it is essential to gauge every aspect of business especially software project management. This paper introduces techniques for estimating software cost and effort required for software development, launch, and maintenance. Software projects are critically known for going past their deadlines, crossing budget limits or both. The problem resides in estimating the number of efforts required for the development of a project. Software cost estimation is the process of predicting the effort, time required for developing a software system. The cost estimation is also depending on the size of a project which may be a line of code or function-point. There are various techniques for performing software cost estimation including algorithmic or non-algorithmic models. Each model has its own advantages and margins. The thought of paper accomplishes on the range of software cost estimation models and their performance with respect to effective software project management practices.

Keywords: *Software cost estimation, Project management, Algorithmic methodologies, Non- Algorithmic methodologies*

1. INTRODUCTION

Nowadays business structures are both dynamic and complex. The organization faces issues because of constantly changing requirements and growing demands, as well as tight finances and turnaround demands. Fundamentally organizations are taking efforts to do more with limited assets, with less finance, and, in many cases with a reduced workforce. Therefore, it is necessary to optimize multiple features of the business, especially of project management and software cost estimations. It is very well known that performance and outcome will be maintained always.

Traditional project management and estimations are task-driven and predictive; It is presumed that circumstances affecting the project are predictable. But existing and upcoming projects are constantly growing, integrating with other application and services. Requirements are changing frequently. Hence project needs highly iterative and incremental process in which constant communication is required and predictability of schedules, efforts, size, and cost should be gauged. Even functions of project management and business analysis is an inherent and critical element to success.

The procedure of calculating the time, plan, effort, size of the software solution, and overall cost associated with developing the software application is stated as software cost estimation. A software cost estimation accomplished at the start of the project helps to decide which functions can be involved inside the project.

The challenges and risks of the project are rises when the most important functions are involved in the completion of the project. Hence, cost estimation may have a large influence on the life cycle and its schedule. This is the most complicated work for managing and conserving software. While in entire development and implementation, development method cost and time assessment perform vital share in software cost estimation method.

In spirit, both algorithmic and non-algorithmic estimation approaches represent calculations, principles, and practices that aim to deliver measurable results which are symbolized as Software Cost Estimation.

The aim of this paper is demonstrating abilities of software cost estimation methods. The entire paper is organized as Vital Intent of successful Project Management. Overview of Software Cost Estimation, segregation of software estimation methodologies, models, analysis of software cost estimation approaches and final conclusion is illustrated.

2. VITAL INTEND: SUCCESSFUL SOFTWARE PROJECT MANAGEMENT

Heart of each project lays in a plan that drives the activities and this result in the ultimate outcome of the intended project or product initially envisioned by the client. In the project management world, the client, it may be internal or external is very important. Delivering to the customer what was planned in a timely manner, makes the difference between a good customer experience and erroneous one. The major thing about every project plan is that the project stakeholders get involved to define the level of complexity, life cycle, external affecting factors and structure of the project. Thus, few roles like project manager, coordinators are accountable for building the plan to manage client's expectations while identifying the requirements. Similarly, efficiencies and needs of team members have to be checked so that deliverables can be efficiently managed and the project should hit the decided and highest point.

Developing the perfect project plan need the skill set to truly examine team members' efficiencies, needs, required technologies and hardware while satisfying the final customer. Depending on the project, varying levels of detail are required in the project plan. Projects always require the careful guidance of team members along a detailed path; a more detailed plan makes more intelligence. The project manager is ultimately responsible for delivering a successful project. [4], [6]

2.1 Lifecycle of the project management

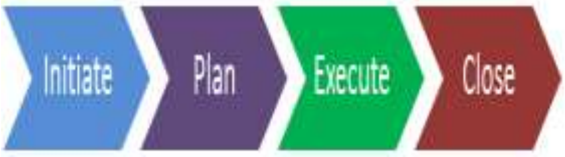
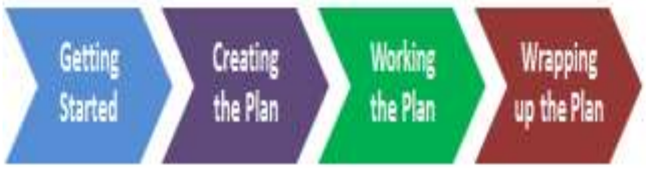


Fig. 1: project management lifecycle

Above are the stages of a project and they illustrate the most efficient path from beginning to end. Generally, the project gets started executing, but then realize that team is unaware of what they are executing. So the team has to come back and restart the lifecycle path and work out what they want to do. In this case, the team is unaware of or can't gauge the deadline. Similarly, the project cannot track details of the budget. Hence team has to work on planning. In this case cost of project shoots up because of rework and elapsed time. While visualizing this scenario, consider organization is trying to launch a new product to market before your competitor and unaware of when it will be ready to launch. Then marketing team cannot do their job. Similarly, if cost is unidentified or misidentified then no one can decide on a price. Hence organization or team has to consider the more mature cycle of project execution as below.



Getting Started

Project Kick off

Project Objective & Scope

Project Schedule and Budgeting

Stage Schedule and Budgeting

Project Organization

Project Control Procedures

Business Case

Project Initiation Stage Assessment

Project Control

Stage Kick off

Project Board Meetings

Quality Control

Progress Control

Change Control

Issues Management

Exception Situation

Stage End Assessment

Project Closure

Final Product Evaluation

Project Completion

Process Improvement

A. Focus on Project Schedule and Estimations

For the successful completion of project following points need to ponder and mandatory to execute.

- Develop time schedule for the project
- Defines the overall development approach for the project
- Selects the appropriate template for creating an estimation
- Need to produce a customized work breakdown structure (task list) specific to the project,
- Need to create a range of estimate for the target completion date and the associated total cost of the project
- Identifying the major Project Board control points
- Identifying the Client, development and operations resources required to accomplish the project
- Identifying hardware, software, contractor, and training resources required to carry out the project

Hence a schedule is always useful to guide the project and to use as a benchmark by which progress on the project can be monitored and controlled. As the objective and scope of the project are decided, then it is necessary to determine what activities are mandatory to be carried out to meet the objective. The dependencies between these activities can then be determined, which in turn allows resources and timescales to be estimated. This task needs to be done for the entire project plan and for all activities of the project, and then in more detail for the next stage of the project. This step is concerned with the overall project. The project schedule is not planned to direct resources to specific activities on certain dates.

The ideal situation is to have a proven process of estimation or budgeting on the basis of the project tasks. This may be a standard IT process or an outside process. It is necessary to build a process from a series of the kernel and common steps.

While executing a project plan, once a process is chosen, it should be customized to the particular needs of the project. If a proven process is not available, it is mandatory to develop a process for the project which conforms to the stage, step, and task. Next, it is necessary to estimate the resources, both technical and non-technical. These will be expressed by resource role as Systems Analyst, and Analyst Programmer, developer and many rather than by individual.

The necessities as other types of resource, such as office space and equipment, electricity, transport, marketing, accommodation should also be determined. A timeline and cost estimate for the project can then be developed. The development of the project schedule is an iterative activity during Project Initiation. The project plan applies to medium, large, multi-stage projects. For few medium-sized, single stage projects it is not necessary to create a project plan. The detailed stage plan will define the whole project from beginning to end.

The reason for the project schedule is to offer an overall structure to the project. The top-down stage and step estimates will be cross-checked and revised as and when each stage schedule is produced with detailed bottom-up estimates. Creating and customizing steps for "Basic Work Breakdown Structure" creating is always effective. Determine Project Stages and Steps, Review the stages and steps from the selected process template. On the basis of the project objective, scope, assumptions, and constraints, alter and tune the project work breakdown structure to:

- Add new stages and/or steps,
- Remove unnecessary stages and/or steps,
- Merge stages and/or steps, removing stages and steps will probably increase project risk.
- Document the risks and identify proposed countermeasures.

Project schedule estimates will be determined as duration or elapsed time. Determine the duration of each step. It is recommended that you estimate in units of a day. Step-level estimates are unlikely to be less than one day. When estimating duration, include expected non-productive and non-effective time. Alternatively, produce an initial estimate for each stage and allocate to the component or steps. It may be desirable to express the stage and step estimates as ranges, these may be the best case and worst case. It needs to create a single estimate as above and then extrapolate the best and not effective cases.

Determine the proposed project start date and follow the schedule to measure the stage and step start and end dates. Cross check these dates with respect to the project constraints. Adjust scope, dependency, and resources until an acceptable equilibrium is achieved. Revise required hypothesis that has been modified. Prepare and follow Project Budget Estimate for the internal and external staff costs by step. Staff costs are a function of step duration, resource equivalent, and chargeable rate.

Estimate non-staff costs as hardware, technologies, project training for the project team to develop the application, installation training, training for clients, operations, etc. to use the application. Miscellaneous costs as supplies, printing, or other costs which cannot be classified above.

These will be used to measure the success of the process used for the project management. Analyze project schedule; carry out a product review of the Project Schedule and Estimate to ensure that the scope meet up the business requirements with all technical standards. [5]

B. Software Cost Estimation

Estimation is the process of gauging the controls, processes, required properties and their effective calculations. This value is utilizable for definite determination even if input essentials may be incomplete, uncertain, or unstable. The value is yet helpful because it has been derived from the best information available. [1], [7]

The gauging factors provide information that can be predictable, through various formal or informal processes, to determine a range express the absent information. An estimate which turns out to be inappropriate will be an overestimate if the estimate exceeded the definite result. And it will be underestimated if the estimate cut down of the actual result.

The 'Wideman Comparative Glossary of Common Project Management Terms' defines estimating cost as, "*The process of forecasting a future result in terms of cost, based upon information available at the time.*"

Software cost estimation comes up and studied in several ways, there is the participation of various concept technique, method, and even model. SCE is significant pre-development activity before starting actual development of software system. In spite of the great number of cost factors collected and the exact data collection, a lot of ambiguity in the estimated can be observed. The key strength of effective Cost estimation practices is for scheduling, visualizing future of the project, impact on the organization and financial growth rate of the industry.

Nowadays software cost of estimation has become a complex branched science hence many functional sizing techniques, sizing metrics, cost and effort models come into sight. Next section shows the common techniques used in Software Cost Estimation.

The history of Software cost Estimation began in 1960 by Frank Freiman to build up the concept of parametric estimation models. This leads to the development of PRICE model for hardware, in 1970. The researchers from this area analyze many projects using different techniques attempting to identify the factors which affect the cost of software development using correlation and regression techniques. After that, by the end of 1980, the Constructive Cost Model COCOMO is being formulated by Barry W. Boehm and PRICE software cost estimation parametric model was developed by the end of the same decade. Allan Albrecht and John Gaffney of IBM developed function point analysis FPA to estimate the size and effort of information systems. In 1990 the principal person of software cost estimation researcher Barry Boehm reformulates his model into COCOMO II which consists of three main submodels as Application Composition, Early Design, and Post architecture models where he used many software sizing models like Object Points, Function Points and a source line of code. In last two decades, in Software Cost Estimation approaches gets the rapid exponential improvement in software and information technology industry. Specifically the researches who handle the new phenomena in software engineering such as reusable component, Object Oriented environment, agile projects, Component-Based Programming, real-time systems etc via new sizing techniques, inventing some new measures to measure the new existed items and improving and standardize the previous models and techniques to be applicable to the new projects and overall in Information technology and related industry.

Basically, from the literature review and practices observed in the industry researcher come to know that there are multiple shades and ranges of software cost estimation approaches. Those approaches are nothing but a combination of methodologies, models, techniques, and processes. Methodologies are purely a concept of thought based on theories and industrial observations. Models are built upon that identified concept in a structured way. There are two types of methodologies,

1. Algorithmic Methodologies / Models
2. Non-Algorithmic Methodologies

Algorithmic Methodologies / Models

Algorithmic models also called parametric models. These methods start using a formula to calculate the cost estimation. In this method, costs are analyzed using mathematical formulae, inputs with metrics to produce an estimated output. This method uses the mathematical equations to accomplish the application estimation. The exact equations used in historical equations or theory.

Non-Algorithmic Methodologies

In this, method estimation process is done according to the analysis of previous datasets. Non-algorithmic methods usually do not use any formula to calculate the software cost estimation. This method makes evaluation purely on basis of comparison between previous dataset and existing dataset. After understanding the categories of software cost estimation, we see the spectrum of the concept of Software Cost Estimation, which shows efficiency, reliability, and accuracy without applying the exhaustive procedures.

Here researcher has covered 10 Software cost estimation approaches. These approaches help to improve correctness of software development effort estimation techniques and models have been proposed. These approaches include Algorithmic and Non-Algorithmic estimation, analogy based estimation, data mining techniques, rule initiation, soft computing techniques, and function point based analysis model, neural network based approaches and expert judgment based techniques.

A Headway of estimation echoes the reality of project's progress. It accomplishes cost or budget & controls overruns. Not an individual method is necessarily superior or inferior to the other, in actual, strengths and weaknesses of each are often complementary to one other. So introducing and focusing on the estimation methods seem necessary for achieving to the accurate and reliable estimations. In the current study, most of the earlier and present estimation approaches have been illustrated systematically.

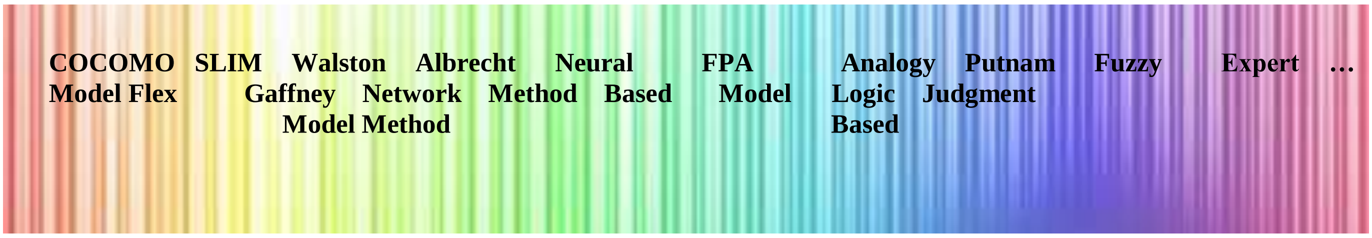

COCOMO SLIM Walston Albrecht Neural FPA Analogy Putnam Fuzzy Expert …
Model Flex Gaffney Network Method Based Model Logic Judgment
Model Method Based

Fig. 2: Spectrum of Algorithmic and Non-Algorithmic Software Cost Estimation Approaches

Table 1: Illustration of Software Cost Estimation Approaches

Cost Estimation Model	Type of Method	Effort Equation and description	Explanation	Benefits	confines
COCOMO Model	Algorithmic	E = a× (KLOC)*b × EAF Developed by BOHEM, constant value a, b, depends on project type whether it is organic semi-detached or Embedded.	Developed by BOHEM, constant value a, b, depends on project type whether it is organic semi-detached or Embedded.	The very common approach provides Clear results.	This model is not suitable for many projects as large Amount of data is required.
SLIM Model		Technical constant, C = Size × B1/3 × T4/3 Total person months, B=1/T4 × (Size/C)*3. T=Development Time in years. C= Parameter dependent on development Environment.	It is empirical effort estimation model developed by Lawrence H Putnam in 1978. It provides description of Time and effort needed to complete a s/w project of Specified type.		
Walston - Felix Model		E = 5.2 (KLOC) 0.91 D = 4.1 (KLOC) 0.36	Developed by C.E.Walston and C.P. Felix in 1977. It is a method of programming Measurement and estimation.		

Albrecht-Gaffney MODEL		It uses Function point to estimate Efforts.	Developed by IBM DP Services Organizations.		
Kemerer MODEL			It is cost estimation model uses Function points and Linear Regression.		
Neural network based estimation methods	Non-algorithmic (Machine Learning Method)	These methods provide the power of reasoning and are consistent with unlike databases.	Large training data is Required. Lack of adequate amount of data set effects Performance, no guidelines are available for designing.		
Function point analysis	Algorithmic	Results are better than SLOC, Language independent, function points are based on a system users external view of the system, Non-tech users have a better Understanding of what FP is Measuring. Resulting metrics are straightforward and logical.	Mechanization is hard to do as precise counting require In-depth knowledge of Standards.		
Analogy-based estimation	Non-algorithmic	Having special experts is not important,		Works based on actual experiences	Key info of past & same project is required. It may not available at times.
Putnam model	Algorithmic	This model is basically based on two variables which are time and size.			This does not ponder all other aspects of software development lifecycle.
Fuzzy logic based estimation	Non-algorithmic (Machine Learning Method)	This approach is capable to handle ambiguity & Provides reliable estimates.		A short training is required, another advantage is its Flexibility.	This method is hard to use, estimation cost of complex features is much tedious.
Expert Judgment	Non-Algorithmic		Very effective for finding data and requirements of complex, large size project which needs extensive experience		Repeated and its success depends on expert. Many times it remains incomplete.

Lefley and Sheppered
Oliveira
Reddy et al
Witting and Finnie
Andreou et al

Fig. 3: Development and Verdicts of Software Cost Estimation Models in Recent times

Table 2: Description of Appraches and Findings

Sr. No.	Team of Investigator / Researcher	Description of approaches and findings
1	Lefley and Shepperd [3]	Team has worked on the concept of genetic programming to improve software cost estimation on public datasets with great success
2	Oliveira [17]	Approach is founded on comparative study on support vector regression (SVR), radial basis functions neural networks (RBFNs) and linear regression for estimation of software project effort. Result of this clears that SVR significantly outperforms RBFNs and linear regression.
3	Reddy et al [18]	By putting on Gaussian membership function which provides better performance than the trapezoidal function. It is useful for presenting cost drivers and work enhanced fuzzy approach for software effort of the COCOMO.
4	Witting and Finnie [1][2]	To evaluate software development effort it describes use of back propagation learning algorithms on a multilayer perception.
5	Andreou et al [19]	This is measured as Fuzzy Decision Trees (FDTs) for estimating required effort and software size in cost estimation as if robust evidence about those fuzzy transformations of cost drivers contributed to enhancing the prediction process.

Envisioning developments of Software Cost Estimation

Further research would be conducted in a different context to see if larger projects or continuous product development projects suffer from traditional problems. The research of more accurate size and effort estimation methods led to revising former models and approaches to this problem. New estimation approaches include many alternatives of Putnam's model, function point analysis, application of fuzzy logic, neural network, test execution effort, and Bayesian belief network.

Bayesian Belief Networks is Software cost and effort estimation is the process of forecasting the software effort to estimate software costs of both development and maintenance. An important fact to realize about Bayesian Belief Networks is that they are not dependent on knowing exact historical information or current evidence.

3. CONCLUSION

This paper focuses on the earlier, existing and envisioning software estimation methods. Software cost estimation is really a vital activity that requires knowledge of a number of key attributes. At the early stage of a project, there is high uncertainty about these project attributes. Information about the past and/or the current situation is vague, incomplete, conflicting, and uncertain. Estimation techniques focus only on the actual development effort furthermore. This paper provides a review of the importance of Cost Estimation as a key factor within Project Management Life Cycle. For accomplishing successful project Software Cost estimation is mandatory. In absence of estimation, the project cannot move ahead. It actually affects to the business relationship with customer profit as loss of customer or getting more work from the customer. Hence it directly hit the financial growth of organization.

4. REFERENCES

[1] H. Agahi, S. Malhotra, and J. Quirk, Estimating Software Productivity and Cost for NASA Projects, Journal of Parametrics, pp. 59-71, 1998.

[2] S. Chulani, B. Boehm and B. Steece, From Multiple Regression to Bayesian Analysis for Calibrating COCOMO, Journal of Parametrics, vol. 15(2), pp. 175-188, 1999.

[3] M. Lefley and M. J. Shepperd, Using Genetic Programming to Improve Software Effort Estimation Based on General Data Sets, LNCS, Genetic and Evolutionary Computation — ISBN: 978-3-540-40603-7, page-208, GECCO 200.

[4] Boehm, 1981 "Software Engineering Economics", Prentice Hall.

[5] Boehm, B. W. and P. N. Papaccio (1988). ―Understanding and controlling software costs" IEEE Transactions on Software Engineering 14(10): 1462-1477.

[6] Benton, B.: Model-Based Time and Cost Estimation for Software Testing Environment. In: Sixth International Conference on Information Technology: New Generations, pp. 801-806. IEEE (2009)

[7] Ms. Aruna Shrivastava, Asst. Professor Leena Sahu - Software Cost Estimation: A Survey of Current Practices for International Journal of Engineering and Technical Research (IJETR) ISSN: 2321-0869, Volume-3, Issue-5, May 2015.

[8] System Level Metrics for Software Development Estimation by Ricardo Valerdi, Yue Chen, Ye Yang,Center for Software Engineering, University of Southern California.

[9] Issues in Software Cost Estimation by Abedallah Zaid, Mohd Hasan Selamat, Abdual Azim Abd Ghani, Rodziah Atan and Tieng Wei Koh, Faculty of Computer Science and Information Technology, University Putra Malaysia.

[10] Accounting Testing in Software Cost Estimation: A Case Study of the Current Practice and Impacts by Jurka Rahikkala, Sami Hyrynsalmi, Ville Leppänen Vaadin Ltd, Turku, Finland

[11] Review of Various Software Cost Estimation Techniques by Shivangi Shekhar, Umesh Kumar Govt.Women Engg.College,Ajmer,Rajasthan for International Journal of Computer Applications (0975 – 8887) Volume 141 – No.11, May 2016.

[12] Caper Jones, "Estimating Software Cost", Tata Mc–Graw Hill Edition,2007.

[13] Kusuma Kumari B.M, "Software Cost Estimation Techniques", International Journal of Engineering Research in Management and Technology, Volume -3, Issue- 4,2014.

[14] M.V.Deshpande, S.G.Bhirud, “Analysis of Combining Software Estimation Techniques”, International Journal of Computer Application (0975-8887)”, August,2010.

[15] Chetan Nagar, Anurag Dixit, ”Software efforts and cost estima- tion with sy stematic approach”, IJETCIS, ISSN:2079 -8407, V ol.2 No.7, July 2011.

[16] Pressman, Roger S., ”Software Engineering: A Practioner’ s Approach”, 6th Edn., McGraw-Hill New York, USA., ISBN: 13: 9780073019338,2005.

[17] Pichai Jodpimai, Peraphon Sophatsathit, and Chidchanok Lursin- sap, ”Analysis of Effort Estimation based on Software Project Models”, IEEE, 2009.

[18] Andriano L.I. Oliveira, Estimation of Software Project Effort with Support Vector Regression, www.journals.elsevier.com/neurocomputing, Vol.- 69, Issues 13–15, pp. 1749–1753, August 2006.

[19] S. Reddy, K. Raju, An Improved Fuzzy Approach for COCOMO„s Effort Estimation using Gaussian Membership Function, Journal of Software, VOL. 4, NO. 5, pp. 452-459, 2009.

[20] [19] A. S. Andreou, E. Papatheocharous, Software Cost Estimation using Fuzzy Decision Trees, 23rd IEEE/ACM International Conference on Automated Software Engineering, pp. 371 - 374, 2008.

[29]

[21] T.N.Sharma, Anil Bhardwaj, Anita Sharma, Cost Estimation Tool for Commercial Software Development Industries for International Journal of Computer Trends and Technology- volume3Issue2- 2012

[22] Manisha Arora, Richa Arya, Dinesh Tagra, Anil Saroliya, Varun Sharma, Cost Estimation Tool for Commercial Software Development Industries for International

[23] Journal of Computer Trends and Technology- volume3Issue2- 2012

[24] Steve McConell,”Software Estimation Demistifying the Black Art “, Microsoft Press, Dreamtech Press 2013

[25] M. A. Parthsarathy, “Practical Software Estimation” for PEARSON Education, 2007

[26] “FAHSCEP: Fuzzy and Analogy based Hybrid Software Cost Estimation Process”, International Review on Computers and Software, 2013.

[27] S. Reddy, K. Raju, An Improved Fuzzy Approach for COCOMO„s Effort Estimation using Gaussian Membership Function, Journal of Software, VOL. 4, NO. 5, pp. 452-459, 2009.

[28] Jurka Rahikkala, Sami Hyrynsalmi, Ville Leppänen Vaadin Ltd, Turku, Accounting Testing in Software Cost Estimation: A Case Study of the Current Practice and Impacts, Finland.