УДК 004.9:005.8
DOI https://doi.org/10.32689/maup.it.2023.2.9

Marian SLABINOHA
PhD, Associate Professor, Associate Professor at Computer Systems and Networks Department, Ivano-Frankivsk National Technical University of Oil and Gas, Karpatska str., 15, Ivano-Frankivsk, Ukraine, 76019 (marian.slabinoha@nung.edu.ua)

Taras MYKHAILOV
Product Manager, Contractor at Dev.Pro, Klovskyi uzviz str., 7, Kyiv, Ukraine, 01021 (taras.mykhailov@gmail.com)

Мар'ян СЛАБІНОГА
кандидат технічних наук, доцент, доцент кафедри комп'ютерних систем і мереж, Івано-Франківський національний технічний університет нафти і газу, вул. Карпатська, 15, Івано-Франківськ, Україна, індекс 76019 (marian.slabinoha@nung.edu.ua)

ORCID: 0000-0002-7296-0356

Тарас МИХАЙЛОВ
продакт менеджер, контрактор, Dev.Pro, вул. Кловський узвіз, 7, Київ, Україна, індекс 01021 (taras.mykhailov@gmail.com)

ORCID: 0009-0006-3071-2529

Bibliographic description of the article: Slabinoha M., Mykhailov T. (2023). Zastosuvannia metodiv ekspertnoi otsinky pry prohnozuvanni terminiv vykonannia proektiv v haluzi IT [Using expert judgement techniques to estimate IT projects completion time]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society,* 2(8), 79–83. DOI: https://doi.org/10.32689/maup.it.2023.2.9

Бібліографічний опис статті: Слабінога М., Михайлов Т. Застосування методів експертної оцінки при прогнозуванні термінів виконання проектів в галузі ІТ. *Інформаційні технології та суспільство*. 2023. Вип. 2(8). 79–83. DOI: https://doi.org/10.32689/maup.it.2023.2.9

USING EXPERT JUDGEMENT TECHNIQUES TO ESTIMATE IT PROJECTS COMPLETION TIME

Accurate estimation is crucial for the success of software development projects, but overestimation or underestimation of tasks remains a common issue. This article aims to explore the use of expert judgement techniques to improve IT project completion estimations. Expert judgement involves seeking input from experienced professionals to estimate the work required for a task. However, expert judgement is subject to biases and cognitive biases that can affect the accuracy of estimates. This research examines the causes of over and underestimation in software development projects and proposes practical solutions to reduce the risks associated with expert judgement estimation.

The study utilizes a case study approach and compares single-point estimation with the Program Evaluation and Review Technique (PERT) to improve accuracy. The results demonstrate that PERT estimation provides a range of estimates with an expected case, and the average estimation is closer to the best case than the original estimates. Additionally, confidence intervals are calculated to provide a range of estimates at different confidence levels. The findings of the study demonstrate that PERT estimation, with its range of estimates and expected case calculation, improves estimation accuracy compared to single-point estimation. Additionally, the research calculates the Mean Relative Error (MRE) to measure the accuracy of estimates, indicating the potential for ongoing improvement in estimation accuracy. This article contributes to the field of IT project management by highlighting the importance of expert judgement techniques, identifying the challenges associated with over and underestimation, and providing practical tools and methodologies to enhance estimation accuracy. By minimizing subjectiveness and continuously improving estimates, IT projects can be delivered on time and within budget, leading to improved project outcomes and stakeholder satisfaction.

Key words: *project management, project estimation, expert judgement, PERT, confidence intervals.*

ЗАСТОСУВАННЯ МЕТОДІВ ЕКСПЕРТНОЇ ОЦІНКИ ПРИ ПРОГНОЗУВАННІ ТЕРМІНІВ ВИКОНАННЯ ПРОЕКТІВ В ГАЛУЗІ ІТ

Точна оцінка має вирішальне значення для успіху проектів розробки програмного забезпечення, але переоцінка або недооцінка завдань залишається загальною проблемою. Метою статті є дослідження використання методів експертного оцінювання для покращення оцінки термінів виконання ІТ-проекту. Експертне судження передбачає звернення до досвідчених професіоналів для оцінки роботи, необхідної для виконання завдання. Однак експертне судження піддається суб'єктивізму і когнітивним упередженням, які можуть вплинути на точність оцінок. У цьому дослідженні розглядаються причини завищеної та заниженої оцінки в проектах розробки програмного забезпечення та пропонуються практичні рішення для зменшення ризиків, пов'язаних з експертною оцінкою. У дослідженні вико-

ристовується підхід case study та порівнюється одинарна оцінка з технікою оцінки та перегляду програми (PERT) для підвищення точності. Результати демонструють, що оцінка PERT надає діапазон оцінок із очікуваним результатом, а середня оцінка ближча до найкращого випадку, ніж початкові оцінки. Крім того, обчислюються довірчі інтервали, щоб забезпечити діапазон оцінок на різних рівнях довіри. Результати дослідження демонструють, що оцінка PERT із її діапазоном оцінок і розрахунком очікуваного випадку покращує точність оцінки порівняно з одинарною оцінкою. Крім того, дослідження розраховує середню відносну похибку (MRE) для вимірювання точності оцінок, що вказує на потенціал для постійного вдосконалення точності оцінок. Ця стаття робить внесок у сферу управління IT-проектами, підкреслюючи важливість методів експертного оцінювання, визначаючи проблеми, пов'язані з перевищенням і заниженням оцінки, а також надання практичних інструментів і методологій для підвищення точності оцінки. Завдяки мінімізації суб'єктивності та постійному вдосконаленню оцінок IT-проекти можна реалізувати вчасно та в рамках бюджету, що призведе до кращих результатів проекту та задоволення зацікавлених сторін.

Ключові слова: *керування проектами, оцінка часозатратності проекту, експертна оцінка, PERT, довірчі інтервали.*

Problem statement. In software development, accurate estimation is critical to the success of any software development project, as it helps in determining the timeline, budget, and resource allocation required for project completion [1;2]. However, despite the availability of various estimation techniques and tools, overestimation or underestimation of project tasks remains a common issue. Therefore, the issue of expert judgement in software estimation, the causes of over or underestimation, and provide practical tips for achieving accurate estimation in software development projects are the crucial topics to discuss and investigate in the field of IT projects management.

Overestimation or underestimation of project tasks can lead to project delays, budget overruns, and unsatisfactory outcomes, causing frustration for project stakeholders and affecting business outcomes. Expert judgement plays a crucial role in software estimation, as it involves the subjective assessment of project requirements, complexity, and uncertainty. Despite the availability of various estimation techniques and tools, overestimation or underestimation of tasks remains a common issue in software development projects. Therefore, understanding the causes of over or underestimation and learning how to avoid them can significantly improve project outcomes and ensure that software development projects are delivered on time and within budget.

Latest research and publications analysis. There is the wide of publications dedicated to software estimation techniques, with comparative analysis [3; 4], data driven approaches [5], and, obviously, containing practitioners perspective [6]. Still, the problem of over- and underestimation of time effort to complete IT projects and their sub-features is big challenges for project and product managers.

Aim of the research. The aim of this research is to examine the issue of over and underestimation in software development projects and provide practical solutions for reducing the risks associated with expert judgement software estimation.

Methodology. Expert judgement [7; 8] is one of the most common methods for software estimation. It involves seeking input from experienced professionals to arrive at an informed estimate of the work required to complete a task. However, despite its potential benefits, expert judgement can be fraught with challenges and limitations.

Expert judgement is the process of seeking input from a single expert (or a group of experts) to estimate a software development task. This approach can be effective when the task is well defined and the expert has relevant experience and knowledge. However, there are several limitations to individual expert judgement that must be considered.

Firstly, experts may have biases that can affect their estimates. These biases can arise from various sources, such as personal experience, beliefs, or preferences. For example, an expert who has experience with a particular technology may be biased towards overestimating tasks related to that technology.

Secondly, experts may be prone to cognitive biases that can affect their estimates. These biases can lead to over or underestimation of tasks. For example, experts may be influenced by the anchoring effect, where the first estimation they receive from colleagues anchors their subsequent estimates.

Thirdly, experts may have limited knowledge or experience in certain areas. This can lead to underestimation of tasks that require expertise in those areas.

Case study. Consider the following case: team has to estimate a new feature for an existing product. It is a mature product with more than 20 years of development history. Unfortunately, the team joined this product about 6 months ago. It means that the team already has some experience and historical data within this project, but not a lot. The unknown area is much bigger than known. Estimates should be provided within a day so there is no possibility to spend a lot of time on requirements, design, etc.

Estimation process. Firstly, the single-point estimation technique was used. Judgement estimation in its essence means that we need to get the experts in one place and ask them to estimate the feature, and this was done in this case.

One of the best ways to improve the accuracy of the feature estimate is to split it into smaller sub-features. This was done, taking into account the rule that every sub-feature should be done in 2 weeks or less by one developer. “Done” means that it should be developed, reviewed, merged, tested and as the result successfully closed. The results of estimation are presented in table 1.

Table 1

Project estimation using single-point estimation

Sub-feature	Estimated Days to Complete
Sub-feature 1	6
Sub-feature 2	4
Sub-feature 3	6
Sub-feature 4	10
Sub-feature 5	6
Sub-feature 6	6
Sub-feature 7	5
Total	**43**

Usually estimation ends on this stage. Some project managers add 30% as a ‘safe days’ and show this number to the stakeholders, which in our case would be about 56 days. During such estimation presentation to stakeholders they usually ask details how the current estimate was reached. And as soon as they understand that the original estimate was 43 days and 13 days were added by the manager, the last number is totally ignored and 43 days is considered as the commitment.

Single-Point Estimation does not provide any flexibility or ranges. To compare these results to more sophisticated technique, Program Evaluation and Review Technique (PERT) was used next [9;10;11]. It enables the calculation of an Expected Case that may not necessarily be in the middle of the range from the best case to the worst case.

After a short break previously estimated numbers were hidden from experts. Now they were asked to estimate the Best Case and the Worst Case. Best case means that everything will go smoothly and no problems should appear. Worst case means that everything will go wrong.

The results of estimation are presented in table 2.

Table 2

Project estimation using PERT technique estimation

Sub-feature	Estimated Days to Complete	
	Best Case	Worst Case
Sub-feature 1	5	10
Sub-feature 2	3	5
Sub-feature 3	3	10
Sub-feature 4	10	20
Sub-feature 5	5	10
Sub-feature 6	5	10
Sub-feature 7	5	10
Total	**36**	**75**

When original single-point estimates are compared to the Best Case and the Worst Case estimates, we can see that the original 43 days total is much closer to the Best Case estimate of 36 days than to the Worst Case of 75 days.

To find the Best Case and the Worst Case is a very important step, but it provides too wide range of possible estimates and the following question is still valid: “What estimate should be used?” The answer is none of the above, including original estimates. And the average estimate cannot be used too, because usually the Worst Case is much worse than what’s called the Expected Case. Taking the average of the ranges can result in an unnecessary high estimate.

To properly use PERT technique an additional Most Likely Case estimation is added, which is also estimated using expert judgement estimation.

The results after adding Most Likely Case estimation are presented in table 3.

Table 3

Project estimation using PERT technique estimation with Most Likely Case added

Sub-feature	Estimated Days to Complete		
	Best Case	Most Likely Case	Worst Case
Sub-feature 1	5	6	10
Sub-feature 2	3	4	5
Sub-feature 3	3	6	10
Sub-feature 4	10	15	20
Sub-feature 5	5	6	10
Sub-feature 6	5	6	10
Sub-feature 7	5	6	10
Total	**36**	**51**	**75**

As the result, the Most Likely Case is still closer to the Best Case of 36 days, but it is higher than the original estimates of 43 days.

According to the PERT we can use the following formula to calculate the Expected Case:

Expected Case = (Best Case + (4 x Most Likely Case) + Worst Case) / 6

This formula accounts for the full width of the range as well as the exposition of the Most Likely Case within the range. The Table 4 represents results from the previous table with the addition of Expected Case.

Table 4

Project estimation using PERT technique estimation with Most Likely Case added

Sub-feature	Estimated Days to Complete			
	Best Case	Most Likely Case	Worst Case	Expected Case
Sub-feature 1	5	6	10	6.5
Sub-feature 2	3	4	5	4.0
Sub-feature 3	3	6	10	6.17
Sub-feature 4	10	15	20	15
Sub-feature 5	5	6	10	6.5
Sub-feature 6	5	6	10	6.5
Sub-feature 7	5	6	10	7.83
Total	**36**	**51**	**75**	**52.50**

As a result the Expected Estimation is 52.50 days.

Although PERT estimation technique results provide more than just one number, they are still numbers with no confidence level. To do that we need to calculate Range first using the following formulas:

Lower Bound = Expected Case – (Range * Z)

Upper Bound = Expected Case + (Range * Z)

The Z-value [12] is based on standard normal distribution and it is different for different confidence levels.

Based on this data, Lower and Upper Bounds can be calculated for each confidence level. They will represent the range of estimates.

Table 5

Calculating Upper and Lower Bounds based on Z-value

Confidence Level	Lower Bound	Upper Bound
80%	52.5 – (6.5 * 1.282) ≈ 44,17	52.5 + (6.5 * 1.282) ≈ 60,83
85%	52.5 – (6.5 * 1.440) ≈ 43,14	52.5 + (6.5 * 1.440) ≈ 61,86
90%	52.5 – (6.5 * 1.645) ≈ 41,81	52.5 + (6.5 * 1.645) ≈ 63,19
95%	52.5 – (6.5 * 1.960) ≈ 39,76	52.5 + (6.5 * 1.960) ≈ 65,24
99%	52.5 – (6.5 * 2.576) ≈ 35,76	52.5 + (6.5 * 2.576) ≈ 69,24
99.9%	52.5 – (6.5 * 3.291) ≈ 31,11	52.5 + (6.5 * 3.291) ≈ 73,89

In IT project management, confidence level below 90% is considered too risky, everything above has too wide range. It means that with a 90% confidence level, the estimated effort falls within the range of approximately from 42 to 63 days. The range is still big (50% difference) and a lot of stakeholders usually ignore the second number, but detailed explanation of how those numbers occurred always helps.

Comparison of estimates to actual results. The Expected Case estimation of 52.50 days in addition to the range 42-63 days (with 90% confidence level) were provided to the stakeholders. Those are important numbers, but they mean a little without comparing it to the Actual Results. In addition Mean Relative Error (MRE) should be calculated using the following formula:

MRE = |(Actual Result – Expected Case) / Actual Result|

Table 6 represents expected and actual results, as well as MREs for all sub-features.

Table 6

Comparison of expected and actual results

Sub-feature	Estimated Days to Complete		
	Expected Case	Actual Results	MRE
Sub-feature 1	6.5	6	8%
Sub-feature 2	4.0	2	100%
Sub-feature 3	6.17	6	3%
Sub-feature 4	15	20	25%
Sub-feature 5	6.5	10	35%
Sub-feature 6	6.5	8	19%
Sub-feature 7	7.83	7	12%
Total	**52.50**	**59**	
Average MRE for sub-feature estimates			**29%**

Discussion of the results and conclusions. In the table 5, the MRE is calculated for each estimate. The average MRE is 29% for the set of estimates. It can be used to measure accuracy of estimates. As estimates improve, MRE should go down.

Expert judgement estimation is the most popular estimation in the world. Based on different researches from 70% to 90% of all project estimates were based on expert judgement estimation.

It is very common, but also very risky, because it provides too many subjective numbers. The following article aimed to minimise subjectiveness and, which is more important, provides a tool to constantly improve estimates.

References:

1. Steve McConnell. (2006). Software Estimation. Microsoft press.
2. Richard D. Stutzke. (2005) Estimating Software-intensive Systems: Projects, Products, And Processes. Addison-Wesley Professional.
3. Shekhar, S., & Kumar, U. (2016, May 17). Review of Various Software Cost Estimation Techniques. International Journal of Computer Applications, 141(11), 31–34. https://doi.org/10.5120/ijca2016909867
4. Aljohani, M., & Qureshi, R. (2017, November 30). Comparative Study of Software Estimation Techniques. International Journal of Software Engineering & Applications, 8(6), 39–53. https://doi.org/10.5121/ijsea.2017.8603
5. Alsaadi, B., & Saeedi, K. (2022). Data-driven effort estimation techniques of agile user stories: a systematic literature review. Artificial Intelligence Review, 55(7), 5485-5516.
6. Sandeep, R. C., Sánchez-Gordón, M., Colomo-Palacios, R., & Kristiansen, M. (2022, January). Effort estimation in agile software development: a exploratory study of practitioners' perspective. In Lean and Agile Software Development: 6th International Conference, LASD 2022, Virtual Event, January 22, 2022, Proceedings (pp. 136-149). Cham: Springer International Publishing.
7. Rae, A., & Alexander, R. (2017, November). Forecasts or fortune-telling: When are expert judgements of safety risk valid? Safety Science, 99, 156–165. https://doi.org/10.1016/j.ssci.2017.02.018
8. Jørgensen, M. (2007, July). Forecasting of software development work effort: Evidence on expert judgement and formal models. *International Journal of Forecasting*, *23*(3), 449–462. https://doi.org/10.1016/j.ijforecast.2007.05.008
9. Plebankiewicz, E., Juszczyk, M., & Malara, J. (2015, September 1). Estimation Of Task Completion Times With The Use Of The PERT Method On The Example Of A Real Construction Project. Archives of Civil Engineering, 61(3), 51–62. https://doi.org/10.1515/ace-2015-0024
10. Mohamed, M., Abdel-Basset, M., Hussien, A., & Smarandache, F. (2016). Using Neutrosophic Sets to Obtain PERT Three-Times Estimates in Project Management. ResearchGate. https://www.researchgate.net/publication/318673213_Using_Neutrosophic_Sets_to_Obtain_PERT_Three-Times_Estimates_in_Project_Management
11. Ba'Its, H. A., Puspita, I. A., & Bay, A. F. (2020). Combination of program evaluation and review technique (PERT) and critical path method (CPM) for project schedule development. International Journal of Integrated Engineering, 12(3), 68-75.
12. University of Arizona. Standard normal distribution: Z score. Retrieved June 23, 2023, from https://www.math.arizona.edu/~rsims/ma464/standardnormaltable.pdf