ISSN 1816-6075 (Print), 1818-0523 (Online)
Journal of System and Management Sciences
Vol. 15 (2025) No. 2, pp. 73-90
DOI:10.33168/JSMS.2025.0205

Developing an Adaptive Risk Analysis Tool for Enhancing IT Project Efficiency: A Machine Learning-Based Approach

Joseph Teguh Santoso [1], Danny Manongga [2], Ade Iriani [2], Hendry [2]

[1] Faculty of Computer and Business University of Computer Science and Technology (STEKOM University) Semarang Indonesia

[2]Faculty of Information Technology Satya Wacana Christian University (SWCU) Salatiga Indonesia.

joseph_teguh@stekom.ac.id (Corresponding Author), danny.manongga@uksw.edu, ade.iriani@uksw.edu, hendry@uksw.edu

Abstract. This study proposes an Adaptive Risk Analysis Tool based on machine learning techniques to enhance IT project efficiency through improved risk analysis and project management. The tool utilizes Instance-based learning and Regression models to provide adaptive and responsive risk analysis, learning from historical data and user experiences. The research aims to address the high project failure rates attributed to poor planning and inadequate risk management by developing a novel machine learning-based solution integrated into a user-friendly web platform. The effectiveness of the proposed approach is evaluated using generated datasets, demonstrating its ability to accurately predict and prevent project risks. The Adaptive Risk Analysis Tool has the potential to significantly improve project success rates by assisting project managers in identifying and mitigating risks more effectively. Future research should incorporate global user patterns, integrate external data, and conduct further testing with complex real-world scenarios to enhance the tool's performance and applicability.

Keywords: IT Project Management, Machine Learning, Risk Analysis, Project Planning

1. Introduction

In the mid-20th century, the world experienced an information revolution through various new inventions and widespread computer adoption, ultimately enhancing communication, and productivity, and accelerating global economic development (Bahrini & Qaffas, 2019; Schweikl & Obermaier, 2020; Skare & Riberio Soriano, 2021; Venturini, 2022). In 2021, the adoption of the Internet for communication gave rise to millions of information technology (IT) job markets because of the necessity to create and incorporate an increasing array of IT software initiatives. Despite the positive impact of technological advancements, IT projects often face challenges. (Menezes et al., 2019; Schmidt, 2023; Westenberger et al., 2022) highlight several reasons for IT project failures, including unrealistic goals, unclear system requirements, limited technical skills, inadequate risk management, poor communication, and unrealistic estimates. (Menezes et al., 2019) specifically identify risk factors in software development, emphasizing the need for technical skills. Additionally, (Iriarte & Bayona, 2020) adds a new dimension by mentioning three main factors influencing IT project management failures: poor business case definition, unclear or poorly defined project benefits, and poor financial management. This approach provides a holistic view of issues related to IT projects. Recent findings (Bogopa & Marnewick, 2022) highlight factors contributing to project failure related to processes, such as unrealistic budgets, schedules, or methodologies that can contribute to the failure of IT project management. A profound understanding of IT project failure involves technical, managerial, and procedural aspects, and therefore, addressing these issues requires a comprehensive and coordinated approach. In this case, several studies have found a lack of a proactive approach in IT project risk management. Previous research (Schmidt, 2023; Westenberger et al., 2022) indicates that many IT projects fail due to insufficient proactive risk identification and mitigation. Additionally, limitations in risk analysis have been identified. For instance, (Menezes et al., 2019) also highlight that limitations in accurate risk analysis can lead to project failure. (Iriarte & Bayona, 2020; Schmidt, 2023; Westenberger et al., 2022) also demonstrate that IT project failures can be caused by various factors, including a lack of understanding of comprehensive risk management. Therefore, there is a need for a comprehensive approach to IT project risk management.

By identifying these gaps, this research aims to make a significant contribution to enhancing the efficiency, effectiveness, and success of IT projects through the development of a machine learning-based adaptive risk management tool. Hence, this study aims to develop an Adaptive Risk Analysis Tool based on machine learning to assist project managers in identifying and managing risks more effectively. By integrating machine learning techniques into risk management tools, this research also aims to improve the accuracy of risk analysis and help reduce uncertainty in project planning while providing a holistic approach to managing IT project risks. In this case, the proposed model is expected to enhance project planning activities through the identification of various issues faced. The research will focus on modeling solutions using a machine-learning algorithm approach that enables risk analysis for planned key points. Therefore, it is important to consider people as a key factor in the proposed solution (for example, project managers' learning over time). For the proposed platform framework to be efficient, implementable, and embraced in practical situations, the machine learning algorithms employed must have the capability to provide real-time solutions (or at least close to real-time) and be integrated into a user-friendly graphical interface. To achieve this, various scholarly articles in the field of machine learning will be extensively analyzed and adapted to project management problems. The main contribution of this research is a solution to prevent project failure risks using machine learning algorithms. The research will demonstrate the effectiveness of the algorithms used in the proposed model in analyzing and preventing project failure risks, as well as addressing learning from mistakes over time conducted by the project team including managers. Thus, this research enhances project efficiency and success rates by helping project managers identify and mitigate potential risks before they occur. A novel approach in this research is the integration of machine learning techniques into project management software, particularly in predicting potential risks during the project planning stage based on previous experiences, thereby improving project efficiency and success.

2. Literature Review

2.1. Information Technology Project Management (ITPM)

ITPM (Information Technology Project Management) is a highly important approach to managing projects related to information technology. IT projects often involve the implementation of new systems, software development, or IT infrastructure upgrades, and IT project management helps direct, and manage resources, and ensure that projects efficiently achieve their goals. According to (Pramanik et al., 2022), IT project management plays a key role in managing the uncertainty and complexity of information technology projects, which often involve rapid changes and high business dynamics. The importance of IT project management is also reflected in the research by (Sankaran et al., 2020), stating that the success of IT projects often depends on good managerial skills, the use of appropriate project management methodologies, and a deep understanding of technical aspects. The implementation of methodologies such as Agile or Scrum in IT project management can help project teams adapt to changing customer needs and enhance project flexibility. In this context, IT project management also includes risk management and information security. According to (Shayan et al., 2022), risk management in IT projects involves the identification, analysis, and mitigation of risks that may arise during the project life cycle. Information security is also a primary concern in IT projects due to their vulnerability to evolving cyber threats. In this regard, various factors can lead to failure in Information Technology (IT) project management. One major cause is ambiguity or changes in project requirements (Asadabadi et al., 2020). If customer requirements are not well-defined or undergo significant changes during the project lifecycle, it can result in a mismatch between the project's results and user expectations. Lack of effective communication is also often a cause of failure, both among project team members and between the team and external stakeholders. Furthermore, inadequate risk management can be a crucial factor leading to IT project failure (Sami Ur Rehman et al., 2022). Failure to identify, evaluate, and manage risks appropriately can result in unforeseen issues, such as project delays, cost overruns, or even overall project failure. Errors in planning can also be a significant factor. Inadequate planning, including inaccurate resource estimates or unrealistic schedules, can impede project progress and lead to failure.

According to (Menezes et al., 2019), IT projects fail due to various factors, including Unrealistic Goals and Expectations. Setting overly ambitious or unattainable project goals can lead to failure. Unrealistic expectations regarding project outcomes, timelines, or resource availability can hinder successful project execution. Additionally, poorly defined system requirements also contribute to project failures. Unclear system requirements can result in misalignment in the project deliverables. (Menezes et al., 2019) also state that poor communication and reporting can impact the failure of IT projects. Poor communication between the project team and stakeholders may result in misunderstandings, delays, and misalignment. Regular project status updates are crucial for informed decision-making. Finally, unrealistic estimations, such as overly optimistic estimates of project effort, time, or cost, can result in missed targets. Accurate estimations are essential for project planning and resource allocation. (Menezes et al., 2019) provide insights into risk factors affecting software development projects. They identify and map risk factors, emphasizing those related to software requirements and technical skills. This underscores the need for further research on these factors to reduce project failure rates. According to (Iriarte & Bayona, 2020), three main factors influence the failure of IT project management: poorly defined business case; unclear or poorly defined project benefits; and poor financial management. Inadequate requirement and scope management. Failure due to inadequate requirement definition or uncontrolled scope during the project life cycle. (Bogopa & Marnewick, 2022) state that IT project management failure occurs because of process-related factors, the issues related to project processes, such as unrealistic budgets, schedules, or methodologies, can contribute to project failure. Critical success factors in software development projects highlight various factors influencing success or failure. Process-related factors (e.g., unrealistic budgets, and schedules) and technical-related factors (such as methodology) play crucial roles.

2.2. Project Design Process

2.2.1. Waterfall Model

The project design phase within the waterfall model is a crucial step in the software development life cycle. The waterfall model represents one of the linear and sequential approaches to the software development life cycle. This process begins with the requirements analysis phase, where system requirements are obtained from stakeholders. Subsequently, the project design stage plays a crucial role in developing technical solutions that meet the identified requirements. In the waterfall model design, it is essential to apply efficient and effective design principles. According to (Attaran et al., 2022), the system architecture should be designed to facilitate flexibility, security, and system performance. User interface design should consider end-user needs and provide an intuitive user experience. Additionally, detailed module or component design should consider inter-module dependencies, interface clarity, and the ability to be tested and maintained. The project design stage in the waterfall model not only focuses on technical aspects but also a deep understanding of user needs and expectations (Thesing et al., 2021). By applying good design principles, the project can achieve outcomes that align with expectations and provide significant added value. This process also enables the identification of potential risks that can be addressed before the actual system implementation, reducing the likelihood of significant issues in the subsequent stages of the project life cycle.

The waterfall model has a clear structure and stages in software development, but its drawbacks include a lack of flexibility in responding to changing requirements. This model is linear, requiring the completion of each stage before progressing, making it difficult and expensive to modify if changes arise mid-course. Additionally, the model does not provide deliverables assessable by the client until the final development stage, increasing the risk of project failure and misalignment with user expectations. Its inability to adapt to rapidly changing environments or technologies makes it less suitable for projects requiring quick responses, where Agile models might be more flexible (Bassil, 2022). It can be argued that the biggest drawback of the waterfall model is the difficulty in fully determining project requirements at the project's outset before design or development takes place. Individuals or organizations commissioning the project may not always know from the early stages what they truly want to build. Users and non-technical entities usually intend to commence a project to address a specific issue, yet certain requirements essential for accomplishing that objective may only be fully revealed in the course of the project's design or development stages. Despite the mentioned drawbacks, the waterfall model is effective in projects with a good technical understanding and stable requirement definition from the early planning stages (Bassil, 2022).

2.2.2. Spiral Model

The spiral model is one of the software development models that combines elements from the waterfall lifecycle approach with the concept of risk management. The project design process in the Spiral Model has interesting characteristics, especially in addressing uncertainty and risks associated with software development. According to (Doshi & Jain, 2021), the creators of the Spiral Model, the design stages in this model are integrated into a recurring cycle consisting of four main stages: planning, risk analysis, engineering, and evaluation. This model provides flexibility to adapt to changing requirements or environmental changes during development. Project design in the Spiral Model includes the development of architectural and detailed designs, similar to the waterfall approach. This process considers a deep understanding of user needs, as well as strategies to address risks identified during risk analysis. The main advantage of the Spiral Model is its ability to manage and reduce risks effectively, allowing projects to adapt to changing needs and providing feedback mechanisms that can be applied in each cycle. Although the Spiral Model has many advantages, there are also drawbacks to consider. One of them is its complexity in management. This model requires careful risk management and strong risk analysis capabilities to ensure that each cycle can deliver the desired results (Akinsola et al., 2020). Additionally, development costs and time tend to increase due to the iterative cycles used to address risks and improve product quality. Projects in the spiral model start on a small scale and gradually grow as risks are reduced. This model relies on risk analysis conducted by experts, with costs increasing as development progresses but risks decreasing (Akinsola et al., 2020). Project leaders can control risks by spending more time and money. This model provides valuable feedback in the early stages, allowing project cancellation if it is not feasible technically or financially. The life cycle of the spiral model is flexible and can be adapted to the project's needs.

2.3. Software Tools for Project Management

The development of technology has brought significant changes in the way project management is conducted, and the use of software tools has become crucial in supporting project efficiency and success. Software tools for project management provide platforms that facilitate planning, tracking, and team collaboration. Some well-known tools include Microsoft Project, Jira, Trello, and Asana. In Microsoft Project, it is commonly used for task planning, resource management, and project progress tracking. These tools offer features such as Gantt charts, project reporting, and risk management to help project teams stay organized. Furthermore, cloud-based project management platforms like Jira and Trello enable better team collaboration. Jira provides tools for task tracking, bug management, and project reporting. Trello, on the other hand, offers an intuitive board approach with cards that can be moved by the team to reflect task progress. The use of these software tools not only accelerates project execution but also facilitates communication and coordination among team members, especially if the team is working in a distributed or remote setup (Arefazar et al., 2022). Therefore, with the existence of these software tools, project management can be more efficient and adaptive to changes that may occur during the project lifecycle. Real-time monitoring, accurate reporting, and the ability to respond quickly to changes in requirements or project constraints are some of the main advantages provided by these software tools to project management practitioners. Software tools for project management that do not offer machine learning solutions still play a significant role in supporting users in project planning, execution, and monitoring. However, the importance of project management software with machine learning capabilities is also evident in the improvement of sustainability and maintenance, ensuring adaptability and efficiency throughout the project life-cycle without relying on non-machine learning technology that may limit responsiveness and require higher technical support.

2.4. Machine Learning and Project Management

The implementation of Machine Learning (ML) in project management has become an increasingly important aspect to enhance efficiency, optimize processes, and provide deeper insights. Machine learning falls within the realm of artificial intelligence, allowing systems to glean insights from data and recognize patterns or trends without the need for explicit programming. In the context of project management, machine learning can be used for various purposes. One of its primary uses is in forecasting project time and costs (Mahmoodzadeh et al., 2021, 2022). By analyzing historical project data, such as task durations and associated costs, ML algorithms can make more accurate estimates for new projects. This assists project managers in planning and managing resources more efficiently. Moreover, machine learning can be utilized to identify project risks. By analyzing data from previous projects, ML systems can predict potential risks that may occur in new projects (Gondia et al., 2019). This enables project managers to take preventive actions or adjust project strategies to mitigate potential risks. The implementation of Machine Learning can also enhance project team management (Chattopadhyay et al., 2021). In this regard, ML algorithms can analyze interaction patterns within the team, provide insights into productivity, and offer recommendations to improve overall team collaboration and performance.

2.4.1. Example-Based Learning (EBL)

EBL is an algorithm that compares new problems with those observed during training, unlike other models that require explicit generalization (Hoogerheide et al., 2020). In this research, the system is expected to be able to learn individual user patterns to guide project management. Traditional ML necessitates a substantial volume of training data, while this study aims to provide solutions and identify risks quickly without needing a large dataset. Therefore, the EBL approach in this research is expected to adapt to small and dynamic data sets, according to user needs and project types. This approach also facilitates an active learning process, allowing team members to apply the acquired knowledge to similar real-life situations in their projects. Additionally, EBL can aid in the development of problem-solving skills (Hoogerheide et al., 2020). By understanding how previous projects tackled issues and obstacles, the team can develop the contextual intelligence needed to identify appropriate solutions in their situations. This learning process also stimulates critical and creative thinking, as team members are expected to think more deeply about various possible approaches.

2.4.2. K-Nearest Neighbour (k-NN) Algorithm

This is a technique within ML used for both classification and regression purposes. It forecasts the value or class of new data by considering the majority of k nearest neighbors within the feature space. The steps involve determining the value of k, measuring distances, identifying neighbors, and selecting based on the majority. For distance measurement, k-NN is calculated using:

$$the\ Distance\ measurement = \sqrt{\sum_{i=1}^{n}(x_i - y_1)^2} \qquad (eq.1)$$

Where x_1 and x_2 represent a data point denoting the coordinates of the data point x. Cosine distance, a metric independent of the vector magnitude, is quantified as the cosine of the angle formed between 2 points of x_1 and x_2. Its value ranges between -1 (opposite), 0 (orthogonal), and 1 (same). To train the model and perform k-NN classification, the following algorithm can be used.

k-NN Training algorithm 1: **procedure TRAIN** 2: Let $D_{examples}$ be a list of training examples 3: For each training example $(x, f(x))$, add the example to the list $D_{examples}$
k-NN Classification algorithm 1: **procedure CLASSIFY** 2: Let x_q be a query instance 3: Let x_1 . . . x_n be the k instances from the training-examples nearest to xq by a distance metric D 4: Return $f(x_n) = argmax \sum_{i=1}^{n} f(x_i)$

Fig. 1: k-NN training and classification algorithm

The k-NN algorithm can be adjusted to give varying weights to each query point (x_q), favoring closer neighbors over those that are more distant. When using the k-NN algorithm for classifying a query point with a parameter of n = 5, if three out of the five nearest neighbors belong to distant clusters from the query point (x_q) and the other two neighbors differ significantly, it can result in noisy errors in the classification of that point. This issue can be significantly mitigated by assigning weight contributions to each point based on their distances.

$$w_i = \frac{1}{d(x_q - x_i)^2} \qquad (Eq.2)$$

In this adaptation of the k-NN algorithm, where d represents the selected distance metric, there's no necessity for the classification algorithm to actively search for the k-NN of the query point. This is because the weighting function, which is based on the inverse square distance, significantly reduces their impact. Consequently, the algorithm is now capable of conducting classification using the contributions derived from the entire stored training dataset.

2.5. Regression Model

2.5.1. Logistic Regression (LR)

This is a statistical method employed to model the relationship between a binary dependent variable and one or more independent variables. The main aim is to predict the likelihood of belonging to a particular category by taking into account the values of the independent variables. Using the logit function produces regression coefficients that indicate the impact of changes in independent variables. To calculate predictions from the logistic regression model, the formula is used as follows:

$$y = \sigma(Wx + b) \quad where \quad \sigma = \frac{1}{1+e^{-2}} \qquad (Eq. 3)$$

The outcome is the sum of inputs after being weighted and passed through the sigmoid activation function. The model parameters must be learned during the training process. The sigmoid function confines values within the range of 0 to 1, introducing non-linearity to the model. Optimization methods involving numerical techniques, like *Stochastic Gradient Descent* (SGD), are employed for model training. Logistic regression can reveal the weight contributions of every characteristic within the feature vector for a specific issue. However, it's important to note that the parameters in logistic regression don't directly indicate the significance of associated features. In the absence of a non-linear activation function, logistic regression is equivalent to linear regression. SGD is an iterative optimization method efficiently used in machine learning model training. Unlike conventional gradient descent, SGD uses one randomly chosen sample in each iteration, enabling efficient model adaptation to large datasets. With data shuffling and parameter adjustments at each step, SGD has become a popular method, especially in training deep learning models (Liu et al., 2021).

2.5.2. Momentum

Momentum is a technique in optimization algorithms, particularly used in training models in machine learning, to accelerate convergence. It applies the concept of physical momentum, allowing the model to move faster through flat areas and steep valleys (Kovachki & Stuart, 2021). Momentum updates parameters by combining the current gradient with the previous momentum, providing acceleration for faster convergence. Careful adjustment of the momentum parameter is essential to avoid overshooting the desired global minimum. During the training of neural network models using the gradient descent algorithm, there are sometimes constraints where the gradient updates oscillate between two walls of a gorge, slowing down training or causing model divergence. To address this, a method called update storing is used, wherein all the model parameters are stored at a particular time step, and the subsequent update is a synthesis of the current gradient and the most recent update (Izadi et al., 2020). This approach helps avoid sudden changes in the update direction, speeding up model convergence in deep neural networks. In this case, SGD with momentum is computed using.

$$W_{ij}^{(l)}(t) = W_{ij}^{(l)}(t) - \eta \frac{\delta}{\delta W_{ij}^{(l)}} J(W, b) + \alpha \Delta W_{ij}^{(l)}(t-1) \qquad (Eq. 4)$$

Where $W_{ij}^{(l)}(t), r, and\ \alpha$ represents a coefficient ranging from 0 to 1, regulating the momentum level used in training.

3. Method and Technique

3.1. Adaptive Risk Analysis Tool

The tools developed in this research are Adaptive Risk Analysis Tools, where the first stage involves risk analysis at the key-point level of project managers to identify the issues listed within it.

3.1.1. Data Model

In the development of the Adaptive Risk Analysis Tool, the data model used comprises important components to understand the project structure and associated risks. This data model is designed to store the necessary information for analyzing and managing IT project risks more effectively. Here is a more comprehensive explanation of the components and relationships between the components in the data model: Key Points serve as pivotal points in the project, dividing it into more manageable parts, with each containing information about objectives, deadlines, and associated responsibilities. Tasks represent the activities to be completed within each key point, detailing task descriptions, estimated completion times, and completion statuses. The Challenge Identification model stores information about issues or challenges encountered during the project, linking each challenge to specific key points and tasks, along with details about risk levels and mitigation steps. The Modified Projection Model tracks changes in task completion time estimates made by project managers or team members, aiding in monitoring project progress and identifying potential delays or emerging risks. These components are interconnected, with each key point potentially linked to one or more tasks, and each task possibly associated with specific challenges influencing project risks. Changes in task completion estimates can prompt the identification of new risks or adjustments in mitigation strategies. Understanding these relationships allows the Adaptive Risk Analysis Tool to offer a holistic view of IT project risks, assisting project managers in making informed decisions to effectively manage risks throughout the project lifecycle.

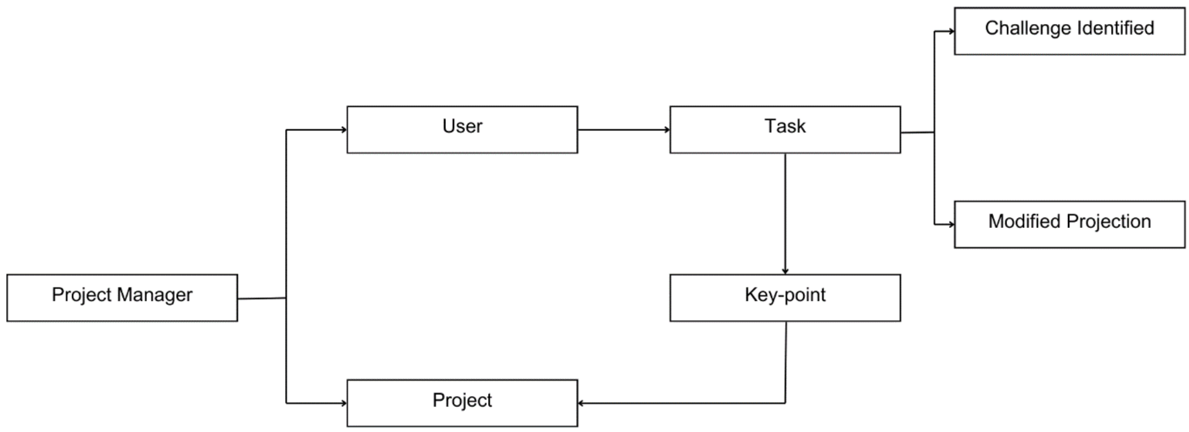


Fig.2.: Data model used in this study

3.2. The Technique used

The technique utilized for creating a resolution to the issue in this research is to create a machine-learning approach to assist managers in project planning. The goal of this approach is to analyze key points for comparison with other key points previously planned by the project manager stored in the system. To do this, features are needed to create vectors for each key point, comprising user count, task count, duration, type, average duration employed, and the sequence of pivotal project key points, standard deviation, and a histogram of task types. The key point type is a categorical feature represented as a one-hot vector to address issues in distance metrics such as Euclidean distance. In the list of key types for a specific project manager, each type is considered different from the others. Representing categories as indices in the list would cause problems in distance calculations, and using a one-hot vector solves this problem by assigning a magnitude of 1 to the dimension corresponding to a specific key point type and 0 to other dimensions.

In the methodology of this research, the selection of specific machine learning techniques, namely Instance-based learning and Regression models, is based on careful consideration. Instance-based learning is chosen due to its characteristic that allows algorithms to learn from specific examples without first building a general model (Geržinič et al., 2023). This is suitable for this research because the developed tool needs to be capable of providing responsive and adaptive risk analysis to real-time project condition changes. With Instance-based learning, the tool can quickly adapt to changing project situations without requiring intensive relearning processes. Meanwhile, regression models are machine learning techniques used to understand the relationship between input and output variables in data. In the context of this research, Regression models can be used to predict project risks based on historical data and previous experiences. By modeling the relationship between relevant variables, the developed tool can provide more accurate risk analysis and assist project managers in making better decisions. The combination of Instance-based learning and Regression models is chosen because they complement each other in providing adaptive and responsive risk analysis. Instance-based learning provides flexibility and speed in adapting to changes, while Regression models provide the ability to understand complex patterns and relationships between variables. Thus, the use of both techniques is expected to enhance the effectiveness of the Adaptive Risk Analysis Tool in managing IT project risks holistically and accurately.

3.2.1.Nearest Neighbor Algorithm

For identifying similar pivotal points and conducting risk analysis in the current project planning, a nearest neighbor algorithm is employed. The outcome of this algorithm comprises the k-NN of the query point, considering these neighbors as key-point vectors. The key points linked to these vectors are examined for related issue occurrences. In our data model, issue occurrences are tied to individual tasks. Given that a key point encompasses multiple tasks, it's possible for a key point to be associated with a specific issue multiple-times. If this is the case, we investigate the most prevalent issue linked to that key point. Additionally, since the system functions as an open web platform utilized by numerous project managers concurrently, specific guidance is offered to users. In this case, a way to represent issue occurrences in the algorithm is needed. In real-world scenarios, various types of issues can occur during the project. In this study, issue categories are denoted by integers, with each integer corresponding to the index of the issue in the user's particular issue list. Nearest neighbor algorithms, such as k-NN, may encounter challenges related to the curse of dimensionality, especially in cases where only a small number of features in the feature vector are relevant to a specific problem. In this study, a project manager might be planning a project and only dealing with project management issues on specific key-point types, such as key points with a specific duration within a time frame. Within the vector representation, the dimension about the period of the key point is depicted as a solitary dimension in the vector space. Simultaneously, other components of the vector encapsulate features that bear low relevance in this particular scenario.

3.2.2.Regression Model (RL)

This approach efficiently enlarges and diminishes the vector space to impact the search for nearest neighbors. As each project manager possesses a unique history and experience, feature weighting is performed individually for each manager. This involves creating a training set comprising key-point vectors planned by the specific project manager stored in the system. The LR model is educated through the Stochastic Gradient Descent algorithm with momentum, enabling real-time convergence in practical scenarios since the number of key points managed by each project manager is typically manageable for swift computation. An alternative method involves testing a logistic regression model integrated with the nearest neighbor algorithm to predict potential project risks based on input key points. The probability of issue occurrences is determined using this approach.

$$P(Y = i|x, W, b) = softmax_i(W_x + b) = \frac{e^{W_i x + b_i}}{\sum j\, e^{W_j x + b_j}} \qquad (Eq. 5)$$

Meanwhile, the top risks, can be predicted using:

$$y_{prediction} = argmax_i\big(P(Y = i|x, W, b)\big) \qquad (Eq. 6)$$

Furthermore, the model will undergo training using the Stochastic Gradient Descent (SGD) algorithm with momentum, enabling it to easily anticipate risks. Not only does the logistic regression model forecast potential risks based on the input key point, but it also engages in multi-categorical classification. As the model learns to anticipate risks based on an input key point, it is the outcome of global optimization, losing information about individual training examples. This presents a limitation in comparison to learning methods based on instances, methods of learning that rely on instances and, in certain scenarios, may offer superior insights to project managers (such as situations where two issues occurred during a specific past key point, suggesting a potential recurrence, but the model lacks information to make such predictions).

3.2.3.Hybrid Approach

The solution that utilizes instance-based learning and the regression model used in this study is a hybrid solution. Even though the regression model no longer has access to the user's past experiences, it is still capable of forecasting the primary risks linked to the input key-point vector. A hybrid approach might entail creating a potential risk table derived from the outputs of both algorithms, with each algorithm's contribution to all risks being evaluated. The predominant risk will be identified as the one with the most substantial contribution.

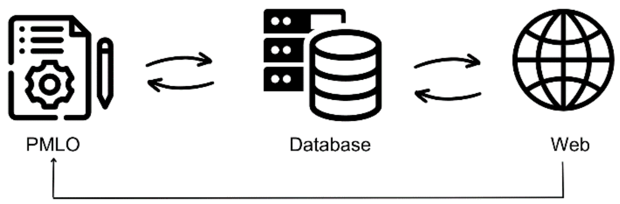
3.2.4.Applying Sanctions to Past Experiences

When designing the project planning system for project managers, it's essential to acknowledge that managers accumulate experience from previous projects. Initially, a project manager may underestimate the time needed for a specific task, but with time, they acquire better estimation skills. These human factors imply that when our system conducts risk analysis by referencing past key-point records, relying solely on the most similar records may not consistently yield optimal results. To address this, penalties can be applied to query results based on the relative time the key point was introduced into the system, giving more weight to newer results, considered more aligned with the current expertise of the project manager. Various approaches can be tested to achieve this goal. Considering the gap between two vectors is essential, taking into consideration not only the time difference but also addressing situations where an older key point resembles what the project manager is currently planning, and there is no recently planned similar key point. The system is anticipated to apply more substantial penalties to key points that are more distant, considering both spatial and temporal aspects.

4. Implementation

4.1. System

The proposed system in this research is divided into three parts: the website platform, the database, and the Project Manager Learner Option (PMLO). In this context, the website is intended for project managers as end-users to manage various ongoing projects, input information, and receive feedback. The database functions as comprehensive data storage, and PMLO serves as a module responsible for handling queries that require machine learning-based responses. The data flow between the designed components is presented in Figure 3. Engagements with the database are interpreted as actions involving reading or writing. Conversely, the communication between the web platform and PMLO functions for generating solutions is described as an activation interaction. In this situation, the platform launches the software, but the outcomes are stored in the database and are not promptly sent back to the web platform.



Figu.3.: data flow architecture

4.2. Web Platform

The first component used is the web platform that allows users to access it anytime, anywhere using various devices with an internet connection. In this component, the web platform has an interface that is easy to understand and use for users. In this instance, users possess the capability to generate, modify, and delete projects, as well as report issues associated with those projects. The web platform is constructed using the Django framework, which adheres to the model-view-template (MVT) design pattern. Django was selected due to its simplicity, robustness as an open-source framework, thriving community, extensive documentation, and a plethora of available plugins for platform extension. It is versatile, supporting multiple database management systems like PostgreSQL, MySQL, and SQLite. Given the platform's objective to aid teams in project management, it is essential to deliver a user interface that is enjoyable, easily comprehensible, and engaging to ensure genuine usage by end-users. To fulfill this requirement, the Twitter Bootstrap framework (getbootstrap.com) for HTML, CSS, and JS is utilized. Bootstrap enables the development of an attractive user interface that adjusts effectively to diverse screen dimensions, including those of mobile devices.

4.3. Database

The second element is the database, serving as a uniform repository for information devoid of distinct features or functionalities. The database plays a pivotal role in this platform as it serves as the storage location for all user-provided data. The web platform is linked to the database, engaging in both reading and writing data. Within this component, there is no necessity for unique platform features to formulate or implement the proposed solution, as the operations involved are straightforward, involving actions like selecting, updating, and deleting for accessing and manipulating data. The database model used as the storage system in this project is a relational database. It is chosen because it allows for practical implementation and has a small scope. In this study, the decision regarding the Database Management System (DBMS) to be employed is once again driven by the need for straightforward infrastructure access. For the development of this platform, SQLite is selected, being an extremely lightweight DBMS that stores information in a single compact file within the OS file system. One notable advantage of SQLite is its simplicity, as it doesn't require special configurations or a dedicated database server. However, it is important to note that SQLite exhibits subpar performance and lacks support for user management, although these drawbacks are inconsequential during the development phase. For the deployment of this platform, PostgreSQL is chosen as the designated DBMS. Transitioning from one DBMS to another won't result in performance implications, as no complex queries are employed that could leverage optimization advantages. All queries are basic and don't involve advanced operators.

4.4. Project Manager Learner Option (PMLO)

The PMLO module is in charge of carrying out all ML operations in the platform. It is linked to both the database and the web platform. When users trigger operations that require machine learning responses (like when a user finishes planning a project key point and requests risk analysis for that key point), the web platform communicates with the PMLO system. Following that, the PMLO system engages with the database, calculates results using the suitable algorithm, and subsequently provides the reply back to the web platform. The reason for retaining a separate PMLO component, both in the system architecture and its implementation, is to guarantee scalability in a production environment. Although the web platform is capable of executing all tasks handled by the PMLO module, the objective is to guarantee that the machine learning component, responsible for resource-intensive calculations, can be effortlessly replicated across multiple servers based on the platform's user load at any given time. In this situation, the web platform will use a load balancer to determine which PMLO server is ready to manage user requests. The PMLO module is coded in the Python programming language, mirroring the language used for the web platform module. This decision is motivated by Python's widespread use in machine learning and data science, which includes several well-optimized and regularly updated ML libraries. Within this module, frequent mathematical calculations are required, and for this purpose, the NumPy Python library is employed. Python is an interpreted language and tends to run slower compared to compiled languages like C and C++, the NumPy library addresses this concern by implementing many operations in C, thereby enhancing execution speed for Python. Additional libraries like TensorFlow and Theano are designed with a focus on efficiency, allowing developers to create a computational graph of mathematical operations, which is then compiled and optimized for both CPU and GPU. The utilization of GPUs provides a substantial performance boost over CPUs in linear algebra operations due to the parallelization of matrix operations.

5. Evaluation

5.1. Evaluation Set and Environment

To evaluate the proposed solution, several datasets are generated. The tasks aimed at supporting project managers in their project planning are influenced not only by the project's characteristics but also by the individual project managers themselves. For instance, an inexperienced project manager might initially miscalculate the time required for a particular task, resulting in delays in key point completion and, subsequently, the overall project. Given that the learning tasks are contingent upon the project manager's experience, it is crucial to construct a dataset that accommodates this variability. This dataset encompasses diverse collections of Projects, key points, and tasks, where each key point may be linked to at least one instance of a Problem, portraying challenges faced during the key point's progression. Assigning values arbitrarily to all attributes of these instance models would render the learning tasks unfeasible and inefficient. Realistic ranges are employed to attribute values to the features of key points and tasks, heightening the dataset's usefulness and relevance. To further enrich the dataset, this study introduces specific biases into the dataset creation algorithm. For example, a key point of type O, with more than n tasks, has a k% chance of encountering an event of type P. Another critical aspect of constructing the dataset is the accumulation of knowledge by project managers over time. In the initial stages, a project manager new to our platform may undervalue the durations of key points but can develop the skill to accurately estimate them through experience. To address these real-world dynamics, a portion of the evaluation set is also generated by accounting for this learning process.

The use of synthetic data in research outcome evaluation has significant limitations that need to be considered when interpreting findings and the generalizability of results. Synthetic data tends to inaccurately represent the complexity and variability present in real-world situations. This can result in developed models being overly idealistic or unable to capture nuances that may arise in actual project environments. Additionally, the use of synthetic data may limit the generalizability of results because models trained only on synthetic data may not be able to address the variations that may occur in different project situations. As a result, the model's ability to be applied widely across various project contexts may be restricted. Moreover, synthetic data is vulnerable to biases and errors that may occur during the dataset creation process. This can affect the quality of the developed model and lead to inaccurate or unreliable results. If the model is only evaluated on synthetic data, decisions made based on the model may not be optimal when applied in real-world situations. Consequently, tools developed based on synthetic data may not be able to provide effective solutions in managing project risks practically. To enhance the generalizability of results, it is important to consider the use of real data or conduct additional validation on various datasets to ensure the reliability and validity of the developed model. Thus, research findings can be more relevant and applicable across a wider range of IT project contexts.

5.1.1. Data Sample

Table 1 provides a breakdown of the dataset utilized for assessing our learning algorithm. Subsequently, this dataset is randomly partitioned into a training set and a validation set with a 70% and 30% ratio. As for the training data samples, they are presented in Table 3.

Table 1.: Dataset for machine learning algorithm evaluation

Dataset	Number of Projects	Number of key points	Number of tasks	Overtime
1	2	18	120	No
2	4	28	150	No
3	4	40	200	Yes

Table 2.: Training data sample

# users	# tasks	Duration	Type	Average duration	Project duration	Order	Standard Deviation	Top. Problem
7	14	40	O	8.14	130	1	6.83	X
3	6	8	O	7.21	130	3	3.32	Y
5	12	19	P	8.11	130	2	7.00	-
3	7	14	O	8.84	200	6	4.62	X
2	9	24	B	12.31	200	1	7.41	Z

The training procedure involves using the fit method from sklearn (scikit-learn.org, 2023) using the Python language, which fits the model to the dataset and subsequently generates predictions. Throughout the training phase, the time taken for the complete training of each model is documented and detailed in Table 2. In the testing phase, the pre-trained models are used to predict labels for the test data, and a comparison between the predicted labels and an analysis of the outcome labels is performed to calculate the classification accuracy provided by each model. Evaluation is performed using the accuracy measurement function.

5.1.2.F1-Score and Accuracy

Dalam penilaian klasifikasi, tiga metrik utama yang digunakan untuk mengukur kinerja model termasuk F1 Score, Recall (Sensitivitas), dan Presisi. F1 Score, sebagai metrik gabungan dari Presisi dan Recall, memberikan gambaran holistik tentang kemampuan model dalam mengklasifikasikan data, terutama berguna ketika ada ketidakseimbangan antara kelas atau ketika kinerja pada kelas tertentu tidak bisa dikorbankan. Sebaliknya, presisi mengukur seberapa baik model dapat mengidentifikasi instansi positif tanpa memberikan hasil positif palsu yang signifikan. Untuk menghitung F1, rumus yang digunakan adalah sebagai berikut:

$$F1\ Score = 2 * \left(\frac{Precision\ x\ Recall}{Precision + Recall}\right)$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

Table 3.: F1-Score result of the algorithms in the classes used in the study

Dataset	**Recall**	**Precision**	**F1 Score**
D1	0.84	0.87	0.75
D2	0.79	0.88	0.81
D3	0.85	0.83	0.73

Accuracy is the proportion of accurate predictions to the overall number of predictions conducted. The formula used to calculate overall accuracy is:

$$Overall\ accuracy = \frac{Number\ of\ correct\ predictions}{Total\ number\ of\ predictions} x100$$

After the model was trained, we made predictions on the test data and contrasted them with the genuine labels to compute the accuracy score. Table 4 and Figure 4 present the accuracy of various applied techniques in predicting the top potential issues in a project key point. The logistic regression model performs admirably, although its efficacy tends to slightly diminish in scenarios with fewer training examples. Logistic regression also exhibits lower performance in datasets where users learn over time, as expected, given that it undergoes training utilizing the complete user history, making it marginally less suitable for real-world scenarios. An additional noteworthy aspect of our solution, particularly within the web platform, is its scalability. We implement a mechanism that links our web platform to a versatile machine learning backend, facilitating easy replacement with minimal developer effort.

Table 3.: The accuracy of each dataset for each algorithm

ML Algorithm	**Dataset**		
	1	**2**	**3**
K-NN	78%	88.7%	32.1%
K-NN+F	100%	100%	43.2%
KNN+C	78%	88.7%	88.7%
KNN+F+C	100%	100%	100%
LR	89.1%	99%	32.1%
Note: F is for feature, C is for classifier, LR is for Logistic regression			

The Adaptive Risk Analysis Tool stands out for its adaptive and responsive capabilities, thanks to the implementation of advanced machine learning techniques, enabling more accurate and effective risk analysis. Compared to traditional methods or existing techniques, the proposed Adaptive Risk Analysis Tool in this research offers a more dynamic solution, reducing human limitations, and enhancing project managers' abilities in managing IT project risks. By leveraging historical data and past experiences, the Adaptive Risk Analysis Tool can provide more detailed and adaptive risk analysis, thus improving efficiency and effectiveness in IT project risk management. Therefore, the Adaptive Risk Analysis Tool offers a more sophisticated and effective approach to IT project risk analysis compared to existing methods or techniques.

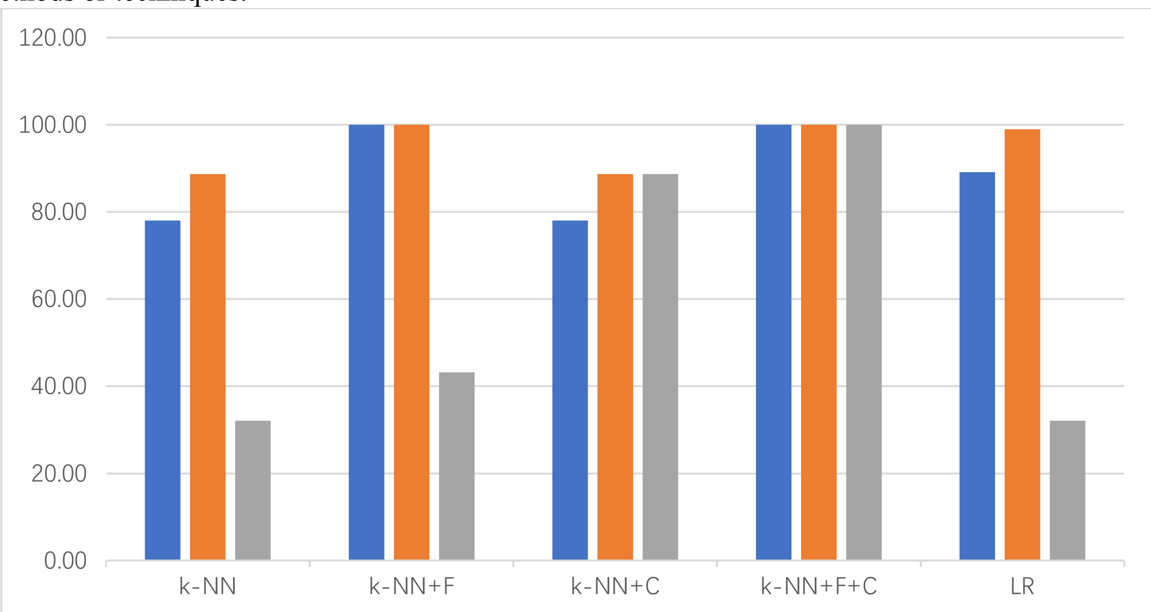


Fig.4.: Results of Accuracy Scores for each Classifier Model used in the study

6. Discussion

This research indicates that the development of a project management system based on machine learning has significantly contributed to enhancing the capabilities of project managers in analyzing and managing project risks. Furthermore, the use of machine learning techniques, such as Instance-based learning and Regression models, has shown promising outcomes in forecasting project risks derived from user experience and work history. The proposed machine learning algorithms have been successfully integrated into a platform accessible for utilization by project managers. Test results indicate that both techniques used are capable of providing satisfactory outcomes when applied to real-world scenarios. The implementation of machine learning techniques in the Adaptive Risk Analysis Tool enables the system to learn from historical data and past experiences, thus offering more accurate recommendations for managing project risks. By leveraging machine learning, this tool can provide adaptive and responsive risk analysis to changes in project conditions, assisting project managers in identifying and addressing risks before they occur. This research also highlights the importance of developing more advanced machine learning models to enhance the accuracy of project risk predictions. Considering factors that can lead to project failures, such as poor planning and inadequate risk analysis, the Adaptive Risk Analysis Tool can be an effective solution in helping project managers manage project risks more effectively.

By leveraging machine learning technology, this system can offer risk predictions based on historical data and user experience, thereby aiding project managers and their teams in enhancing efficiency in project management. The development of an Adaptive Risk Analysis Tool based on machine learning has a significant impact on improving the efficiency and effectiveness of IT project risk management. The integration of machine learning techniques such as Instance-based learning and Regression models allows this tool to provide adaptive and responsive project risk analysis based on historical data and previous experiences. The observed performance in this research can be explained by several key factors. Firstly, the use of machine learning enables the identification of patterns and trends that are difficult to detect manually, thereby allowing project managers to take preventive actions more timely. Secondly, the ability of this tool to provide more accurate and comprehensive risk analysis helps reduce uncertainty in project planning and increase the likelihood of overall project success. Moreover, the integration of machine learning in IT project risk management also provides a deeper understanding of project complexity and helps identify potential areas vulnerable to risks.

There is potential to change the paradigm of project risk management with a more proactive and targeted approach. By using machine learning-based tools such as the Adaptive Risk Analysis Tool, project managers can optimize decision-making, identify potential risks more effectively, and plan more effective risk mitigation strategies. This can not only improve the success of individual projects but also potentially enhance the overall performance of organizations in managing their IT project portfolios. Although these findings are promising, it is important to acknowledge that this research has certain limitations. The use of historical data in developing machine learning models may limit risk predictions for situations that have not occurred before. Therefore, for future research, it is recommended to expand the scope of data, integrate broader contextual factors, and test models in various complex project scenarios. Thus, further research in the development of machine learning-based adaptive risk management tools can significantly contribute to improving the efficiency, effectiveness, and success of IT projects in the future.

7. Conclusion

This research aims to develop an Adaptive Risk Analysis Tool that utilizes machine learning techniques, such as Instance-based learning and Regression models, to provide adaptive and responsive project risk analysis. This tool is designed to assist project managers in identifying, analyzing, and managing project risks more effectively based on historical data and previous experiences. Thus, this research aims to create a solution that can enhance efficiency and effectiveness in project risk management and contribute to improving project success rates through better and more accurate risk analysis. The results of this research indicate that the development of the Adaptive Risk Analysis Tool using machine learning techniques, such as Instance-based learning and Regression models, has significantly contributed to improving the effectiveness of project risk management. This tool is capable of providing adaptive and responsive project risk analysis to changes in project conditions, thereby aiding project managers in identifying and managing risks more effectively. Tests conducted show that both machine learning techniques used can deliver satisfactory results when applied to real-world scenarios. Therefore, the Adaptive Risk Analysis Tool can be a valuable solution for project managers facing complex challenges related to project risks and can help enhance project success rates through more comprehensive and accurate risk analysis.

There are several recommendations for practitioners and organizations:

- **Adoption of Machine Learning-Based Tools**: Organizations can consider integrating machine learning-based tools, such as the Adaptive Risk Analysis Tool, into their project management processes. These tools can enhance risk analysis, improve decision-making, and ultimately lead to more successful project outcomes. Additionally, organizations can invest in training and skill development; practitioners and project managers should invest in training and skill development related to machine learning and data analysis. Understanding how to effectively use these tools can significantly improve project efficiency and risk management.
- **Collaboration and Knowledge Sharing:** Promoting collaboration and knowledge sharing among project teams. By sharing insights and best practices related to risk analysis and project management, organizations can leverage collective expertise to reduce risks and drive project success. Additionally, organizations can implement feedback mechanisms to gather insights from project managers and team members about the effectiveness of machine learning tools in risk analysis. This feedback can help refine tools and processes to better meet organizational needs and requirements.
- **Pilot Testing and Early Evaluation**: Before fully integrating machine learning-based tools into all projects, pilot testing and early evaluation of tool usage can be conducted to assess their impact on project efficiency and risk management. This phased approach can help identify potential challenges and fine-tune tools for optimal performance.

7.1. Future Research Recommendation

Several aspects still need attention to improve the results and impact of the development of project management systems based on machine learning. First, it is recommended to conduct further studies on global user patterns. The use of machine learning algorithms such as k-means Clustering can help group users into several clusters based on their experience and profiles. Such research can provide deeper insights into predicting project risks based on users' experiences with similar profiles, especially for tasks and key points that have not been done before. Second, the development of additional features in the system can be a crucial step. Integration with external data or the development of more advanced machine learning models can enhance the accuracy of risk prediction. The addition of these features can bring new dimensions to project risk analysis and assist project managers in making more informed decisions. Finally, further testing is needed with more complex and diverse scenarios. This testing can assess the reliability of the system under various project conditions and provide a deeper understanding of its performance. Further research in this area will make a significant contribution to the development of project management systems based on machine learning for broader use in various project contexts.

References

Akinsola, J. E. T., Ogunbanwo, A. S., Okesola, O. J., Odun-Ayo, I. J., Ayegbusi, F. D., & Adebiyi, A. A. (2020). Comparative Analysis of Software Development Life Cycle Models (SDLC). *Advances in Intelligent Systems and Computing*, *1224*, 310–322. https://doi.org/10.1007/978-3-030-51965-0_27/COVER

Arefazar, Y., Nazari, A., Hafezi, M. R., & Maghool, S. A. H. (2022). Prioritizing Agile Project Management Strategies as a Change Management Tool in Construction Projects. *International Journal of Construction Management*, *22*(4), 678–689. https://doi.org/10.1080/15623599.2019.1644757

Asadabadi, M. R., Saberi, M., Zwikael, O., & Chang, E. (2020). Ambiguous Requirements: A Semi-Automated Approach to Identify and Clarify Ambiguity in Large-Scale Projects. *Computers & Industrial Engineering*, *149*, 106828. https://doi.org/10.1016/J.CIE.2020.106828

Attaran, H., Kheibari, N., & Bahrepour, D. (2022). Toward Integrated Smart City: a New Model for Implementation and Design Challenges. *GeoJournal*, *87*(4), 511–526. https://doi.org/10.1007/S10708-021-10560-W/FIGURES/10

Bahrini, R., & Qaffas, A. A. (2019). Impact of Information and Communication Technology on Economic Growth: Evidence from Developing Countries. *Economies 2019, Vol. 7, Page 21*, *7*(1), 21. https://doi.org/10.3390/ECONOMIES7010021

Bassil, Y. (2022). Simulating the Software Development Lifecycle: The Waterfall Model. *Infotech Journal Scientific and Academic* , *3*(1), 151–174. https://infotechjournal.org/index.php/infotech/article/view/16/32

Bogopa, M. E., & Marnewick, C. (2022). Critical Success Factors in Software Development Projects. *South African Computer Journal*, *34*(1), 1–34. https://doi.org/10.18489/SACJ.V34I1.820

Chattapadhyay, D. B., Putta, J., & Rama Mohan Rao, P. (2021). Risk Identification, Assessments, and Prediction for Mega Construction Projects: A Risk Prediction Paradigm Based on Cross Analytical-Machine Learning Model. *Buildings 2021, Vol. 11, Page 172*, *11*(4), 172. https://doi.org/10.3390/BUILDINGS11040172

Doshi, D., & Jain, L. (2021). Review of the Spiral Model and its Application. *International Journal of Engineering Applied Sciences and Technology*, *5*, 311–316. http://www.ijeast.com

Gondia, A., Siam, A., El-Dakhakhni, W., & Nassar, A. H. (2019). Machine Learning Algorithms for Construction Projects Delay Risk Prediction. *Journal of Construction Engineering and Management*, *146*(1), 04019085. https://doi.org/10.1061/(ASCE)CO.1943-7862.0001736

Hoogerheide, V., Roelle, J., Scheiter, K., & Mayer E Y W O R D S, R. K. (2020). Example-based Learning: New Theoretical Perspectives and Use-Inspired Advances to a Contemporary Instructional Approach. *Applied Cognitive Psychology*, *34*(4), 787–792. https://doi.org/10.1002/ACP.3706

Iriarte, C., & Bayona, S. (2020). IT Projects Success Factors: A Literature Review. *International Journal of Information Systems and Project Management*, *8*(2), 49–78. https://doi.org/10.12821/ijispm080203

Izadi, M. R., Fang, Y., Stevenson, R., & Lin, L. (2020). Optimization of Graph Neural Networks with Natural Gradient Descent. *Proceedings IEEE International Conference on Big Data*, 171–179. https://doi.org/10.1109/BIGDATA50022.2020.9378063

Kovachki, N. B., & Stuart, A. M. (2021). Continuous Time Analysis of Momentum Methods. *The Journal of Machine Learning Research*. https://doi.org/10.5555/3546258.3546275

Liu, J., Sun, Y., Gan, W., Xu, X., Wohlberg, B., & Kamilov, U. S. (2021). SGD-Net: Efficient Model-Based Deep Learning with Theoretical Guarantees. *IEEE Transactions on Computational Imaging*, *7*, 598–610. https://doi.org/10.1109/TCI.2021.3085534

Mahmoodzadeh, A., Mohammadi, M., Daraei, A., Farid Hama Ali, H., Ismail Abdullah, A., & Kameran Al-Salihi, N. (2021). Forecasting Tunnel Geology, Construction Time and Costs Using Machine Learning Methods. *Neural Computing and Applications*, *33*(1), 321–348. https://doi.org/10.1007/S00521-020-05006-2/METRICS

Mahmoodzadeh, A., Nejati, H. R., & Mohammadi, M. (2022). Optimized Machine Learning Modelling for Predicting the Construction Cost and Duration of Tunnelling Projects. *Automation in Construction*, *139*, 104305. https://doi.org/10.1016/J.AUTCON.2022.104305

Menezes, J., Gusmão, C., & Moura, H. (2019). Risk Factors in Software Development Projects: a Systematic Literature Review. *Software Quality Journal*, *27*(3), 1149–1174. https://doi.org/10.1007/S11219-018-9427-5/TABLES/13

Pramanik, D., Mondal, S. C., & Haldar, A. (2022). A Framework for Managing Uncertainty in Information System Project Selection: an Intelligent Fuzzy Approach. *Big Data and Information Theory*, 103–112. https://doi.org/10.4324/9781003289173-11

Sami Ur Rehman, M., Thaheem, M. J., Nasir, A. R., & Khan, K. I. A. (2022). Project Schedule Risk Management Through Building Information Modelling. *International Journal of Construction Management*, *22*(8), 1489–1499. https://doi.org/10.1080/15623599.2020.1728606

Sankaran, S., Vaagaasar, A. L., & Bekker, M. C. (2020). Assignment of Project Team Members to Projects: Project Managers' Influence Strategies in Practice. *International Journal of Managing Projects in Business*, *13*(6), 1381–1402. https://doi.org/10.1108/IJMPB-12-2018-0285/FULL/XML

Schmidt, J. (2023). Mitigating risk of failure in information technology projects: Causes and mechanisms. *Project Leadership and Society*, *4*, 100097. https://doi.org/10.1016/J.PLAS.2023.100097

Schweikl, S., & Obermaier, R. (2020). Lessons from Three Decades of IT Productivity Research: Towards a Better Understanding of IT-Induced Productivity Effects. *Management Review Quarterly*, *70*(4), 461–507. https://doi.org/10.1007/S11301-019-00173-6/METRICS

scikit-learn.org. (2023). *scikit-learn*. https://scikit-learn.org/stable/

Shayan, S., Pyung Kim, K., & Tam, V. W. Y. (2022). Critical Success Factor Analysis for Effective Risk Management at the Execution Stage of a Construction Project. *International Journal of Construction Management*, *22*(3), 379–386. https://doi.org/10.1080/15623599.2019.1624678

Skare, M., & Riberio Soriano, D. (2021). How Globalization is Changing Digital Technology Adoption: An International Perspective. *Journal of Innovation & Knowledge*, *6*(4), 222–233. https://doi.org/10.1016/J.JIK.2021.04.001

Thesing, T., Feldmann, C., & Burchardt, M. (2021). Agile versus Waterfall Project Management: Decision Model for Selecting the Appropriate Approach to a Project. *Procedia Computer Science*, *181*, 746–756. https://doi.org/10.1016/j.procs.2021.01.227

Venturini, F. (2022). Intelligent Technologies and Productivity Spillovers: Evidence from the Fourth Industrial Revolution. *Journal of Economic Behavior & Organization*, *194*, 220–243. https://doi.org/10.1016/J.JEBO.2021.12.018

Westenberger, J., Schuler, K., & Schlegel, D. (2022). Failure of AI projects: understanding the critical factors. *Procedia Computer Science*, *196*, 69–76. https://doi.org/10.1016/J.PROCS.2021.11.074