https://doi.org/10.31891/2307-5732-2025-353-18
УДК 004.412

LATANSKA LIUDMYLA
Admiral Makarov National University of Shipbuilding
https://orcid.org/0000-0001-6473-7624
e-mail: liudmyla.latanska@nuos.edu.ua
POLOVYNKA ANTON
Admiral Makarov National University of Shipbuilding
https://orcid.org/0009-0007-8648-135X
e-mail: anton.polovynka@nuos.edu.ua

ANALYSIS OF METHODS AND MODELS FOR ESTIMATING THE DURATION OF BANKING SOFTWARE DEVELOPMENT PROJECTS

This paper addresses the issue of inaccurate duration estimation in software development projects, with a particular focus on projects implemented in the banking sector. The relevance of the topic is substantiated by the high failure rate of such projects, a significant portion of which are not completed within the planned timeframe, resulting in substantial financial losses and reduced project management efficiency. The key causes of schedule deviations are analyzed, including the specific characteristics of banking software, such as regulatory constraints, architectural complexity, integration of numerous external services, high security requirements, and multi-level approval processes on the part of the client. It is clarified that traditional models, such as COCOMO and ISBSG, often fail to reflect the actual distribution of empirical data and industry-specific factors, which reduces the accuracy of estimation. The results of multivariate normality testing of project duration and effort distribution for banking software projects, based on a dataset of 482 real projects implemented between 2014 and 2024, are presented using Mardia's criteria. The data were found to deviate from the normal distribution. Based on the same dataset, six COCOMO models and three ISBSG models were built – both for the full dataset and for subgroups categorized by project type (single-team and multi-team). A comparative analysis of model quality showed that none of the resulting models achieved acceptable estimation accuracy based on MMRE and Pred(0.25) metrics. The study justifies the need to improve estimation approaches by developing specialized nonlinear regression models using data transformation techniques. These models would enhance estimation reliability and enable interval-based forecasting. The results are of practical value for analysts, project managers, and planning specialists involved in banking software development.

Keywords: banking software, estimation of software metrics, software development duration, software development effort, COCOMO, ISBSG.

ЛАТАНСЬКА ЛЮДМИЛА, ПОЛОВИНКА АНТОН
Національний університет кораблебудування ім. адмірала Макарова

АНАЛІЗ МЕТОДІВ І МОДЕЛЕЙ ДЛЯ ОЦІНЮВАННЯ ТРИВАЛОСТІ ПРОЄКТІВ З РОЗРОБКИ БАНКІВСЬКОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

У роботі розглянуто проблему недостовірності оцінювання тривалості проєктів з розробки програмного забезпечення, зокрема таких, що реалізуються в банківській сфері. Обґрунтовано актуальність теми з огляду на високий рівень невдач таких проєктів, значну частину яких не вдається завершити у встановлені строки, що спричиняє відчутні фінансові втрати та знижує ефективність управління проєктами. Проаналізовано основні причини відхилень у дотриманні термінів виконання робіт, зокрема специфіку банківського програмного забезпечення, включаючи вплив регуляторних вимог, складність архітектури, інтеграцію численних зовнішніх сервісів, високі вимоги до безпеки та багатоступеневі погодження з боку замовника. Уточнено, що традиційні моделі, зокрема COCOMO та ISBSG, часто не враховують реальні розподіли емпіричних даних та особливості галузі, що знижує достовірність оцінювання. Наведено результати перевірки нормальності багатовимірного розподілу тривалості та трудомісткості розробки банківського ПЗ на прикладі 482 проєктів, що реалізовувалися упродовж 2014–2024 років, за допомогою критеріїв Мардіа. Встановлено невідповідність нормальному закону розподілу для даної вибірки. Також за даною вибіркою побудовано шість моделей COCOMO та три моделі ISBSG, як для повних даних, так і для їх розрізів за типом проєкту (однокомандні та багатокомандні), проведено порівняльний аналіз їх якості, та показано, що жодна з отриманих моделей не досягає прийнятного рівня достовірності за показниками MMRE та Pred(0,25). Обґрунтовано необхідність удосконалення підходів до оцінювання тривалості проєктів з розробки банківського ПЗ, зокрема побудову спеціалізованих нелінійних регресійних моделей із застосуванням нормалізуючих перетворень, що дозволить підвищити достовірність та забезпечити можливість інтервального оцінювання. Результати роботи мають практичну цінність для аналітиків, проєктних менеджерів та фахівців з планування, що працюють у сфері розробки банківського ПЗ.

Ключові слова: банківське ПЗ, оцінювання метрик ПЗ, тривалість розробки ПЗ, трудомісткість розробки ПЗ, COCOMO, ISBSG.

Стаття надійшла до редакції / Received 27.04.2025
Прийнята до друку / Accepted 16.05.2025

Statement of the problem

Research in the field of software development shows that a significant portion of software projects are not completed within the planned timeframe. This leads to substantial financial losses for both customers and software developers. According to the Standish Group CHAOS Report 2020, only 31% of IT projects were completed successfully, 50% were finished with cost overruns, schedule delays, or reduced functionality, and 19% were completely failed, i.e., canceled at some stage of implementation [1]. This trend has remained virtually unchanged since the 1990s. A project is considered successful if its objectives are achieved within the defined constraints of scope, time, cost, and quality [2].

According to research in software project management, delays in implementation are the most common type of failure encountered by project teams. In most cases, these are not just due to technical difficulties – organizational, managerial, or communication issues are often to blame.

Therefore, the accuracy of duration estimation within project time management is one of the key factors determining the success of software engineering projects and contributing to their timely completion.

Accurate estimation is traditionally considered one of the most difficult stages of software development. The main reasons for this complexity include:

– the lack of a reliable historical database with prior estimates;

– the high dependency of software development on numerous interrelated factors, many of which are still not fully understood.

For each application domain, including banking software, these factors are unique and should be considered to improve the reliability of further analysis.

Analysis of research and publications

Most traditional banks still rely on outdated software systems that have limited flexibility, complex integration mechanisms, and high maintenance costs. These systems hinder the ability to respond quickly to market demands, launch new digital products, and scale technological solutions efficiently.

According to recent surveys, 46% of banking executives consider legacy systems to be the main obstacle to the growth of commercial banks. On average, large incumbent banks operate with more than 20 legacy software systems, many of which were developed decades ago. Modifying these systems becomes increasingly difficult over time, as each newly added function complicates future migration efforts [3].

Such architectures present a significant barrier to innovation and necessitate a fundamental transformation of IT infrastructure. This includes adopting a modular approach to software system development, transitioning to cloud technologies and implementing microservice architecture. These solutions allow banks to achieve the necessary speed and adaptability [4, 5]. Cloud platforms provide scalable computing resources, reduce operational costs, and improve risk management. Meanwhile, microservice architecture allows the decomposition of complex banking systems into autonomous components, each of which can be developed, tested, and deployed independently. This significantly simplifies change management, accelerates the delivery of new functionality, and is one of the key enablers for effective cloud migration.

These changes are not only technical but also strategic in nature, since without their implementation banks risk losing their competitive edge against fintech companies that are originally built to operate in flexible digital environments. As of 2022, approximately a quarter of core banking operations were hosted on cloud services, and 14% of overall banking functionality was delivered through public cloud services or cloud-based platforms [6].

Banking software projects also have a range of additional specific characteristics that distinguish them from software development projects in many other domains. These features directly impact project planning complexity, the accuracy of duration estimation, and the actual ability to adhere to project timelines.

The most common causes of delays in implementing financial IT solutions include the complexity of regulatory requirements, legacy system architecture [4, 6], and a high density of integrated services, including external service integrations [7]. Additionally, high security standards, elevated levels of formalization, and longer decision-making cycles typical of banks—compared to general commercial development – also contribute to increased project duration.

An analysis of modern approaches to estimating the duration of software development projects has shown that probabilistic models remain among the most widely used. These include beta distributions within the PERT method and nonlinear regression models that determine project duration as a function of effort, such as the COCOMO model [8, 9] and models built using the ISBSG database [10, 11]. However, these approaches have limitations that affect estimation accuracy and reduce reliability when applied to banking software development projects.

In many cases, the empirical distributions of project duration and effort deviate from the normal distribution. [11–13]. This prevents the effective use of models that rely on the assumption of normality for constructing confidence intervals of duration estimates. In such cases, either the distribution is artificially assumed to be normal, or nonparametric methods are applied [14]. Both options negatively affect the reliability of duration estimates in software projects.

Therefore, the development of models that improve the reliability of duration estimation for banking software development projects – and the creation of information processing technologies based on such models – remains a relevant and practically valuable task.

Formulation of the article's objectives

The aim of this study is to analyze approaches for improving the reliability of duration estimation in banking software development projects and to justify the need for further research in this area.

To achieve this goal, the following objectives have been set:

- To analyze the main causes of discrepancies between the planned and actual completion dates of banking software development projects.

- To identify the specific features of duration estimation in banking software projects.

- To examine the existing methods and models used for estimating the duration of software development projects.

- To determine ways to enhance the reliability of duration estimation in banking software development.

- To substantiate the necessity of further research on the estimation of project duration in the context of banking software development.

Presentation of main material

Estimating the duration of software development projects has been the subject of numerous scientific studies and applied developments. The range of available estimation methods continues to expand, contributing to the creation of models of various types and levels of complexity. There are a number of methods and expert systems for determining the duration of software development projects, including expert judgment, analogy-based estimation, parametric estimation, the three-point estimation method (PERT), bottom-up estimation, reserve analysis, and others [15].

Parametric estimation methods are based on mathematical models that describe the relationship between project parameters and estimated characteristics such as project duration and effort. Input parameters may include the software size (measured in function points or lines of code), hardware and software environment, level of team and management experience, and the development methodology used. These models generate estimates using empirical data collected from past projects and observed patterns [8, 16].

The advantages of parametric methods lie in their reproducibility, transparency, and the ability to justify estimates. They are particularly useful for large or critical projects, where formalized and documented estimation is important. However, these methods require access to a high-quality historical project database, as well as precise specification of current project parameters, which may be difficult at the early planning stages [17].

The most widely known parametric models include COCOMO and COCOMO II, which estimate effort and duration based on code size and a set of modifying factors [8, 9], and the ISBSG model, developed as part of an international initiative maintaining the largest independent empirical software project dataset. A nonlinear regression model derived from this data allows for the estimation of project effort and duration [10]. These models calculate effort E in person-months and duration D in calendar months based on project size in thousands of lines of code (KLOC) and a set of coefficients that account for various factors such as personnel competence, software complexity, platform type, etc.

COCOMO further distinguishes between three project types: organic (small-scale projects, 2–50 KLOC), semi-detached (medium-complexity projects, 50–300 KLOC), and embedded (large-scale projects, over 300 KLOC). For each type, COCOMO offers a separate empirical formula for estimating duration based on effort [8, 9]:

– for the organic type:

$$D = 0.371 \cdot E^{0,38}, \tag{1}$$

– for the semi-detached type:

$$D = 0.431 \cdot E^{0,35}, \tag{2}$$

– for the embedded type:

$$D = 0.501 \cdot E^{0,32},$$

where: D – the project duration in months, E – the effort in person-months.

To estimate duration based on project effort using the ISBSG model, the following formula is used [10]:

$$D = 0.662 \cdot E^{0,328}, \tag{3}$$

where: D – the duration in months, E – the effort in person-hours.

Both models apply logarithmic transformation to normalize empirical data on project duration and effort. However, in many cases, this transformation does not effectively normalize the data, thus creating the need for alternative normalizing transformations.

Therefore, the first step in the analysis is to verify whether the available data on the duration and effort of banking software development projects follow a normal distribution. If the data deviate from the normal distribution, it becomes impossible to build reliable linear regression models, and the use of nonlinear regression models with normalizing transformations becomes necessary.

In such a case, it is reasonable to initially apply existing nonlinear regression models such as COCOMO and ISBSG to the available data and assess the reliability of the results. However, as these models rely on logarithmic transformation – which does not always adequately normalize empirical data – the reliability of duration estimates may be compromised. Additionally, banking software development has specific characteristics that significantly influence project timelines and differ substantially from other types of software projects.

Thus, if the accuracy indicators of the existing regression models prove to be unsatisfactory for current data on banking software projects, it will be necessary to enhance the reliability of project duration estimation. To achieve this, one or more specialized nonlinear regression models should be developed using normalizing transformations that better reflect the distribution of project duration and effort data in banking software development. These models will also support confidence interval estimation, allowing the uncertainty of point estimates to be taken into account. The application of these methods will enable the development of more reliable models by taking into account the empirical distribution characteristics of project duration and effort (in particular, skewness and kurtosis), and will also provide the ability to construct confidence intervals for the duration of banking software development projects.

Anonymized data from 482 real-world banking software development projects implemented between 2014 and 2024 were used as empirical data for the analysis and for exploring ways to improve the reliability of project duration estimation. The dataset included both backend development projects written in Java and frontend development projects using Angular and React frameworks. Project effort ranged from 40 to 2080 person-days, and project durations ranged from 37 to 1722 calendar days. The number of developers involved in each project varied from 4 to 17. Projects with 4 to 6 developers were considered single-team projects, while those with larger teams were treated as multi-team projects.

This classification was important due to the substantial differences between single-team and multi-team banking software projects. In multi-team development, complexity increases significantly compared to projects handled by a single team. These challenges are caused by the need to coordinate between teams, synchronize release schedules, align development methodologies and designs, and conduct integration testing, which often requires separate test environments. Moreover, such projects demand additional planning phases and the involvement of a project manager to monitor execution timelines and make necessary adjustments.

Verification of the empirical project data for multivariate normality was carried out using Mardia's criteria, based on multivariate skewness β_1 and multivariate kurtosis β_2 [18]:

$$\hat{\beta}_1 = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left\{(X_i-\bar{X})^T S_N^{-1}(X_j-\bar{X})\right\}^3, \quad (4)$$

$$\hat{\beta}_2 = \frac{1}{N}\sum_{i=1}^{N}\{(X_i-\bar{X})^T S_N^{-1}(X_i-\bar{X})\}^2, \quad (5)$$

where: N – the number of observations, X_i – the i-th observation vector, $\bar{X}$ – the sample mean vector, S_N^{-1} – the inverse of the covariance matrix.

For $\hat{\beta}_1$ was used the test statistic:

$$\frac{N}{6}\hat{\beta}_1, \quad (6)$$

which approximately follows a chi-squared (χ^2) distribution with $k(k+1)(k+2)/6$ degrees of freedom and a significance level of $\alpha = 0.005$, where k – the number of variables in the multivariate distribution.

If the calculated values of β_1 test statistic and β_2 exceed the corresponding quantiles, this indicates that the empirical data do not follow a multivariate normal distribution.

The test was conducted both for the full dataset and separately for two groups based on team size: single-team projects and multi-team projects. The results of the multivariate normality test for project duration and effort data in banking software development are presented in Table 1.

Table 1

Multivariate Normality Test Results for Empirical Data

Indicator	All Projects		Single-Team Projects		Multi-Team Projects	
	Value	Quantile Value	Value	Quantile Value	Value	Quantile Value
β_1 statistic by (4), (6)	2539.5269	14.8603	442.4637	14.8603	853.7690	14.8603
β_2 by (5)	60.8453	8.3420	21.5726	8.6197	42.2105	8.7632

As seen in the table, none of the calculated values were below the corresponding quantile. This leads to the conclusion that the empirical data do not follow a multivariate normal distribution. Therefore, to improve the reliability of duration estimation for banking software development projects, it is necessary to consider nonlinear regression models. First, we examine the existing nonlinear regression models COCOMO and ISBSG.

Model quality assessment was performed using the following indicators: Mean Magnitude of Relative Error (MMRE) and prediction level Pred(0.25) [19]. To be considered satisfactory, a model should have an MMRE not exceeding 0.25 and a Pred(0.25) not lower than 0.75.

Based on the available empirical data, three groups of COCOMO models were constructed: using data from all projects, from single-team projects, and from multi-team projects. Each group considered the classification of projects into organic (formula (1)) and semi-detached (formula (2)) types, based on the number of lines of code. In total, six models were built, with the results presented in Table 2. Similarly, three ISBSG models were constructed using formula (3), and the results are shown in Table 3.

Table 2

Model Quality Indicators for COCOMO

Quality Indicator	**All Projects**		**Single-Team Projects**		**Multi-Team Projects**	
	Organic	Semi-Detached	Organic	Semi-Detached	Organic	Semi-Detached
MMRE	0.5429	0.2636	0.5741	0.2285	0.4090	0.2734
Pred(0,25)	0.2056	0.5248	0.1849	0.6129	0.2941	0.5000

Table 3

Model Quality Indicators for ISBSG

Quality Indicator	**All Projects**	**Single-Team Projects**	**Multi-Team Projects**
MMRE	0.6542	0.8704	0.3885
Pred(0,25)	0.2928	0.1751	0.4375

According to the results, only one model (COCOMO, single-team, semi-detached) had an MMRE value below 0.25, while all Pred(0.25) values were significantly below 0.75. This indicates that the quality of the constructed COCOMO and ISBSG models is insufficient for reliably estimating the duration and effort of banking software development projects based on the analyzed empirical data.

Accordingly, this creates the need to develop one or more specialized probabilistic models that can account for the specific characteristics of the project data and ensure the required reliability of duration estimation for banking software development projects. These models should also enable the construction of confidence intervals for project duration. To build nonlinear regression models and corresponding confidence intervals, it is advisable to apply methods based on normalizing transformations of the existing empirical data [20, 21].

Conclusions

This study addressed the important task of improving the reliability of duration estimation for banking software development projects and substantiating the need for further research in this area.

It has been shown that the majority of software development projects are not completed within their initially planned timelines. The key reasons for such delays have been analyzed, and specific features of banking software development projects have been identified. These features significantly distinguish them from typical commercial projects and have a considerable impact on their timelines.

It was determined that one of the main factors affecting the successful completion of banking software development projects is the reliability of duration estimation. The accuracy of such estimation largely depends on the models and methods applied.

A dataset of empirical data from real banking software development projects was collected and analyzed to assess the normality of distributions. The results showed that the distributions deviate from normality, which prevents the effective use of models based on the assumption of normality—particularly for estimating confidence intervals of project duration.

Nonlinear regression models – COCOMO and ISBSG – were constructed based on various subsets of the empirical data. However, the quality indicators of these models showed that they are not sufficiently reliable for estimating the actual duration and effort of banking software projects.

As a result, the need to develop specialized regression models that can account for the specific characteristics of the project data and ensure the required reliability of duration estimation for banking software development projects has been substantiated. Possible methods for constructing such models have also been identified.

References

1. Mersino, A. (2020). *Project managers fail to help software projects (Standish Group Chaos 2020)*. Medium. https://medium.com/leadership-and-agility/project-managers-fail-to-help-software-projects-standish-group-chaos-2020-e65d803e99f2
2. Project Management Institute. (2021). *A guide to the project management body of knowledge* (7th ed.). Project Management Institute.
3. Khropatyy, P. (n.d.). *Modernizing legacy banking systems: How to get it right*. Intellias. https://intellias.com/modernizing-legacy-banking-systems-how-to-get-it-right/
4. Kot, P., Schöbl, S., & Soller, H. (n.d.). *How banks can achieve next-generation legacy modernization*. McKinsey & Company. https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/tech-forward/how-banks-can-achieve-next-generation-legacy-modernization
5. Kot, P., Schöbl, S., & Soller, H. (n.d.). *Should US banks be moving to next-generation core banking platforms?* McKinsey & Company. https://www.mckinsey.com/industries/financial-services/our-insights/should-us-banks-be-moving-to-next-generation-core-banking-platforms
6. Fareniuk, N. (2022). Perspektyvy vykorystannia khmarnykh tekhnolohii dlia tsyfrovykh bankiv. *Visnyk Kyivskoho natsionalnoho universytetu imeni Tarasa Shevchenka*, (56), 56–60.
7. Stfalcon. (n.d.). *Podolannia vyklykiv u rozrobtsi fintekh-prohramnoho zabezpechennia: Ekspertni stratehii na 2025 rik*. https://stfalcon.com/uk/blog/post/challenges-in-fintech-software-development
8. Boehm, B. W. (1981). *Software engineering economics*. Prentice Hall.
9. Boehm, B. W. (2000). *Software cost estimation with COCOMO II*. Prentice Hall.
10. International Software Benchmarking Standards Group. (2020). *ISBSG data repository release 2020*. https://www.isbsg.org
11. Elyassami, S., & Idri, A. (2012). *Investigating effort prediction of software projects on the ISBSG dataset*. arXiv. https://arxiv.org/abs/1204.2404
12. Flyvbjerg, B., Budzier, A., Lee, J. S., Keil, M., Lunn, D., & Bester, D. (2022). *The empirical reality of IT project cost overruns: Discovering a power-law distribution*. arXiv. https://arxiv.org/abs/2210.01573
13. Bosu, M. F., MacDonell, S. G., & Whigham, P. A. (2021). *Analyzing the stationarity process in software effort estimation datasets*. arXiv. https://arxiv.org/abs/2107.01616
14. Wasserman, L. (2006). *All of nonparametric statistics*. Springer.
15. Project Management Institute. (2017). *A guide to the project management body of knowledge (PMBOK® guide)* (6th ed.). Project Management Institute.

16. Jørgensen, M., & Shepperd, M. (2007). A systematic review of software development cost estimation studies. *IEEE Transactions on Software Engineering*, *33*(1), 33–53. https://doi.org/10.1109/TSE.2007.256943

17. McConnell, S. (2006). *Software estimation: Demystifying the black art*. Microsoft Press.

18. Mardia, K. V. (1970). Measures of multivariate skewness and kurtosis with applications. *Biometrika*, *57*(3), 519–530. https://doi.org/10.1093/biomet/57.3.519

19. Prykhodko, S. B., Makarova, L. M., Prykhodko, N. V., & Pukhalevych, A. V. (2023). *Metodychni vkazivky ta zavdannia do vykonannia laboratornykh robit z dystsypliny "Empirychni metody prohramnoi inshenerii"*. NUK.

20. Prykhodko, S. B. (2007). Intervalne otsiniuvannia parametriv stokhastychnykh dyferentsialnykh system na osnovi modyfikatsii uzahalnenoho metodu momentiv. In *Materialy XIII Mizhnarodnoi konferentsii z avtomatychnoho upravlinnia* (pp. 69–75). UNIVERSUM-Vinnytsia.

21. Prykhodko, S. B. (2011). Intervalne otsiniuvannia statystychnykh momentiv nehausivskykh vypadkovykh velychyn na osnovi normalizuvalnykh peretvoren. *Matematychne modeliuvannia*, (1(24)), 9–13.