Original Research

Evaluating Factors Affecting Project Success: An Agile Approach

Mohammad Amin Amani[1], Ayria Behdinian[1], Mohammad Sheikhalishahi[1, *]

* Corresponding Author, m.alishahi@ut.ac.ir
1- School of Industrial Engineering, College of Engineering, University of Tehran, Tehran, Iran

Revised: 1 April 2022 / Accept: 11 July 2022 / Published online: 20 July 2022

Abstract
Selection of the most influential factors to improve the performance of organizations has consistently been a significant task for project managers. These underlying factors aim to prevent the failure of the project and to improve the performance of employees. The success of the organization's projects is directly correlated to customer satisfaction, time, cost, and product quality at the time of project completion. In this paper, after reviewing the literature on the elements influencing the project's success, the extent to which each factor affects the project's success is accessed. A practical data evaluation method to predict the most underlying item is a machine learning algorithm, a perfect contributory method for project managers to examine the influential factors. After identifying the component with the highest effect on the project's success, validating the selected items in a real-world practice paves the way for assessing that factor's effectiveness. In this study, after selecting the Agile approach as the most notable, the simulation models were utilized to measure the proportion of organizational performance improvement. Agile Management, which is considered in the actual case, signifies implementing the Scrum method and all the definitions and phases related to this method in the organization. The analyzed Agile practice (Scrum) for the case study decremented the project cost and time substantially and enhanced the service and product quality.

Keywords - Agile project management; Genetic algorithm; Machine learning; Scrum; Simulation; Software project management

MOTIVATION

Throughout the years, a variety of methods in software project management have been presented and analyzed; however, many companies are not profit-driven industries. The real case (Software company) that we assessed in this research dealt with financial obstacles due to project management inefficiency. Furthermore, there was no efficient cooperation and potential feedback in order to perform projects in the organization. Hence, employing approaches or changing factors to enhance the organization's performance can significantly impact workforce performance in the company mentioned above. Hence, influential factors in the organization can be evaluated for improving project management action. The primary motivation for expanding this framework is to examine functional approaches such as agile for the software organization along with other features such as the project mission, project schedule, or top management support to analyze the impact of Agile Management along with fundamental conventional software projects management factors.

INTRODUCTION AND LITERATURE REVIEW

Project management involves leading the work team to achieve the goals set in the early stages of the project, usually in terms of cost, time, product quality, and customer satisfaction. Project management is utilized in many fields,

including construction, energy consumption, information technology, software development, healthcare, consulting, marketing and pharmaceutical.

The Management of software projects and the complexity of the tasks related to these projects have been researched for decades, and in this regard, the control of such projects is highly important [1]. The success of software projects depends on the time, cost, and quality at the end of the project compared to the project's predictions; Furthermore, customer satisfaction is one of the influential factors in achieving project goals that should be considered [2] [3].

Project managers take advantage of many components to achieve project goals during the implementation of different phases. Each of the factors utilized is somewhat influential in the project's success, so it is of significance to determine the most significant factors on employee performance and, consequently, the project's success [4]. Implementing each critical factor requires more energy and high cost and restructuring the organization, so identifying the factors that affect the project's accomplishment can lead the project to higher quality and lower cost [5]. Over time, software project managers have utilized many elements and techniques to meet the initial requirements and achieve the desired goals; thus, employees' activities and performance improvement and be more predictable, resulting in premier monitoring and accuracy during the implementation of software projects [6].

Nowadays models of project management are fully professional and give emphasis to quality and creativity [7]. Numerous methods have been applied to control software projects in various organizations to successfully perform the software development process. These methods, such as Waterfall, Prototyping, Iterative & Incremental, Spiral, Rapid Application Development, Extreme Programming, V-Model, and Scrum, can positively or negatively affect organizations concerning their size (small, medium, large) [8]. These approaches have substantial factors that can affect projects' success differently; hence, it is highly significant to discover the most underlying factor to consider [9].

It is of significance to identify why most software projects fail to address the initial goal and the path to iron out this problem derived from a lack of knowledge in terms of factors affecting the projects' performance [10]. It is estimated that 20-30% of all software projects fail, and 30-60% of projects fail to meet some of the initial goals, so it is crucial to recognize the influential factors [11]. Lyytinen and Hirschheim [12] Mentioned the reasons for failures, mainly in terms of low-quality system design, inappropriate process, and not taking customer and stakeholder satisfaction. According to [13], there are many root causes: employees' experience, comprehensible expansion of the project scope, defining understandable organization's tasks, which should be considered in order to direct a project to be successful.

Wallace, Keil [14] examined the risk of software projects and stated that various project risk levels (low, medium, high) should be determined according to the complexity of planning the projects effectively, and as a result, the probability of success is higher. Niazi, Mahmood [15] analyzed the essential factors influencing the project's success, and a total of 18 factors as the main factors affecting the software project process were extracted from previous research. Among those 18 factors, five factors with a high number of applications in organizations were selected as the most effective components in the software project's success, which include organization structure, project managers'skill, communication, requirement specification, and cultural awareness.

Zdanytė and Neverauskas [16] mentioned that the Project management methodology includes the rules and standards that the organization follows to achieve its goals. The most basic way of managing software projects, known as the traditional method or waterfall, was introduced by Winston Royce [17]. The operation of this method has been that the project is divided into different phases, and each phase will be allowed to start after the completion of the previous phase. This method paves the way for project managers to utilize the backward and forward progress between the steps [18]. Proponents declare that this method effectively improves the products' quality and estimates the project time and cost [19]; however, critics stated that because of a lack of clearness about specifications and requirements, performing the projects are likely to be failed [20]. Although the waterfall method was utilized in many organizations, the emergence of an iterative methodology called agile was a revolution in software project management [21].

Groups of developers introduced the agile approach in order to speed up the development process [22]. Due to the lack of traditional method flexibility, agile project management became popular among project managers since it provides the situation to change the scopes defined in the earlier steps [23]. Augustine [24] stated the positive side of the agile methodology, which is mainly related to customer satisfaction since customers' requirements are likely to change during the performance of the software projects. Consequently, the traditional method fails to address the difficulty, even though agile project management (APM) takes this conversion during the implementation of the projects into account.

According to [23], who conducted empirical research in terms of utilizing APM in different areas of industries, it resulted in that information technology has the highest

proportion of applying agile with the 25.1%, and the main reason for this usability is the possibility of changing the project domains throughout the projects. There are some methods in APM, such as Scrum and Kanban, the former utilized by [25] for the first time and the latter employed by Taiichi Ohno for the first time [26]. Kropp, Meier [27] researched the level of satisfaction among employees and managers while utilizing agile in the organization. It was concluded that the satisfaction of individuals in the organization is considerably high due to the challenges they face. Behutiye, Karhapää [28] mentioned that product quality management is improved by employing agile methods; thus, customers can take advantage of higher product quality, and the speed of the whole development process has incrased significantly.

The most common method of APM, which is utilized widely by agile practitioners, is Scrum, with a proportion of 68% [29]. This method is highly premier for adopting the environmental change and the complexity the team may encounter [30]. The underlying difference between Scrum and the traditional method is that the sprint is unpredictable (Sprint is an iterative period of less than one month) [31]. Three roles exist in the Scrum method: scrum master[1] (SM), product owner[2] (PO), and development team. A product owner is responsible for defining and prioritizing the stories (Tasks in Scrum method) and determining the initial overall requirements for the team. The development team's responsibility is to execute each sprint successfully and complete the whole project according to the requirements. The scrum master is accountable for accomplishing the scrum process and fitting the whole process to the organizational culture [32]. In Figure 1, the whole process of Scrum has been demonstrated basically, starting from Backlog (a prioritized feature list), defining, and ending with sprint review. The daily meeting occurs every morning, and the whole team discusses the stories context they have to make them done throughout that day.

FIGURE 1
SCRUM PROCESS

Some methods, such as machine learning, simulation, parametric models, expert estimation, etc., can be utilized to analyze and approximate several factors and goals in software projects and help to improve the project's performance.

Machine learning is a data science method to find patterns in amounts of data, which can be utilized for different objectives such as project success, estimate costs, and other targets based on independent variables[33, 34], and it has applications in the various field[35, 36]. Braga, Oliveira [37] utilized machine learning algorithms to estimate the effort. Linares-Vásquez, McMillan [38] employed a machine learning algorithm to classify the software application into specific categories to help stakeholders to realize requirements and predict maintenance problems in software projects. López-Martín and Abran [39] built a predictive model by a multi-layer perceptron (MLP) algorithm to predict new software projects' durations. Tadayon [40] utilized Neural Network to build a model to predict the cost estimation in software projects. López-Martín [41] investigated five machine learning models created to predict the software testing effort. Rathore and Kumar [42] utilized machine learning algorithms to create a predictive model for software fault prediction (SFP). Mehta and Patnaik [43] utilized a combination of Partial Least Square (PLS) Regression and Recursive Feature Elimination (RFE) to feature selection and extraction and built a predictive model by XGBoost to classify the software modules into defect prone or non-defect prone.

Simulation is one of the methodologies that lay the foundation for evaluating organizational behaviors to examine the performance of employees in various situations and make the situation for analyzing variant factor effectiveness on organization performance [44]. Various methods can be applied in a simulated model with low cost in diverse areas. [45] employed a constructive cost model (COCOMO) by utilizing a simulated model to predict learning effectiveness in software project management.

Uzzafer [46] Simulated the integrated framework of risk management, cost prediction, project planning, and the model's output was the budget required for performing all the activities. Li, Deng [47] Utilized a simulation model by taking the changes and mistakes in the software requirements into consideration, and the result demonstrated the effectiveness of Knowledge Management (KM) as an underlying factor to control the software development process.

Project managers applied various methodologies to control the software development process; therefore, evaluating these methods on a simulated model would be effective. Test-Driven Development is one method in agile practices, which was simulated by (Turnu et al., 2006), and the outcomes represented an improvement in product quality. Ghane [48] employed a Lean Six Sigma model, one of the agile methods, to develop software products and predict software development's speed and cost. Another popular method that has been widely utilized in software development is Scrum, which was simulated by [49], and the purpose of this multi-agent simulation is to investigate the impact of factors on the accomplishment of software development. These factors include the complexity of the activities and the capabilities of the team members, ultimately leading to analyzing the amount of work that the team can carry out within a fixed time.

In this paper, software project management is analyzed from various dimensions. When the underlying factors in project management are identified, and their impact on the

[1] Who ensures the Scrum framework is followed

[2] Who defines the stories and prioritizes the team backlog

project success is examined, it is conceivable for managers to pick out the most effective factors to be selected for performing a specific project. Selection of projects according to the existing factors in the organization is one of the significant dimensions of the project manager's responsibility, which is substantially considered in this research to lay the foundation for project managers to highlight the components which can affectionate various types of projects successfully. In addition, depending on the human resources and structure of software organizations, managers can evaluate several projects to figure out which type can markedly fix their organization and consequently prioritize the projects to be carried out with high productivity. For this purpose, the machine learning algorithm is utilized to build a predictive model for project selection and determine the most effective factor for project success. Moreover, a simulation model is created to measure the improvement of the project based on utilizing the most influential factor. The main objective of this paper is to examine the substantial factors in software project management that directly impact the cost, time, and quality of projects in a real case; moreover, determining the value and effectiveness of each factor is of importance. This goal can be achieved by utilizing state-of-the-art methods such as machine learning algorithms and simulation models. A brief summarization of this paper's contributions is mentioned as follows:

- Applying an integrated machine learning and simulation model to signify the importance of agile project management (Scrum method for this case) in software development
- Comparing the underlying factors affecting a software project's success by employing a machine learning algorithm
- Highlighting Scrum structures (product owner, scrum master, and the development team) effectiveness on a software organization's productivity by implementing simulation models
- Employing Agile project management (Scrum resources and steps) in software development to enhance software project management
- The decrement in the software project's time cycle and cost and the remarkable increment in software products quality by applying the Scrum method

The remainders of the paper are organized as follows. Section 2 expresses the problem definition and the problem factors. Section 3 defines a Genetic algorithm for feature selection and a Logistic regression algorithm which is a Machine learning algorithm and simulation method. In section 4, experimental results are presented. Section 5 expresses the conclusion.

PROBLEM DEFINITION

Project managers come across plenty of obstacles during the implementation of project activities; however, project success is the team's goal to reach. The project's success depends on completing the project with the time, cost, and quality specified at the initial steps for that particular product. Each software project factor can have a different impact on the project; Therefore, it is highly considerable to pick out the most influential factors concerning the project scopes, taking into account the initial requirements. In the analyzed case study, the problem that the organization faces is that despite the high working hours, it can not prepare the products at the desired time and provide the customers with remarkable quality; therefore, if the practical approaches and factors positively affect the project success can be extracted; It can assist improving the performance of the organization. One of the most functional methods for predicting each independent variable's effectiveness on the project's success is the Machine learning algorithm. This method can be perfectly applied to determine the impact of elements in the organizations, consequently paving the way for project managers to employ the company's most significant elements. After finding the most influential factor, assessing its impact on the organization by utilizing case-based data in simulation models can represent the proportion of improvement in the employees' performance. The research problem and solving process are represented in Figure 2.

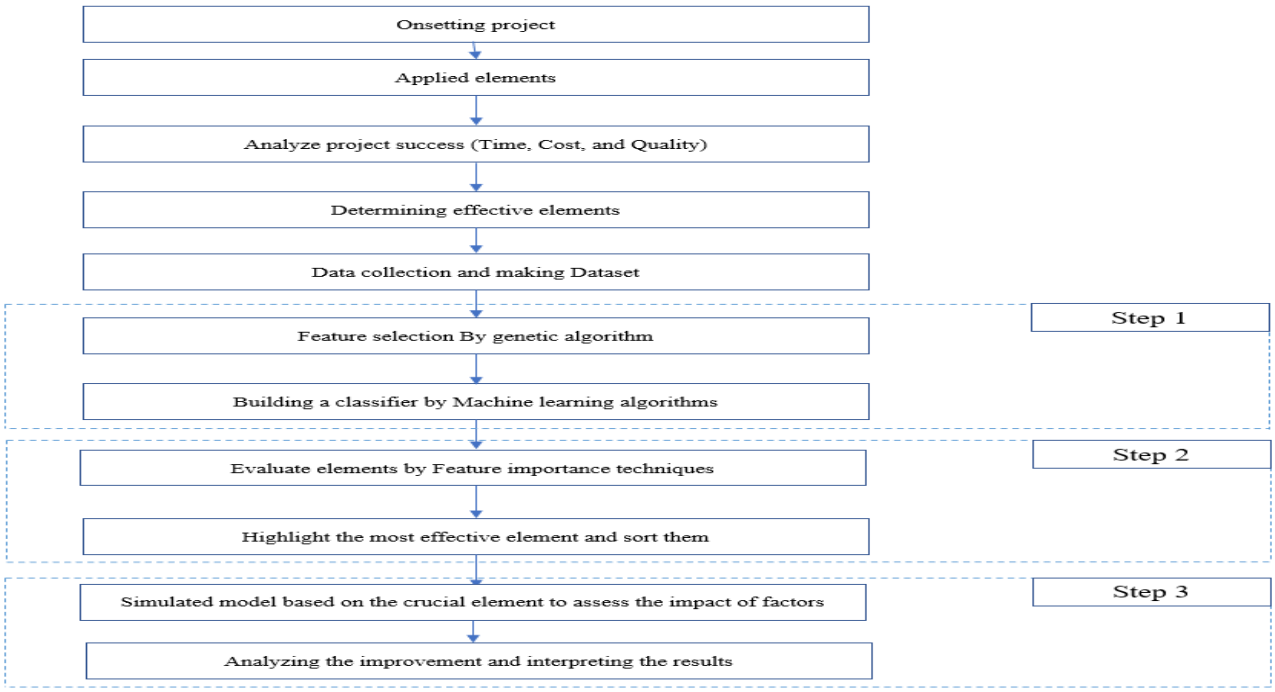

FIGURE 2
THE PROBLEM AND SOLVING PROCESS

I. *Case study*

This study used a merged dataset combining relevant information from the workforce and projects by the Donya Pardazesh Abri Occupational informatics center. The database provides all information regarding the variables and Project results. The data of this research has been collected from a software company from the actual process tracking in the Jira software[3]. Analyzing the software project management factors in a real case would significantly validate the research output. The organization's tasks primarily were generating new software products; furthermore, some projects were defined for redeveloping the conventional projects. This company breaks the projects into small tasks during the initial steps; afterward, the project manager applies Jira software as functional software to control the whole process [50]. This organization utilized the traditional method, and after years, the project manager decided to employ the Scrum method to develop the software products; thus, the latter's effect on the company's performance can be extracted. For the traditional method, twelve developers, one designer, one analyzer, and two quality assurance have participated in the model, and for the Scrum method, the Product Owner and Scrum master have been added to the software development team.

II. *Factors*

A broad number of factors have been mentioned in project management, and their effects on the organization's performance have been analyzed. Project management methods have been utilized in order to employ the most influential factors in real-world practice, consequently leading the project to success [51]. The agile method is the approach that remarkably affects the project management area, especially software project management [52]. Except for being the underlying element in project success, the agile approaches positively affect the employees and customers, leading to innovative behavior [53]. Managing project scope, time, and cost in the APM is perfectly executed; therefore, it is of significance to employ this method within the industrial sectors [54]. Successes and failures are defined concerning employing Agile in the organization; The positive points of this method are unequivocal identification of customer requirements [55] along with incrementing team communication and capability [56]; furthermore, It is applicable to prioritize projects effectively due to the complexity of the projects and their necessities. Apart from benefits, there are some disadvantages for this method, including employing this method in large organizations, which is not suitable according to some real cases. Furthermore, communication in Agile would be time-consuming and wasteful if the teams have a lot of resources [57].

[58] researched in terms of critical success factors in the project management area and mentioned 14 factors as the crucial factors in implementing the project [59]. Combining key project success factors with the agile method as an underlying element in the implementation of software

development paves the way to extract out the most critical element in software project success. These 14 independent factors, along with agile, have been opted for predicting project success, represented in Table 1.

TABLE 1
THE PROJECT FACTORS

Factor	Description	Judgement
Project mission	Clear definition of goals	(EP,P,F,N,G,VG,E)*
Top management support	The responsibilities are provided by top Management	(EP,P,F,N,G,VG,E)
Project Schedule	Details of project implementation	(EP,P,F,N,G,VG,E)
Client consultation	Taking stakeholders' expectations into account	(EP,P,F,N,G,VG,E)
Personnel	Levels of experience and expertise	(EP,P,F,N,G,VG,E)
Technical tasks	Presence of required technology	(EP,P,F,N,G,VG,E)
Client acceptance	The projects supported by the clients	(EP,P,F,N,G,VG,E)
Monitoring and feedback	Each step of the implementation is controlled	(EP,P,F,N,G,VG,E)
Communication	Communication for all the required data	(EP,P,F,N,G,VG,E)
Troubleshooting	Unexpected crises are managed	(EP,P,F,N,G,VG,E)
Agile	Applying one of the agile methods	No, Yes
Power and politics	The degree of political activity in the organization	(EP,P,F,N,G,VG,E)
Environmental events	External organizational or environmental factors	Negative,Neutral, Positive
Urgency	Importance of the project	(EP,P,F,N,G,VG,E)
Characteristics of the project team leader	Competence of project leader	administratively technically interpersonally

*Note: EP=Extreamly poor, P=Poor, F=Fair, N=Nuetral, G=Good, VG=Very good, E=Excellent

METHODOLOGY

In this section, the Genetic algorithm and its application for feature selection will be explained, and the Logistic regression algorithm will be expressed to see how it can be utilized to build the predictive model and determine the most effective independent variable which will be used in the simulation model to find its impact on project improvement, then simulation method and its input and output variables will be expressed.

I. *Genetic algorithm*

The Genetic algorithm is a search-based optimization algorithm that is inspired by the principles of genetics and evolution in nature. This algorithm utilizes to find a near-optimal response for complex problems that can't be solved in a normal time. Figure 3, illustrates the Genetic algorithm process.

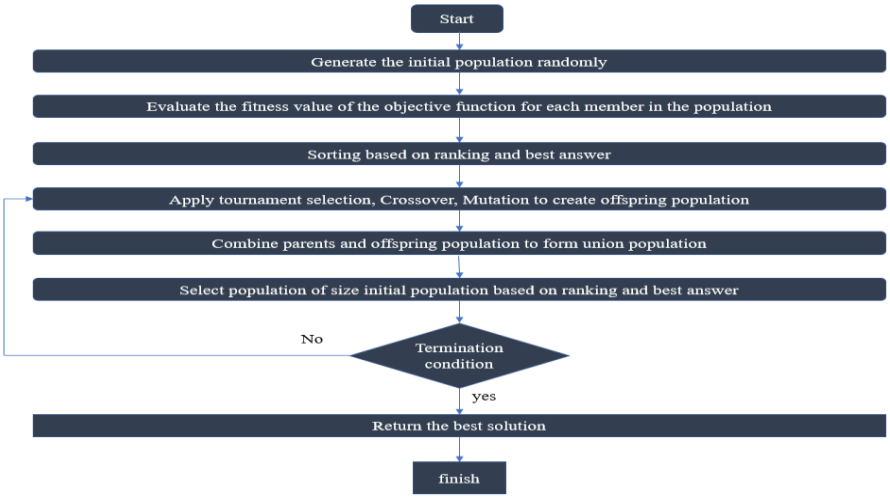

FIGURE 3
THE GENETIC ALGORITHM PROCESS

The Genetic algorithm is employed to improve the predictive model accuracy and efficiency by selecting the best combination of independent variables from the initial dataset. In the feature selection process, the main goal is to maximize the predictive model accuracy by choosing the best features and dropping redundant features.

II. *Solution reprensatation*

The algorithm's goal is to discover a combination of features that have the best efficiency in the machine learning model. Therefore, a solution's objective is to determine which features are chosen and which are dropped. Accordingly, a list of binary values is made to represent the features that are selected or dropped. Every element in that list corresponds to one of the features in the initial dataset. The selected corresponding feature is illustrated by a value of 1, and a 0 value is shown the feature isn't chosen.

- *Tournament selection*

The tournament selection operator is utilized to choose the parents from the set of individuals. The winner of each tournament (i.e., the one with the best fitness) is selected as parents to generate a new population for the next generation.

- *Crossover*

Crossover is the most prominent step in a genetic algorithm. The single-point crossover improves the exploration process in solution space by combining the parents' genetic information to generate new offspring. Offspring are made by swapping the genes of parents among themselves until the crossover point is reached. Figure 4 illustrates the single-point crossover process.

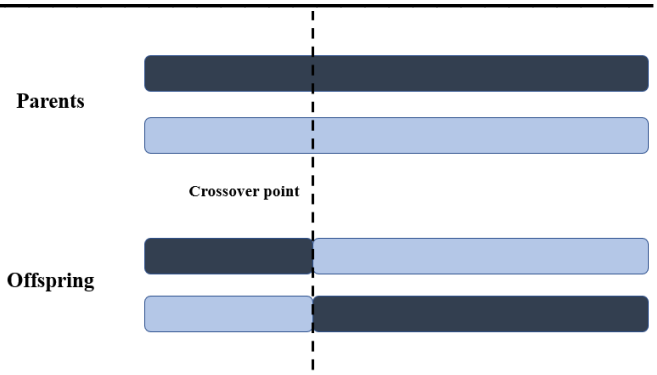

FIGURE 4
THE SINGLE-POINT CROSSOVER PROCESS

- *Mutation*

The mutation operator by producing a new population ensures the solution diversity as well as avoids falling the exploration process into a local optimum. The Flip-Bit mutation that is utilized in this paper takes the genome Bit and inverts the Bit; for example, if the genome Bit is 0, it's changed to 1. Figure 5 shows the Flip-Bit mutation process.

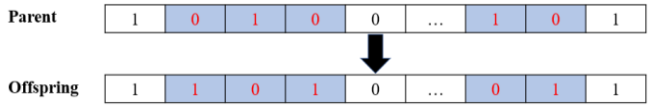

FIGURE 5
THE FLIP-BIT MUTATION PROCESS

- Machine learning

Classification refers to supervised machine learning models that classify predictions to specific classes. There are several classification algorithms, such as K-Nearest Neighbors (KNN), Random Forest, Decision Tree, multi-layer perceptron (MLP), Logistic Regression, etc. Figure 6 illustrates the classification process. The main classifier is chosen by comparison based on the predictive model accuracy among the algorithms are shown in Figure 6.

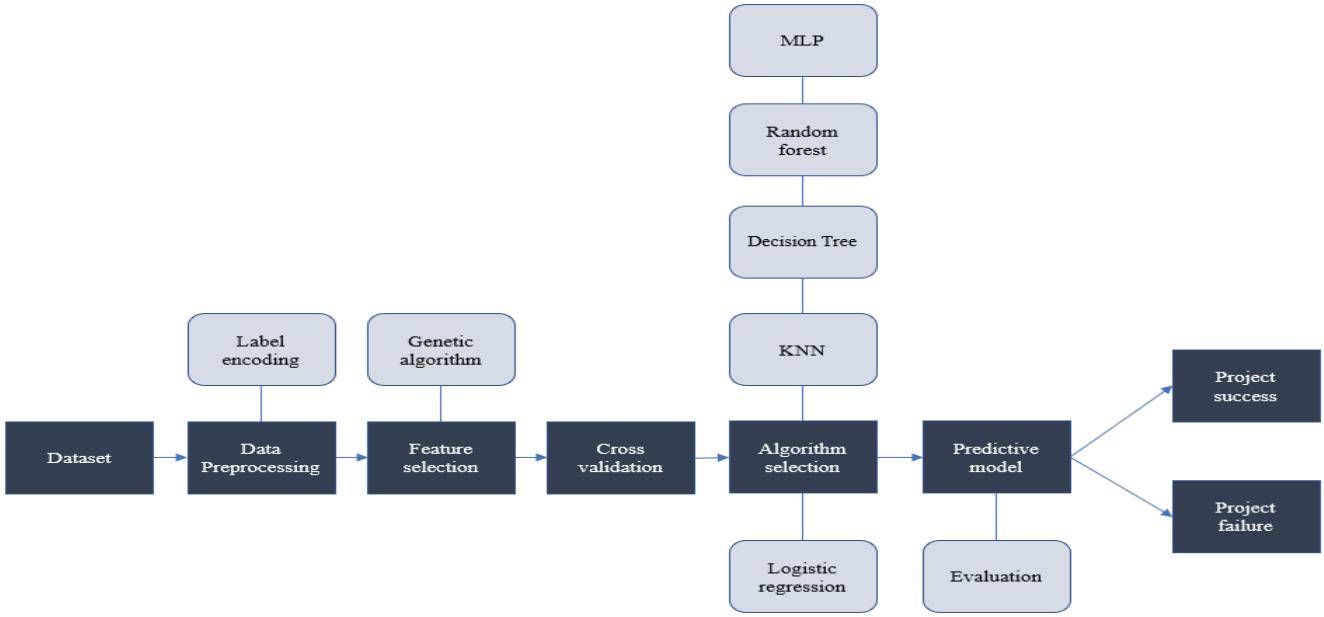

FIGURE 6
CLASSIFICATION PROCESS

- Logistic regression

Based on the statistical view, if the model parameters are linear, then the model is counted as linear too [60]. The Logistic regression output depends on the sum of the inputs and parameters; thereupon, Logistic regression is supposed as a generalized linear model. The Linear regression hypothesis is formulated as follows:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_n X_n \quad (1)$$

Where β is the coefficient and X is the independent variable. In this algorithm, the objective is to map the prediction to probability which means mapping the values to a value between 0 and 1; to reach this goal, the Sigmoid function is employed. The Sigmoid function is formulated as follows:

$$S(x) = \frac{1}{1 + e^{-(X)}} \quad (2)$$

Next, by exerting the Sigmoid function to the Linear regression hypothesis, the Logistic regression is created, which is denoted in equation (3):

$$R_\beta(X) = \frac{1}{1 + e^{-(\beta^T X)}} \quad (3)$$

The threshold is selected 0.5, which means if the probability calculated by equation (3) is equal to or bigger than the threshold $(i.e. R_\beta(X) \geq 0.5)$, the Logistic regression classified the probability to class 1, and if the probability is lower than the threshold $(i.e.\ R_\beta(X) < 0.5)$ the algorithm classified it to class 0. To find the best hyperplane to separate the classes and appraise the parameters, should minimize the Logistic regression cost function. The Logistic regression cost function is calculated by Likelihood function help as follows:

$$p(y = 1|x;\ \beta) = R_\beta(X)$$

$$p(y = 0|x;\ \beta) = 1 - R_\beta(X)$$

$$p(y|x;\ \beta) = R_\beta(X)^y (1 - R_\beta(X))^{1-y}$$

$$L(\beta) = p(Y|X;\ \beta) = \prod_{i=1}^{m} p(y^{(i)}|x^{(i)};\ \beta) \quad (4)$$

$$= \prod_{i=1}^{m} R_\beta(x^{(i)})^{y^{(i)}} (1 - R_\beta(x^{(i)}))^{1-y^{(i)}}$$

Now, the logarithm function is applied:

$$
\begin{aligned}
h(\beta) = logL(\beta) &= p(Y|X;\beta) \\
&= log\prod_{i=1}^{m} R_\beta\left(x^{(i)}\right)^{y^{(i)}}\left(1\right. \\
&\left. - R_\beta\left(x^{(i)}\right)\right)^{1-y^{(i)}} \\
&= \sum_{i=1}^{m} log\left(R_\beta\left(x^{(i)}\right)^{y^{(i)}}\left(1\right.\right. \\
&\left.\left. - R_\beta\left(x^{(i)}\right)\right)^{1-y^{(i)}}\right) \\
&= \sum_{i=1}^{m} y^{(i)} log R_\beta\left(x^{(i)}\right) + (1 \\
&- y^{(i)})\, log\,(1 - R_\beta\left(x^{(i)}\right))
\end{aligned}
\tag{5}
$$

Now, by multiple (-1) to both sides of the equation (5), the Logistic regression cost function is created, which is shown in equation (6).

$$
\begin{aligned}
J(\beta) = -h(\beta) &= -\sum_{i=1}^{m} y^{(i)} log R_\beta\left(x^{(i)}\right) + (1 \\
&- y^{(i)})\, log\,(1 - R_\beta\left(x^{(i)}\right))
\end{aligned}
\tag{6}
$$

- Validation method

There are plenty of techniques for validating the machine learning models, such as Train/test split, k-fold Cross-Validation, Leave-one-out Cross-Validation (LOOCV), Leave-one-group-out Cross-Validation (LOGOCV), Nested Cross-Validation (NCV), Wilcoxon signed-rank test, Mc Nemar's test, etc. among these techniques, k-fold Cross-validation is one of the most applied methods which is employed in this paper too.

K-fold cross-validation is applied to small datasets to avoid overfitting (i.e., the model has trained properly but can't predict well). In this technique, the training set is apportioned into K smaller sections, and then the machine learning model is trained by using K-1 of the sections, and the remaining section is utilized for validating. This process repeats, and each time, the training folds and validation section change, and the test part is used for final evaluation; consequently, the average accuracy is calculated. Figure 7 illustrates the K-fold cross-validation process.

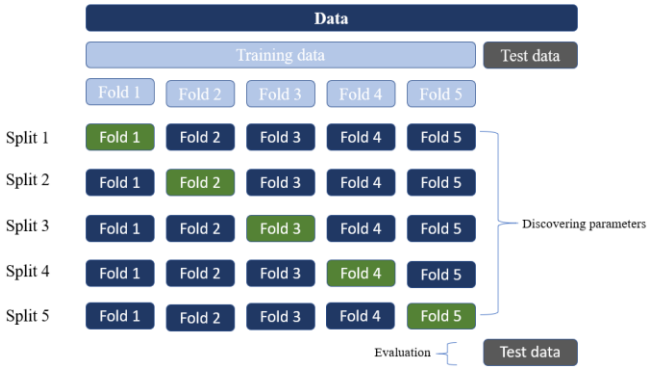

FIGURE 7
THE K-FOLD CROSS-VALIDATION PROCESS

- Performance evaluation metrics

There are several methods to evaluate the performance of the classification's model, such as Precision, Recall, F1-score, Accuracy, etc. The Precision, Recall, F1-score, and Accuracy metrics are calculated based on the Confusion Matrix in equations (7) – (10).

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

$$F1 - score = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{10}$$

Where TP, TN, FP, and FN are described as follows:

- True Positive (TP): the predictive model predicted positive, and the main value is positive.
- True Negative (TN): the predictive model predicted negative, and the main value is negative.
- False Positive (FP): the predictive model predicted positive, but the main value is negative (Type 1 error).
- False Negative (FN): the predictive model predicted negative, but the main value is positive (Type 2 error).

- Feature importance

To find the most effective independent variable on the objective, the Euler number to the power of each independent variable's coefficient is calculated, and then the results are compared, and the bigger number represents that the feature has more effect on project success which can be proved as follows:

$$\frac{P\,(y=1)}{P\,(y=0)} = \frac{P\,(y=1)}{1-P\,(y=1)} = \frac{\frac{1}{1+e^{-z}}}{1-\frac{1}{1+e^{-z}}} = e^{z} = e^{\theta_0+\theta_1X_1+\theta_2X_2+\ldots+\theta_nX_n} \quad (11)$$

Now, when we make a change on any independent variable, such as increasing X_1 by 1 unit, the prediction will change as follows:

$$\frac{y_{new}}{y} = \frac{e^{\theta_0+\theta_1(X_1+1)+\theta_2X_2+\ldots+\theta_nX_n}}{e^{\theta_0+\theta_1X_1+\theta_2X_2+\ldots+\theta_nX_n}} \quad (12)$$

$$= e^{\theta_0+\theta_1X_1+\theta_2X_2+\ldots+\theta_nX_n-\theta_0-\theta_1X_1-\theta_2X_2-\ldots-\theta_nX_n}$$

$$= e^{\theta_1(X_1+1)-\theta_1X_1} = e^{\theta_1(X_1+1-X_1)} = e^{\theta_1}$$

So, if the feature X_1 is increased 1 unit; therefore, the prediction will change e^{θ_1}, which is the Euler number to the power of the independent variable's coefficient.

III. Simulation

After implementing the machine learning process, the influential factors in the project's success can be identified. As a result, project management concentrates on the most significant factors of project goals. In the present study, six factors had the most considerable impact on the projects' success, among which Agile was recognized as the most influential factor. For evaluating the importance of the Agile method, the Simulation approach was opted. This method provides the situation to modify the components to demonstrate the impact of each on the organization's performance. In the present study, two separate models, called traditional and Agile approaches, have been developed to show the importance of the Agile method. The outputs of the two models can be compared and would effectively analyze how much improvement the organization can experience during the implementation of the Agile procedure. For indicating the effect of a factor, other inputs of the simulation model must be fixed so that the outputs can be examined precisely based on a factor. The inputs and outputs of the simulation models are specified in Table 2.

TABLE 2
SIMULATION INPUT AND OUTPUT VARIABLES

Variable	Role
Agile	Input
Top management support	Input
Project schedule/plan	Input
Client consultation	Input
Personnel	Input
Organization expenditure in the agile and traditional method	Input
Total cost for a project	Output
Cycle time for a project	Output
Total bugs for a specific project	Output

For analyzing the organization's expenditure during the traditional and agile method, it is required to calculate the company's different costs, represented in Table 3.

TABLE 3
THE COMPANY'S DIFFERENT COSTS

Factor	Value ($)	
	Traditional	Agile
Building rent	995	995
Facilities	1050	1170
Staffing	3252	3652
Total monthly expenditure	5297	5817
Hourly expenditure	22.07	24.23

According to the monthly expenditure, the agile method cost is higher, and the main reason for this is joining employees such as SM and PO. The simulation model hourly cost would be utilized to calculate each project implementation expenditure in the Agile and traditional approaches.

- Non-agile model

The Non-agile method (Traditional method called Waterfall) Concentrated on how organizations implement the projects before applying Agile methods. The waterfall model comprises the steps: Analysis, Design, Implementation, Testing, and deployment, all of which have been considered in the discrete model [61]. The resources that participated in this model are demonstrated in Figure 8: one project management, one analyzer, one designer, six backend programmers, six frontend programmers, two quality controllers. The overview of the Waterfall simulation model is demonstrated in Figure 9. The tasks for each of the resources have been displayed, and the resources aligned to that specific task are named in a circular shape. By entering different inputs, each task's time would alter; therefore, the output would be diverse.

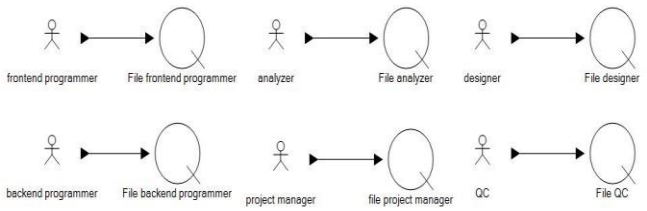

FIGURE 8
NON-AGILE MODEL RESOURCES

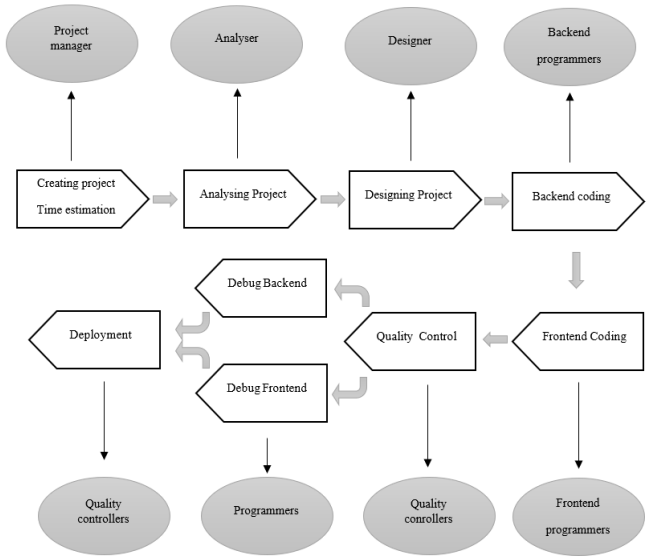

FIGURE 9
NON-AGILE MODEL (TRADITIONAL MODEL)

- Agile model

For the Agile model, the Scrum method, one of the most popular approaches in the Software Project Management, has been applied. The resources that participated in the Agile method are displayed in Fig 10: one Product owner, One scrum Master, 16 members of three Development teams comprising one analyzer, one designer, 12 developers, two quality controllers. According to the literature review, the process of Scrum has been demonstrated in Figure 11. begging with prioritizing the stories in the Backlog by the product owner and ending with a Sprint review. The circular shape shows the resources.

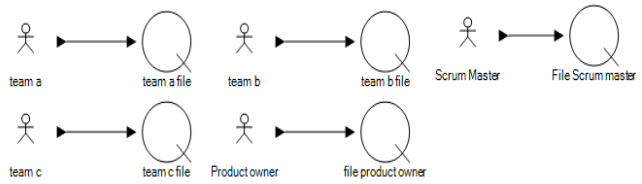

FIGURE 10
AGILE RESOURCES

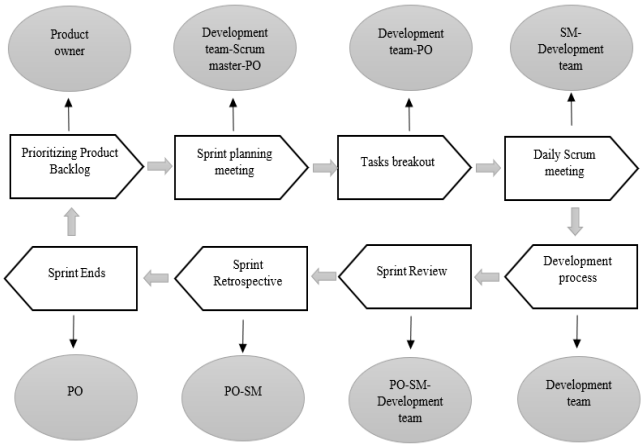

FIGURE 11
SCRUM MODEL

NUMERICAL RESULTS AND DISCUSSION

In this section, the predictive model will be built by Logistic regression and evaluated by different metrics. The most effective independent variable will be determined; then, the simulation model will be made to see the impact of the independent variable on the project improvement.

I. Machine learning results

The Label-encoding technique is applied before the predictive model is built to change the categorical variables to numerical. The One-hot encoding technique is more popular than Label-encoding, but in this case, the dataset is included nominal variables that each of them has several categories; therefore, Label-encoding was a more suitable technique.

In the next step, the Genetic algorithm is utilized to find the best subset of independent variables to build the predictive model. The feature selection result is mentioned in Table 4.

TABLE 4
FEATURE SELECTION RESULT

	Genetic algorithm	
Instance	Feature (PM*,TMS*,PS*,CC*,P*,TT*,CA*,MF*,C*,T*,A*,EE*,U*,PP*,CPTL*)	Accuracy (%)
1	(1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0)	93

2	(1, 1, 1, 1, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 0)	92.23
3	(1, 1, 1, 1, 1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0)	91.55
4	(1, 1, 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 0)	91.4
5	(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1)	89.9

*Note: PM= Project mission, TMS= Top management support, PS= Project schedule/plan, CC=Client consultation, P=Personnel, TT=Technical tasks, CA= Client acceptance, MF=Monitoring and feedback, C=Communication, T=Troubleshooting, A=Agile, EE=Environmental effect, U=Urgency, PP=Power and politics, CPTL=Characteristic of the project team leader

1

Due to Table 4, the best performance is achieved when the selected features are Project mission, Top management support, Project schedule/plan, Client consultation, Personnel, and Agile.

Now, by comparing the machine learning classification algorithms based on accuracy, the best algorithm is chosen among others. The results are denoted in Table 5.

TABLE 5
THE MACHINE LEARNING ALGORITHM COMPARISON

Algorithm	Accuracy (%)
Logistic regression	93
Decision Tree	83
Random Forest	92
KNN	91
MLP	89

Based on Table 5, the Logistic regression is selected as a classifier to build the predictive model. Besides the accuracy, there are other metrics such as the Confusion matrix, Precision, Recall, and F1-score to evaluate the predictive model's performance created by the Logistic regression algorithm. Figure 12 illustrates the Confusion matrix result.

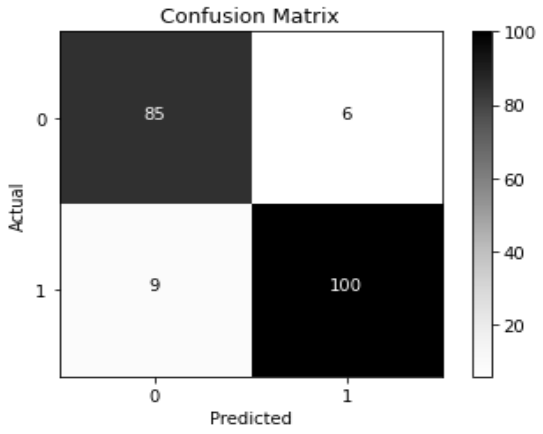

FIGURE 12
CONFUSION MATRIX RESULT

Precision, Recall, and F1-score metrics are calculated for each class, and the results are presented in Table 6.

TABLE 6
THE EVALUATION RESULT

	Metrics		
Class	Precision (%)	Recall (%)	F1-score (%)
0	90	93	92
1	94	92	93

The predictive model is built by a Logistic regression algorithm, and evaluation metrics showed excellent results. This model is useful to assess the new project's result before starting it and to find out the project will be successful or fail. In other words, this is a model for project selection.

Several factors are effective on the project's result. This is important to understand which factor has more effect on the project's success; then, by investing in that factor and considering the financial and other resources constraints, get a better result. For this purpose, the feature importance method based on the coefficient explained in section 3.2.3 is utilized. The results are shown in Figure 13.

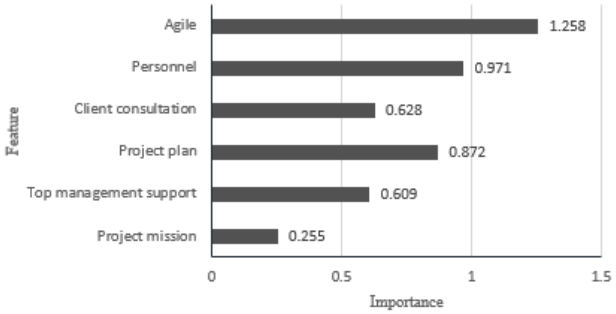

FIGURE 13. FEATURE IMPORTANCE RESULT

Based on Fig 13, Agile has the most effect on the project's success. In the next section, the simulation model is created to analyze the improvement percentage that can happen by utilizing Agile in projects.

II. Simulation results

The simulation result is displayed in this section. The time, Cost, and Quality of the 20 projects selected from the dataset are presented for comparing the two discrete models. In table 7, the time cycle for twenty projects in two discrete models, which are traditional and agile, is displayed, and the amount of improvement by applying agile is reported. For more credibility, the 100 run numbers have been considered for both models.

TABLE 7
EVALUATION OF AGILE EFFECTIVENESS ON PROJECT IMPROVEMENT BASED ON CYCLE TIME

Project code	PM*	TMS*	PS*	CC*	personnel	Traditional model time(hour)	Agile model time(hour)	Improvement (%)
1	6	6	3	6	5	59.56	49.91	10
2	4	6	5	5	2	59.17	46.37	13
3	5	4	6	6	3	53.26	38.38	15
4	6	5	3	5	2	63.76	46.18	18
5	3	3	5	5	2	61.22	51.17	10
6	5	6	4	3	4	59.81	45.74	14
7	1	6	4	1	3	74.94	58.35	17
8	4	1	6	5	4	61.28	46.42	15
9	0	0	4	3	6	61.53	55.29	6
10	1	3	4	0	3	74.12	60.31	14
11	5	4	6	3	2	58.98	42.92	16
12	6	0	3	4	4	68.17	49.68	18
13	1	5	6	3	4	62.04	50.04	12
14	0	0	0	1	1	93.94	73.25	21
15	5	5	1	6	4	63.31	50.85	12
16	4	3	0	6	4	75.94	51.65	24
17	4	6	3	6	5	55.63	40.44	15
18	1	3	4	0	3	72.78	55.5	17
19	0	0	4	3	6	65.47	59.99	5
20	6	2	4	4	5	58.7	40.36	18
Average						65.18	50.64	15

*Note:PM=project mission,TMS=Top management support,PS=project schedule,CC=client consultation

The comparison between the two models signified the total amount of 15% improvement in agile approach implementation.

After calculating Agile and traditional methods' expenditure from the information within the organization, the entire cost for these twenty projects can be computed. Figure 14 demonstrates the sum of twenty project costs in dollar units.

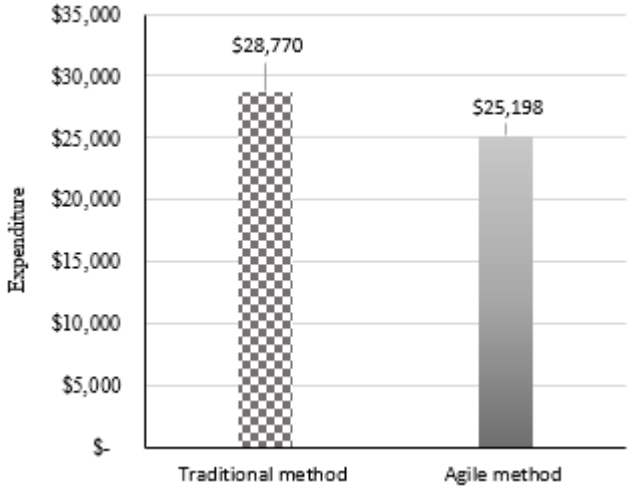

FIGURE 14
AGILE VS. TRADITIONAL METHOD EXPENDITURE

The comparison between these two columns represents the total proportion of 13% improvement in the projects executed by employing the Agile method. Even though the staffing cost has incremented by adding some roles for developing the Scrum method in the organization, the total expenditure has decreased.

When it comes to quality within the organization, the number of bugs that the quality controller figures out is highly significant. Two kinds of bugs may be discovered in the whole software development, the first of which is identified by the quality controller within the organization, and the second type is related to customer recognition. By implementing the simulation models, both types of bugs have been calculated and displayed in Figure 15.

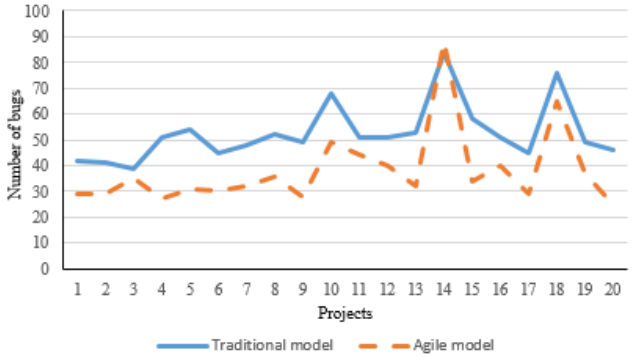

FIGURE 15. NUMBER OF THE BUGS FOR THE SIMULATION MODELS

The comparison between figures shows the agile method has positively affected the software employees' work Quality. The figures show that the number of bugs has reduced notably, and the organization could improve the employees' performance by 29% if the Agile approach is utilized.
From a practical point of view, employing the Agile method paves the way for project managers to increment the team capability and improve the relationships with the teams. By concerning the time, cost, and quality, the Cycle time and cost decreased in the whole process, and the employees' performance Quality has significantly enhanced.

SENSITIVITY ANALYSIS

According to the results of the surveys, apart from the Agile factor, Personnel is a substantial factor to be considered. To analyze the impact of personnel capabilities on Project Management factors (Time, Quality, and Cost), we examined a specific project (PM=6, TMS=6, PS=3, CC=5, Personnel=1) under different conditions. Transformation in the situation is such that all factors except Personnel are fixed. The amount of improvement in time, cost, and quality after changing Personnel skills from Extremely Poor to Excellent in the simulation model is presented in Figure 16.

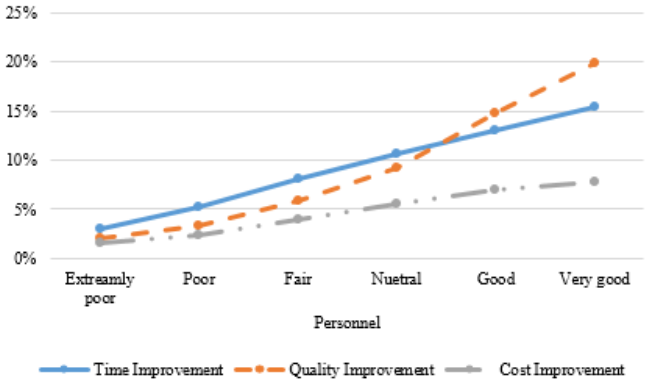

FIGURE 16
PERSONNEL SENSITIVITY ANALYSIS

The results show that skilled Personnel can positively affect the quality of work and reduce project time. However, It can not significantly impact costs, and the main reason is the high cost of skilled human resources in the software development field.

MANAGERIAL INSIGHTS

The proposed approach can perfectly fit the software company. From a practical point of view, agile project management is a practical option for a web-based organization to boost products and service quality and improve the organization's productivity. Secondly, Customer satisfaction, which is an essential factor in project success, is highlighted in agile practice since in every step of the project, the customer expectations are considered to be addressed, and continuously inspecting and adapting ways of work can decrease uncertainty during the implementation of the projects and if shifted is required, it is feasible in this approach. Apart from these items, A product or service is delivered with a higher frequency on agile Management; hence, if a task has been executed incorrectly, it can be figured out and addressed immediately not to lead the project to failure. A face-to-face meeting contributes to determining the flaws in projects and paving the way to get feedback according to the activities that roles in the organization have performed; therefore, this practice can significantly enhance efficiency.

CONCLUSION

The influential factors for project success are still being explored in organizations regarding Project Management. In this study, an attempt was made to obtain significant factors involved in project management by reviewing the literature and measuring the impact of each on the success of the project. In this research, an effective method in software project management called Agile project management (Scrum method for this case) was included alongside other influential factors of project managers to determine whether this method can be the most influential factor or whether paying attention to other factors and utilizing the traditional method can be more applicable. According to the data analysis done by applying a Machine learning algorithm, it was represented that Agile Management has the most considerable impact on employee performance. As a result, shifting the organization's structure from traditional to Agile mode can substantially affect the organization's productivity. For validating the effectiveness of Agile Management,

simulation methods were employed. Consequently, the amount of time, cost, and quality of projects was determined, and the value of utilizing this method was highlighted for project managers in the software area. Due to the results, the time and cost of the projects completed with Agile have been diminished compared to the traditional projects, and the quality of the teamwork has increased significantly, which is a result of reducing the products' bugs. One of the notable effects of lessening errors in final products is undeniably correlated to incrementing customer satisfaction.

In the research conducted in Agile project management, the Scrum method was utilized, and a considerable effect was observed in improving the organization's performance. In this paper, some limitations affected the whole process. First, the captured data was from a software organization, and the proportion of data was unstructured. Hence, the number of features that could be assessed was limited; the dataset was formed regarding project management reports. Second, implementing real projects within the organization requires a lot of budget and time, forcing researchers to execute a tiny number of projects in the actual workplace.

For future research, the other agile method, namely, Kanban, Hybrid, Bimodal, Lean, etc., can be analyzed to indicate their effectiveness on organizations depending on their sizes (small, medium, large). Therefore, it is substantial to review each method and determine the impact of each on the project's success. Furthermore, it is substantial to determine whether the alteration from the traditional model of software development to the agile approaches mentioned above is feasible in terms of financial, time and quality or not.

CONFLICT OF INTERESTS

The authors declare no conflict of interest during this study.

1
2

REFERENCES

[1] Abdel-Hamid, T.K., K. Sengupta, and C. Swett, *The impact of goals on software project management: An experimental investigation.* MIS quarterly, 1999: p. 531-555.

[2] Ramírez-Mora, S.L., H. Oktaba, and H. Gómez-Adorno, Descriptions of issues and comments for predicting issue success in software projects. Journal of Systems and Software, 2020. 168: p. 110663.

[3] Ling, F.Y.Y., et al., *Human resource management practices to improve project managers' job satisfaction.* Engineering, Construction and Architectural Management, 2018.

[4] Nasir, M.H.N. and S. Sahibuddin, *Critical success factors for software projects: A comparative study.* Scientific research and essays, 2011. **6**(10): p. 2174-2186.

[5] Sudhakar, G.P., *A model of critical success factors for software projects.* Journal of Enterprise Information Management, 2012.

[6] Cerdeiral, C.T. and G. Santos, *Software project management in high maturity: A systematic literature mapping.* Journal of Systems and Software, 2019. **148**: p. 56-87.

[7] Tohidi,H. Human resources management main role in information technology project management. Procedia Computer Science, 2011. 3 p. 925-929.

[8] Saeed, S., et al., *Analysis of Software Development Methodologies.* International Journal of Computing and Digital Systems, 2019. **8**(5): p. 446-460.

[9] Subramaniam, C., R. Sen, and M.L. Nelson, *Determinants of open source software project success: A longitudinal study.* Decision Support Systems, 2009. **46**(2): p. 576-585.

[10] Arbabi, H., M.-J. Salehi-Taleshi, and K. Ghods, *The role of project management office in developing knowledge management infrastructure.* Engineering, Construction and Architectural Management, 2020.

[11] Taherdoost, H. and A. Keshavarzsaleh, *A theoretical review on IT project success/failure factors and evaluating the associated risks.* Mathematical and Computational Methods in Electrical Engineering, 2015.

[12] Lyytinen, K. and R. Hirschheim, *Information systems failures—a survey and classification of the empirical literature.* Oxford surveys in information technology, 1988: p. 257-309.

[13] Murray, M. and G. Coffin, *A case study analysis of factors for success in ERP system implementations.* AMCIS 2001 Proceedings, 2001: p. 196.

[14] Wallace, L., M. Keil, and A. Rai, *Understanding software project risk: a cluster analysis.* Information & management, 2004. 42(1): p. 115-125.

[15] Niazi, M., et al., *Toward successful project management in global software development.* International Journal of Project Management, 2016. **34**(8): p. 1553-1567.

[16] Zdanytė, K. and B. Neverauskas, *The theoretical substation of project management challenges.* Economics & Management, 2011. **16**.

[17] Scacchi, W., *Process models in software engineering.* Encyclopedia of software engineering, 2002.

[18] Thomas, G. and W. Fernández, *Success in IT projects: A matter of definition?* International journal of project management, 2008. **26**(7): p. 733-742.

[19] Stare, A., *Agile project management in product development projects.* Procedia-Social and Behavioral Sciences, 2014. **119**: p. 295-304.

[20] Saynisch, M., *Beyond frontiers of traditional project management: An approach to evolutionary, self-organizational principles and the complexity theory—results of the research program.* Project Management Journal, 2010. **41**(2): p. 21-37.

[21] Špundak, M., *Mixed agile/traditional project management methodology–reality or illusion?* Procedia-Social and Behavioral Sciences, 2014. **119**: p. 939-948.

[22] Williams, T., *Assessing and moving on from the dominant project management discourse in the light of project overruns.* IEEE Transactions on engineering management, 2005. **52**(4): p. 497-508.

[23] Ciric, D., et al., *Agile vs. Traditional Approach in Project Management: Strategies, challenges and reasons to introduce agile.* Procedia Manufacturing, 2019. **39**: p. 1407-1414.

[24] Augustine, S., *Managing agile projects*. 2005: Prentice Hall PTR.

[25] Takeuchi, H. and I. Nonaka, *The new new product development game.* Harvard business review, 1986. **64**(1): p. 137-146.

[26] Huang, C.-C. and A. Kusiak, *Overview of kanban systems.* 1996.
[27] Kropp, M., et al., *Satisfaction and its correlates in agile software development.* Journal of Systems and Software, 2020. **164**: p. 110544.
[28] Behutiye, W., et al., *Management of quality requirements in agile and rapid software development: A systematic mapping study.* Information and software technology, 2020. **123**: p. 106225.
[29] Allisy-Roberts, P., et al., *The 11th Annual State of Agile Report.* Journal of the ICRU, 2017. **6**(2): p. 7-8.
[30] Bröchner, J. and U. Badenfelt, *Changes and change management in construction and IT projects.* Automation in Construction, 2011. **20**(7): p. 767-775.
[31] Schwaber, K., *Scrum development process*, in *Business object design and implementation*. 1997, Springer. p. 117-134.
[32] Schwaber, K., *Agile project management with Scrum*. 2004: Microsoft press.
[33] Amani, M.A. and S.A. Sarkodie, *Mitigating spread of contamination in meat supply chain management using deep learning.* Scientific Reports, 2022. **12**(1): p. 1-10.
[34] Amani, M.A. and F. Marinello, *A deep learning-based model to reduce costs and increase productivity in the case of small datasets: A case study in cotton cultivation.* Agriculture, 2022. **12**(2): p. 267.
[35] Amani, M.A., M. Ghafari, and M.M. Nasiri, *Targeted vaccination for Covid-19 based on machine learning model: A case study of Jobs' prioritization.* Advances in Industrial Engineering, 2021. **55**(4): p. 433-446.
[36] Behdinian, A., et al., *An integrating Machine learning algorithm and simulation method for improving Software Project Management: A real case study.* Journal of Quality Engineering and Production Optimization, 2022.
[37] Braga, P.L., A.L. Oliveira, and S.R. Meira. *Software effort estimation using machine learning techniques with robust confidence intervals.* in *7th international conference on hybrid intelligent systems (HIS 2007)*. 2007. IEEE.
[38] Linares-Vásquez, M., et al., *On using machine learning to automatically classify software applications into domain categories.* Empirical Software Engineering, 2014. **19**(3): p. 582-618.
[39] López-Martín, C. and A. Abran, *Neural networks for predicting the duration of new software projects.* Journal of Systems and Software, 2015. **101**: p. 127-135.
[40] Tadayon, N. *Neural network approach for software cost estimation*. in *International Conference on Information Technology: Coding and Computing (ITCC'05)-Volume II*. 2005. IEEE.
[41] López-Martín, C., *Machine learning techniques for software testing effort prediction.* Software Quality Journal, 2021: p. 1-36.
[42] Rathore, S.S. and S. Kumar, *Software fault prediction based on the dynamic selection of learning technique: findings from the eclipse project study.* Applied Intelligence, 2021: p. 1-16.
[43] Mehta, S. and K.S. Patnaik, *Improved prediction of software defects using ensemble machine learning techniques.* Neural Computing and Applications, 2021: p. 1-12.
[44] Klemen, P., et al. *From high-level real-time software design to low level hardware simulation: a methodology to evaluate performances of control embedded systems*. in *PDes 2009, Workshop on Programmable Devices and Embedded Systems*. 2009.
[45] Pfahl, D., et al., *Evaluating the learning effectiveness of using simulations in software project management education: results from a twice replicated experiment.* Information and software technology, 2004. **46**(2): p. 127-147.
[46] Uzzafer, M., *A simulation model for strategic management process of software projects.* Journal of Systems and Software, 2013. **86**(1): p. 21-37.
[47] Li, D., et al., *Dynamic simulation modelling of software requirements change management system.* Microprocessors and Microsystems, 2021. **83**: p. 104009.
[48] Ghane, K. *A model and system for applying Lean Six sigma to agile software development using hybrid simulation*. in *2014 IEEE International Technology Management Conference*. 2014. IEEE.
[49] Ramingwong, S. and L. Ramingwong, *Plasticine scrum: an alternative solution for simulating scrum software development*, in *Information Science and Applications*. 2015, Springer. p. 851-858.
[50] Fisher, J., D. Koning, and A. Ludwigsen, *Utilizing Atlassian JIRA for large-scale software development management*. 2013, Citeseer.
[51] Zuo, J., et al., *Soft skills of construction project management professionals and project success factors: A structural equation model.* Engineering, Construction and Architectural Management, 2018.
[52] Lalsing, V., S. Kishnah, and S. Pudaruth, *People factors in agile software development and project management.* International Journal of Software Engineering & Applications, 2012. **3**(1): p. 117.
[53] Malik, M., S. Sarwar, and S. Orr, *Agile practices and performance: Examining the role of psychological empowerment.* International Journal of Project Management, 2021. **39**(1): p. 10-20.
[54] Hassan, A., S. Younas, and A. Bhaumik, *Exploring an Agile Plus Approach for Project Scope, Time, and Cost Management.* International Journal of Information Technology Project Management (IJITPM), 2020. **11**(2): p. 72-89.
[55] Tam, C., et al., *The factors influencing the success of on-going agile software development projects.* International Journal of Project Management, 2020. **38**(3): p. 165-176.
[56] Dikert, K., M. Paasivaara, and C. Lassenius, *Challenges and success factors for large-scale agile transformations: A systematic literature review.* Journal of Systems and Software, 2016. **119**: p. 87-108.
[57] Dhir, S., D. Kumar, and V. Singh, *Success and failure factors that impact on project implementation using agile software development methodology*, in *Software engineering*. 2019, Springer. p. 647-654.
[58] Pinto, J.K. and D.P. Slevin, *Critical success factors in R&D projects.* Research-technology management, 1989. **32**(1): p. 31-35.
[59] Costantino, F., G. Di Gravio, and F. Nonino, *Project selection in project portfolio management: An artificial neural network model based on critical success factors.* International Journal of Project Management, 2015. **33**(8): p. 1744-1754.
[60] Amani, M.A., et al., *A machine learning-based model for the estimation of the temperature-dependent moduli of graphene oxide reinforced nanocomposites and its application in a thermally affected buckling analysis.* Engineering with Computers, 2021. **37**(3): p. 2245-2255.
[61] Bassil, Y., *A simulation model for the waterfall software development life cycle.* arXiv preprint arXiv:1205.6904, 2012.