Importance of Addressing Complexity Factors to Improve Project Success Rates and the Overall Effectiveness of Software Development

Ajay Jaiswal[1], Jagdish Raikwal[2] and
Pushpa Raikwal[3]

[1] Prestige Institute of Engineering Management and Research, Indore (MP), India
[2] Institute of Engineering and Technology, Indore (MP), India
[3]PDPM-IIITDM, Jabalpur (MP), India.
ajay.jaiswal55555@gmail.com

Abstract. Software development is a complicated process that is critical to improving technology and promoting business innovation. Project management methodologies, team dynamics, stakeholder participation, technical innovations, and organizational context all impact success. Despite high standards in these areas, many projects encounter difficulties such as missed deadlines or cost constraints. To address these concerns, this study looks at team performance and overall project success in Sri Lanka's fast-growing IT sector. The study used a mixed-method approach to determine complexity features by analyzing quantitative data from the Kitchenham, Maxwell, and Desharnais datasets. To identify essential factors influencing project out- comes, the Lasso feature selection method was applied. Fuzzy logic was utilized to develop and test an ensemble classifier that demonstrated exception- al accuracy, particularly on the Kitchenham dataset, where it achieved the lowest Root Mean Squared Error (0.2707) and the highest R-squared error (0.9308). The findings highlight the importance of addressing complexity factors in order to improve project success rates and overall software development effectiveness, and they propose that sophisticated techniques such as fuzzy logic-based cost estimation can significantly improve the accuracy and reliability of software project predictions.

Keywords: Software Development, Cost Estimation, Project Factors, and Project Outcomes.

1 Introduction

Modern technological advancement and corporate innovation rely substantially on the complex and diverse process of software development [1]. Project management tactics, team dynamics, stakeholder engagement, technology, and organizational conditions all help software development projects succeed [2]. Waterfall models and agile procedures are examples of effective project management solutions that affect development flexibility and efficiency. Team dynamics, which include cooperation, communication, and skill levels, are all important in deciding project success. Aligning project outcomes with user expectations also requires stakeholder

S. Bhalerao et al. (eds.), *Proceedings of the International Conference on Recent Advancement and Modernization in Sustainable Intelligent Technologies & Applications (RAMSITA-2025),* Advances in Intelligent Systems Research 192,
https://doi.org/10.2991/978-94-6463-716-8_55

involvement and the ability to provide clear objectives and feedback. Platforms, development tools, and upcoming technologies all influence the project's capacity to satisfy present and future objectives [3]. Project Size, Model size, Branching and, and project factors all influence software development success. Causes and Interrelationships and their impact on software development project success or failure [4]. The information technology (IT) sector significantly boosts company performance globally by fostering innovation and productivity in nearly every economic area. IT is one of the industries with the greatest rate of growth, assisting both established and developing nations in creating new goods and gaining access to new partners, investors, and markets. The potential of IT applications in business is enormous and continues to grow [5]. A large fraction of IT projects globally struggles to succeed, with 32% reporting low success rates, 44% exceeding budget and time restrictions, and 24% failing [6]. It uses a mixed-method approach to analyse data from various datasets, using Lasso feature selection to identify key factors. Coopers studied 10,640 projects from 200 businesses in 30 countries and found that 2.5% accomplished all of their obligations. This suggests that effective project completion remains a tough issue and that software development is essentially a high-risk endeavor. While many businesses use tight processes and quality control methods to maintain project quality. Specifically, existing tools and procedures frequently focus on the rational aspects of project management while ignoring the emotional factors that have a considerable impact on project success.

2 Literature Survey

San Cristóbal, J. R. et al. [7] studied projects is becoming more concerning as projects become more complicated. Complexity influences decision-making and goal achievement, thus project managers must understand and manage it. Complexity can make it difficult to identify goals and objectives, choose project organizational forms, and achieve project outputs. Understanding project complexity concepts, factors, types, and models will benefit the global project management community.

Samara Singhe, Samantha et al. [8] studied software development team performance as critical in IT initiatives. It has been found that communication, cohesion, mutual support, expertise coordination, trust, and diversity affect software development team performance. Previous research cannot be applied directly to Sri Lanka due to cultural and behavioral differences between Sri Lankans and Westerners.

Ibraigheeth, Mohammad & Fadzli, Syd. [9] described Software projects that meet quality, cost, schedule, and effort expectations are successful. Several software project reports and case studies to pinpoint the most typical success factors. In addition, case studies of software projects that succeeded and failed are explained. It has been developed to lower the likelihood of failure and raise the success rate of software projects.

Bogopa, M. E., et al. [10] described software development initiatives fail at any point, regardless of approach. Process variables like an unreasonable budget and schedule or technical considerations like methodology might affect project success or failure. The most important success variables (people, process, and technical) affect South African software projects. The vast majority of perceived factors are nontechnical. Table 1 compares reviews by different authors and identifies features.

2.1 Influencing Factors: People and Society

This section investigates the influences by looking at the roles and interactions of the many persons and organizations involved in the system. Typical jobs in software development include developers, development managers, business or user managers, project managers or leaders, team members, trainers, sponsors, vendors, customers, senior management, and external consultants [11]. To enhance our understanding and conceptualization of various aspects of systems development, the following discussion aims to offer a more accurate categorization of the different stakeholders involved in a system.

2.2 Influencing Factors – Project Content

This section addresses only a few variables that directly influence software development and deployment initiatives. These factors include the project's attributes, objectives (Table 1).

Table 1. Features of the impacts and qualities connected to the project content

Characteristics	Attributes
Project features:	Size, Complexity, the Company's People
Scope, Objectives, and Outcomes	Goal alignmentwith the business, appropriate, achievable & consensus, insight, and discourse
Resources Technology	Development duration, funds, and personnel
Technology	Technology and tools for development, Form, quality, availability of the information, Program modification level; novel or unproven technology

2.3 Influencing Factors – System Development Procedure

This section discusses the various factors that impact different components of the systems development process (Table 2). It focuses on key areas such as requirements gathering, project management, the application of a consistent methodology, user involvement in system development, user training, and managing changes resulting from system design and implementation.

Table 2. Factors affecting software cost

Factors	Description
Market Possibilities	A company that develops software may offer low prices to get into a new software market. By taking a small profit on one project, the business could be able to turn a bigger profit later.
Cost Estimate Uncertainty	If a company is unsure about its cost estimate, it may increase costs over and above its typical profit.
Conditions of Contract	The developer may be permitted to utilize the source code in future projects if the client consents to give up the rights to it.
Need Unpredictability	An organisation may choose to reduce its costs in order to secure a contract when it is expected that the criteria will change.
Equitable Finances	A developer may reduce their requested pricing to secure a job if they are having financial difficulties.

2.4 Influencing Factors – Institutional Context

Several authors have suggested that the organizational context can unexpectedly influence the outcomes of systems development and deployment projects. This influence is shaped by the broader socioeconomic and historical framework within which the organization operates (Table 3). Productivity estimations are usually obtained by examining the program's qualities and constraining the resulting figures by the whole amount of effort needed for development [12]. Software productivity is often measured in lines of source code per person-month, or LOC/pm (Table 4).

Table 3. Factors affecting software engineering productivity

Factors	Description
Experience in the application domain	Effective software development requires an understanding of the application domain.
Process Excellence	The selection of a development approach can significantly affect output.
Project Size	The more complex the project appears; the more time it takes for team communication.
Technology support	Effective support technologies, including configuration management systems and CASE tools, can boost productivity.

The post-architecture model (Fig.1) uses four feature categories to modify original estimates and offer multipliers:

3 Methodology

Subsequently, a fuzzy logic-based ensemble classifier was developed and tested against unutilized data to ensure its generalizability and reliability in predicting software project costs and performance metrics.

Examination of the Complexity Factors in Software Projects

The complexity of “software analysis, maintenance, testing, design, and implementation” is influenced by the software's complexity [13] [14] [15]. Complexity in software project management refers to a wide range of project management challenges.

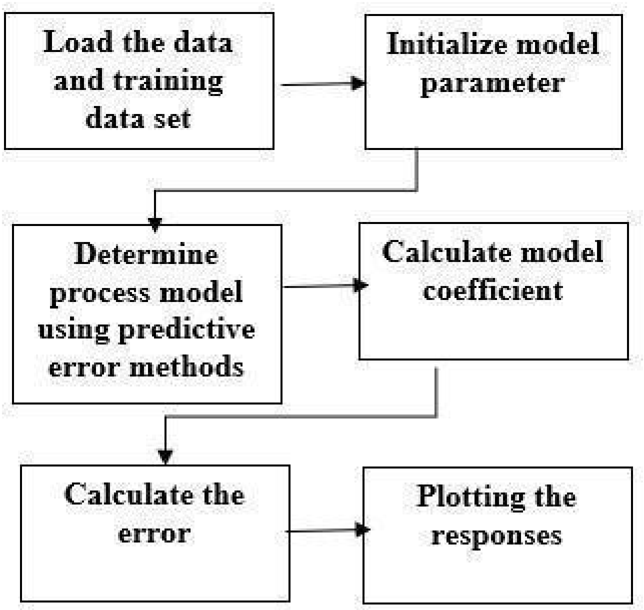
Load the data and training data set
Initialize model parameter
Determine process model using predictive error methods
Calculate model coefficient
Calculate the error
Plotting the responses

Fig. 1: Proposed architecture for software project cost estimation

To enhance the proportion of effectively completed initiatives, it is vital to comprehend the contributing elements (Table 4) and the corresponding procedures to mitigate them.

Table 4. Complexity Factors of IT Projects

Complexity Factors	Factors
Goal, Conditions, and Anticipations	Specific and changing strategic goals, realistic expectations, and precise requirements articulation of Regulation requirements.
Interested Parties, Integration, Stakeholder	User participation, support from upper management, and agreement with the project's methodology from the project sponsor.
Options, Collaboration, and Management	The technical and business components of initiatives are communicated to the other participants and the development team as a whole.
PM Tools, Methods, and Strategies	Iterative or incremental methods are used in projects.

Technology	Implementing new technology into practice and helping IT initiatives

- **Dataset:** The dataset for which feature selection was done is indicated in this column. The Kitchenham, Maxwell, and Desharnais datasets are the three that are cited.

- **Selected Features:** The following column provides the attributes (factors or characteristics) selected for each dataset using the Lasso feature selection method. According to (Table 5), these features are more relevant or helpful for the modeling or analysis activities associated with each dataset.

Table 5. Selected Features using LASSO

Dataset	Selected Features
Desharnais	Project, TeamExp, Transactions, PointsNonAdjust, PointsAjust, Effort
Maxwell	App, Har, Year, Source, Nlan, T05, T09, T15, Duration, Size, Time, Effort
Kitchenham	Clientcode, Project type, Duration, Adjfp, Estimate, Effort

Metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) are used to assess the ensemble classifier's performance and compare it to other classifiers like Random Forest (RF).

MRE (Magnitude of Relative Error): First, identify each data point's Magnitude of Relative inaccuracy, which quantifies the extent of estimating inaccuracy in a single estimate. This stage is used to calculate PRED (n) and acts as a model for the next one. Results that are good are indicated by a figure of 25% or less.

$$\text{MRE} = |\text{predicted} - \text{actual}| / \text{actual} \tag{1}$$

MRE (mean magnitude of the relative error): The average proportion of the absolute values of the relative errors throughout the whole data set is known as the mean magnitude of the relative error, or MMRE.

$$\text{MRE} = (100/N) * \sum/|\text{predicted}_i - \text{actual}_i| / \text{actual}_i \tag{2}$$

where, N = total number of estimates

PRED (n) Prediction Accuracy: A model should also be correct 75% of the time, to within 25%. To find the accuracy rate PRED(n) (represented by n), divide the total number of data points in a data set with an MRE of 0.25 or less (represented by k) by the total number of data points in the data set. With n = 0.25, the resulting equation is PRED (n) = k/n. The average proportion of estimates that were within n percent of the actual values is typically displayed by PRED(n). For example, PRED (42) = 50% means that half of the estimates are within 30% of the actual if there are N datasets.

$$\text{PRED}(x) = (100/N) *\Sigma i\ldots\ldots N\ 1 \text{ if } MREi <= n/100,\ 0 \text{ otherwise} \qquad (3)$$

The estimations' accuracy directly relates to PRED(x) and, in the opposite direction, to MMRE [16].

Table 6. Parameters and values on different Datasets

Dataset	Parameters	Values
	Enter Team Exp.	2
Desharnais: - fuzzy logic	Enter Manager Exp.	4
	Estimated Cost	51.085
	Cost Estimation	Medium
	Enter the value for App	InfServ
	Enter the value for Har	PC
	Enter the value for Dba	GUI
	Enter the value for UI Source	Outsrced
	Enter the value for Telonuse	Yes
	Enter the value for Nlan	1
Desharnais: - fuzzy logic	Enter the value for T01	1
	Enter the value for T02	1
	Enter the value for T03	1
	Computed Effort	32.7450
	Effort Classification	low
	Enter Actual Duration	12
Kitchenham: - fuzzy logic	Enter Adjusted Function Point	21
	Enter First Estimate	32
	Defuzzified Actual Effort	2676.93

When creating real software projects, the best fuzzy logic- based cost-estimating model is applied (Table 6). Before being used for real-world software project cost prediction, the resulting ensemble classifier is tested on fresh unutilized data to guarantee its generalizability and dependability. It describes different kinds of errors calculated for each model (Table 7).

Table 7. Error Matrix on Different Datasets

Datasets	Ensemble R-^{2} Error	Ensemble MAE	Ensemble RMSE	R^2 Error	MAE	RMSE
Desharnais	0.798	0.241	0.336	0.736	0.293	0.383
				0.777	0.352	0.262
				0.671	0.331	0.428
				0.803	0.222	0.331
				0.607	0.588	0.483
				0.239	0.668	0.818
Maxwell	0.507	0.521	0.521			

				0.023	0.685	0.927
				0.446	0.537	0.697
				0.910	0.308	0.249
				0.928	0.308	0.249
Kitchenham	0.930	0.270	0.194	0.858	0.387	0.306
				0.928	0.274	0.202

4 Results

Various complexity factors, such as goal alignment, stakeholder involvement, and the integration of new technologies significantly influence the outcome of software development projects. Understanding and minimizing these complexity elements is critical to increasing project success rates. Projects can achieve improved accuracy and dependability by implementing advanced methodologies such as fuzzy logic-based cost estimating, as evidenced by the ensemble model's performance across various datasets, notably in terms of error minimization and prediction precision. The "Desharnais", "Kitchenham", and "Maxwell" datasets are used in the recommended method for software project cost estimation. These figures offer historical information regarding software initiatives. This data covers a number of areas, including team experience, actual costs, and project size and complexity. A number of pre-processing tasks, including correcting missing values and scaling features, are completed to prepare the data for use. Fuzzy rules are developed using language variables, such as "low", "medium", and "high" for project size. These rules are derived from expert knowledge and historical data. Fuzzy inference rules are created by combining these fuzzy rules using the AND, OR, and NOT operators. [17].

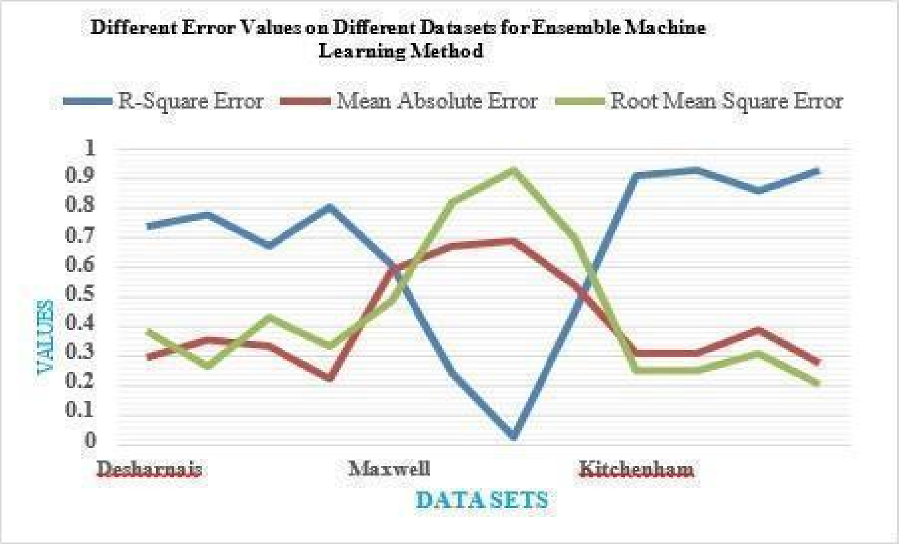

Fig. 2: Different Error Values on Different Datasets for Ensemble ML Method

The result graphs (Fig.2 and Fig. 3) show how, in comparison to the graph created for the current technique, the proposed technique's result graph runs far closer to the real curve. The outcomes of the proposed technique motivate us to declare that it is appropriate for estimating software development efforts in software organizations. For every model, the following metrics were calculated: R- squared error, Root Mean Squared error, and Mean Absolute error. These are shown in the Fig.4. The results show that the SVM model achieved the maximum R-squared error of 0.8037751 in the Desharnais dataset, while the RNN model obtained the minimum at 0.67150729.

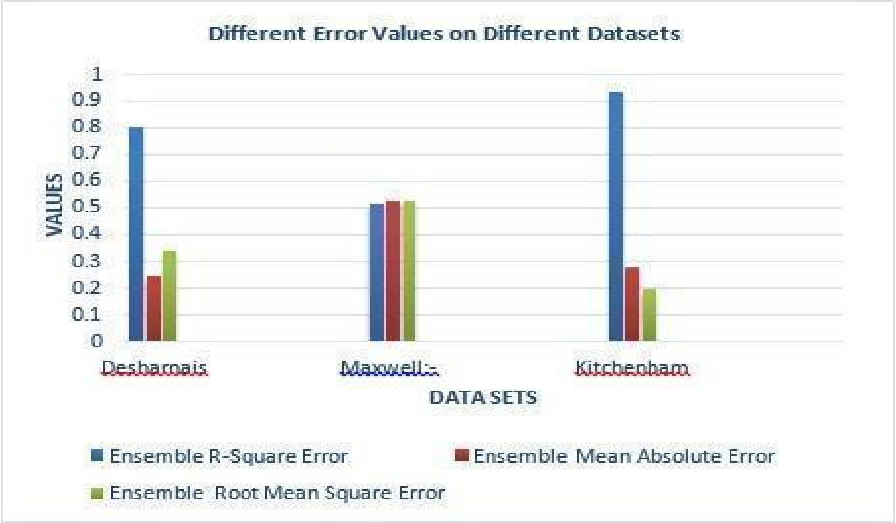

Fig. 3: Different Error Values on Different Datasets

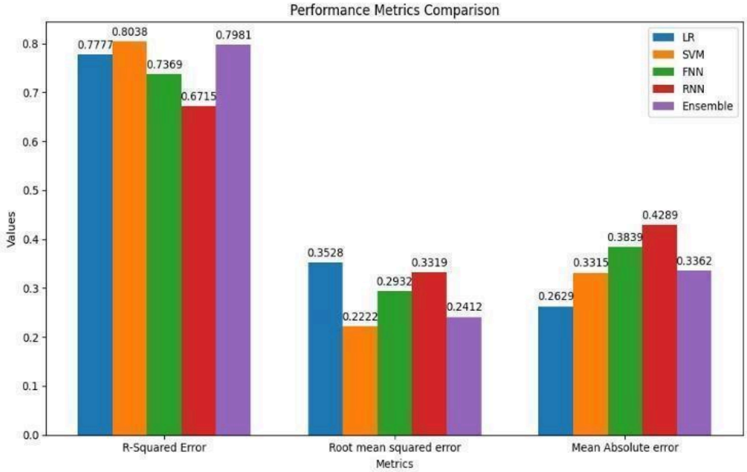

Fig. 4. Model Error Comparison of Desharnais Dataset

The RNN model had the lowest R-squared error for the Maxwell dataset (0.0237821), while the LR model had the worst at 0.6073169. Lastly, the RNN model showed the highest value of 0.3067252 for the Mean Absolute Error in the Kitchenham dataset, whilst the Ensemble model showed the lowest value of 0.1941534.

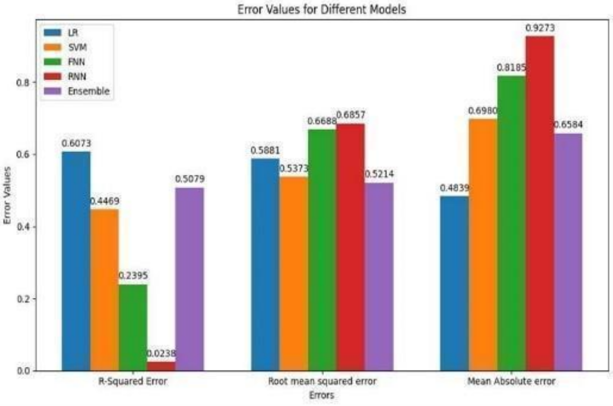

Fig. 5. Model Error Comparison of Kitchenham Dataset

The performance metrics for a number of machine learning models—LR, SVM, FNN, RNN, and Ensemble—applied to the Maxwell, Desharnais and Kitchenham Dataset are shown graphically in respectively (Fig.5 and Fig. 6). The outcome graphs demonstrate that, in comparison to the curve created for the current technique, the suggested technique's result curve runs comparatively considerably closer to the real curve. The outcomes of the suggested technique motivate us to declare that it is appropriate for estimating software development efforts in software organizations.

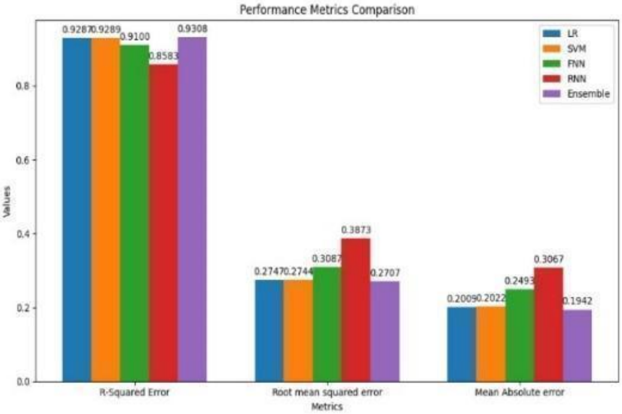

Fig. 6. Model Error Comparison of Maxwell Dataset

The performance of the ensemble classifier is evaluated using metrics such as Mean Absolute Error, Mean Squared Error, and Root Mean Squared Error, and is compared to other classifiers such as Random Forest. The best cost-estimating model based on fuzzy logic is utilized while designing actual software projects. The resulting ensemble classifier is validated on new, unused software project data to ensure it is generalizability and reliability before being utilized for real world cost predicFtion in software projects.

5 Discussion

The paper explores the complex interplay of factors influencing software development project outcomes, highlighting the importance of size, complexity, scope, objectives, and resources in shaping project success. As previously stated, a project's complexity is closely related to its challenges [18]. The study emphasizes the significance of system development practices, such as requirements collecting, project management methods, and user interaction. Effective requirements gathering ensures that the development team is clear about what needs to be delivered. The conclusions of this study have several practical consequences for software development processes. To handle complexity, organizations should prioritize project goals that are aligned with business objectives and use suitable development processes. Enhanced stakeholder participation and effective change management are critical for project success.

6 Future Scope

Accurate cost estimation is required for effective software project planning and execution. Underestimating leads to budget overruns, delayed delivery, or poor product quality, whereas overestimating typically results in resource waste from underutilization. Furthermore, the unpredictability of software development, characterized by constantly changing requirements and emerging technology, complicates the assessment process. Machine learning, with its ability to spot trends and learn from prior projects, is a potential solution for future project cost prediction. The purpose of this study is to apply machine learning to improve the accuracy of software project cost predictions, allowing project managers to make more informed decisions. It aims to provide more accurate and reliable estimates of the costs associated with software development processes so that business leaders may allocate funds correctly and oversee other resource-related tasks. The goal of this project is to use machine learning techniques to create a new cost estimation framework for software project planning [19] [20].

Disclosure of Interests The author(s) declared no potential conflicts of interest concerning this article's research, authorship, and publication.

References

1. Su, M. T., Ling, T. C., Phang, K. K., Liew, C. S., Man, P. Y.: Enhanced software development effort and cost estimation using fuzzy logic model. Malaysian Journal of Computer Science 20(2), 199–207 (2007).
2. Giardino, C., Paternoster, N., Unterkalmsteiner, M., Gorschek, T., Abrahamsson, P.: Software development in startup companies: the greenfield startup model. IEEE Transactions on Software Engineering 42(6), 585–604 (2015).
3. Adegbite, A. O., Adefemi, A., Ukpoju, E. A., Abatan, A., Adekoya, O., Obaedo, B. O.: Innovations in project management: trends and best practices. Engineering Science & Technology Journal 4(6), 509–532 (2023).
4. Kamble, S. S., Gunasekaran, A., Gawankar, S. A.: Sustainable Industry 4.0 framework: A systematic literature review identifying the current trends and future perspectives. Process Safety and Environmental Protection 117, 408–425 (2018).
5. Curcio, K., Malucelli, A., Reinehr, S., Paludo, M. A.: An analysis of the factors determining software product quality: A comparative study. Computer Standards & Interfaces 48, 10–18 (2016).
6. Marambe, A., Jayasundara, C.: The challenges of offshore agile software development in Sri Lanka and effects on the project outcome. Journal of Management Information Systems 9, 10–20 (2014).
7. San Cristóbal, J. R., Carral, L., Diaz, E., Fraguela, J. A., Iglesias, G.: Complexity and project management: A general overview. Complexity 2018(1), 4891286 (2018).
8. Samarasinghe, S. M., Samarasinghe, S. S. U.: Factors influencing team performance in software development projects. Contribution in Proceedings, pp. 1–2. Publisher, Location (2019).
9. Ibraigheeth, M., Fadzli, S. A.: Core factors for software projects success. JOIV: International Journal on Informatics Visualization 3(1), 69–74 (2019).
10. Bogopa, M. E., Marnewick, C.: Critical success factors in software development projects. South African Computer Journal 34(1), 1–34 (2022).
11. Nguyen, D. S.: Success factors that influence agile software development project success. American Scientific Research Journal for Engineering, Technology, and Sciences (ASRJETS) 17(1), 171–222 (2016).
12. Iivari, N.: Enculturation of user involvement in software development organizations-an interpretive case study in the product development context. In: Proceedings of the Third Nordic Conference on Human-Computer Interaction, pp. 287–296 (2004).
13. Buvik, M. P., Rolfsen, M.: Prior ties and trust development in project teams – A case study from the construction industry. International Journal of Project Management 33(7), 1484–1493 (2015).
14. Jaiswal, A., Sharma, M.: An expert estimator tool to estimate project cost and risk with early stage of function points. International Journal of Software Engineering & Applications 3(5), 1–12 (2012).
15. Jaiswal, A., & Sharma, M.: Expert Webest tool: A web-based application, estimate the cost and risk of a software project using function points. In: The II-International Conference on Advances in Computing and Information Technology, vol. 177, pp. 77–86. Chennai (2013).
16. Jaiswal, A., Raikwal, J., Chauhan, C.: Development of software projects: A review of various cost estimation techniques. Journal of Harbin Engineering University 44(7), 1–12 (2023).
17. Jaiswal, A., Raikwal, J.: A hybrid cost estimation method for planning software projects using fuzzy logic and machine learning. International Journal of Intelligent Systems and Applications in Engineering 12(1), 696–707 (2024).

18. Achamu, G., Berhan, E., Geremaw, S.: Demonstrating the interplay of machine learning and optimization methods for operational planning decision. Journal of Data, Information and Management 3(4), 297–324 (2021).
19. Roozbahani, Z., Rezaeenour, J., Shahrooei, R., Emamgholizadeh, H.: Presenting a dataset for collaborator recommending systems in academic social network: A case study on ResearchGate. Journal of Data, Information and Management 3, 29–40 (2021).
20. Liu, J., Wang, L.: How does emission trade scheme affect green total factor productivity: perspectives of efficiency decomposition and mediating effects? Journal of Data, Information and Management, 1–19 (2024).