IEEE *Access*
Multidisciplinary : Rapid Review : Open Access Journal

Systems Thinking in Software Projects - An Artificial Neural Network Approach

Zorica M. Mitrović[1], Aleksandar M. Rakićević[1], Member, IEEE, Dejan Č. Petrović[1], Marko M. Mihić[1], Jovana Đ. Rakićević[1], and Elena T. Jelisić[1]

[1]Faculty of Organizational Sciences University of Belgrade, Belgrade 11 000, Serbia

Corresponding author: Zorica M. Mitrović (e-mail: zorica.mitrovic@fon.bg.ac.rs).

ABSTRACT The authors discuss why the current conceptual base of project management research and practice continues to attract criticism since it does not adequately address the complexity that leads to software-project failure. To do so, the study explores systems thinking and artificial neural networks to shed light on complexity in software-project behavior using nonlinear functional relationships between critical success factors and project success to utilize their connectedness as an approach in order to create project-outcome prediction models. The artificial neural networks were used to create two project-outcome prediction models: one for a binary classification task to discriminate failed from successful projects using a multi-input-single-output configuration and one for a multi-task binary classification to discriminate success from failure in multiple project-success dimensions using a multi-input multi-output configuration. The results yielded high-performance values for a binary classification task, performed to predict overall project success, and slightly lower performance values for the multi-task binary classification, which was also performed to predict success in project-success dimensions. It was found that the nonlinear behavior of critical success factors may be used to create prediction models, by embedding equifinality and connectedness constructs that prove to be useful to understand projects as complex, multi-loop, and nonlinear systems. Further research is needed to investigate the causality between critical success factors in order to explore the possible propagation of critical success factors within a project system network and its implications on project success.

INDEX TERMS Artificial neural networks, critical success factors, project success, prediction models, systems thinking.

I. INTRODUCTION

The software industry has a history of recording a high rate of failure in projects [1]. The CHAOS report indicates that only 32% of software projects in 2009 were successful; complexity was considered to be the main reason behind the failure [2]. A literature review shows that the overall challenge facing research in this field is rethinking project success and its critical factors through addressing complexity as well as broader project conceptualization and context (e.g. [3] - [6]).

Project-failure research has had its origins in the software industry since the latter half of the twentieth century when the term "software crisis" was first noted in 1968 [7]. Around the same time, the concept of critical success factors (CSFs) was created by Daniel [8] in his seminal work "Management Information Crisis". Popularized by

Rockart [9] as an approach to improve information-system strategic planning, it was later implemented in project management practice as a means to improve project performances [10]. With the significant growth of project management research, project success and CSFs have earned their place in research traditions as one of the nine major schools of thought in this field [11]. The more recent focus of research in the success school has been placed onto identifying success as a multidimensional construct, including dimensions such as project management and product success [10], as well as business and future success [1]. Similarly, existing research on CSFs, in addition to their categorization for software projects, have focused on recognizing relationships between varied groups of factors and project success [10] as well as on creating contingency-fit models for traditional plan-based and agile

methodologies [12]. However, in most cases, existing models are created using linear regression, neglecting non-linear relationships that exist between CSFs and project success. The very process of statistical analysis includes the identification of the factors most significant, disregarding "less significant ones" and their relationships with factors previously identified as significant. Doing so results in statistical models excluding factors that could indirectly affect project success. Thus, a major criticism of existing models is insufficient focus on complex functional relationships between individual factors and project success, in particular as concerns the nonlinear nature of such relationships. An additional issue is contingency-fit models failing to recognize that the same final state of project success is achievable from dissimilar initial conditions when using multiple combinations of CSFs.

A review of the literature shows that substantial progress has been made with the application of systems thinking as a holistic discipline that recognizes projects as interconnected technical and social factors producing behavior which otherwise cannot be predicted by simply aggregating the behavior of the projects' elements in isolation [13]. Over the last two decades, there has also been a considerable amount of research into intelligent systems and the application of machine learning (ML) as highly effective techniques to address problems of complexity (e.g. [14], [15]). In regards to prediction models, more recent research has included the application of artificial neural networks - ANNs [16], Bayesian classifiers (e.g. [17] - [19]), genetic algorithms (e.g. [19], [20]), support vector machine [20] and logistic regression models [21]. Nevertheless, ANNs are the most used techniques for software-project issues (e.g. [22], [23]). As may be anticipated, a number of studies have found that ANNs outperform traditional regression models (e.g. [24] - [26]). Furthermore, one of the major benefits of using ANNs is their ability to learn and communicate past knowledge [27] and even include valuable information from unsuccessful projects [28]. However, in the field of project management, most studies have only focused on predicting whether a project is risky or not [16], as well as on efforts estimation (e.g. [29] - [33]), duration estimation [33], and cost estimation (e.g. [34]- [36]). Despite the overwhelming interest, few researchers have yet to address the complex relationship between software-project CSFs and project success. Thus, the authors intended to conduct research that would provide MIMO (multi-input multi-output) and MISO (multi-input single-output) ANN models capturing complex relationships between CSFs and project success directly as based on real data from past software projects.

The main purpose of this study is to explore the nonlinear behavior of CSFs to predict the project success of software projects measured as a multidimensional construct, as well as to search for a management tool to balance success across dimensions of software projects.

This study focuses on making ANN-based project outcome prediction models, taking into account that the same state of the project could be achieved using disparate sets of CSFs.

Specifically, the research aims to answer two questions:
1) "How does the MISO model perform when identifying overall project success vs. failure?"-which will be addressed through a binary classification task to discriminate failed from successful projects;
2) "How does the MIMO model perform when identifying success vs. failure of a project-success dimension?"-which will be answered through a multi-task binary classification to discriminate success from failure in multiple project-success dimensions.

As an end effect, the authors hypothesize that using CSFs allows for creating prediction models to provide valuable data to project managers and improve software development projects and software project success. This study will use the nonlinear behavior of CSFs to bridge the gap of using only contingency and linearity constructs through a systems-thinking approach in order to attain better performance for complex software development projects.

Bearing in mind that it is one of the few papers to do so, this study will contribute to the theory of project management using the nonlinear behavior of CSFs to predict software-project success.

The paper is laid out in 6 sections. Section 2 reviews the key terms and theories utilized in the research. Section 3 covers a literature review of critical success factors. Section 4 details the method used, consisting of a data collection approach and ANN approach. In Section 5 the most significant results are presented and discussed. Section 6 concludes the paper and presents implications for project management theory and future research.

II. BACKGROUND

A. PROJECT SYSTEM AND PROJECT SUCCESS DEFINITION

This paper uses systems thinking to define projects as complex, multiple-loop, non-linear, and open systems (e.g. [37], [38]). Systems thinking allows for an understanding of the project as an open system of interconnected, technical, and social factors that produce a system's behavior [13]. Such a holistic viewpoint of a projects' definition addresses criticism of closed boundaries and the linear causality present in conventional project management theory.

In the context of project management, the boundary of a system is the scope of the stakeholders' interests and is mutable as the scope of interests changes [39]. Indeed, a systems-thinking approach uses the construct of causal connectedness to examine boundary management [38], which is based on the notion that any act of change could unpredictably influence the relationships between stakeholders and the project itself [40]. It is here argued that

software development projects are characterized by high levels of boundary complexity since they tackle both internal and external stakeholders with their diverse characteristics and values; i.e., software-project success is achieved by balancing internal and external stakeholders' interests. There are, therefore, at least three separate dimensions of project success: 1) project management, 2) product and 3) strategic (e.g. [1], [10]).

The first dimension of project management success also referred to as the level of project success, is measured using internal efficiency measures of project management, such as budget, time, and similar performance measures for internally set constraints [1]. The second is product success, focused on delivering benefits to internal and external stakeholders (e.g. [10], [41], [42]). The third, strategic project success, concerns business efficacy and strategic effectiveness, as well as the ability to generate future success for the company (e.g. [1], [10], [42], [43]).

B. CRITICAL SUCCESS FACTORS

CSFs are several key areas which, if successfully managed, will lead to better organization performance (e.g. [9], [42], [44], [45]). In the context of project management, CSFs are characterized as those factors that directly influence and increase the probability of project success (e.g. [41], [46] - [49]. The discourse to prove this definition is composed of several distinct arguments.

In systems-thinking theory "factors" could be interpreted as supernatural, organizing principles [50]. In project systems, if values govern the actions of the project stakeholders, factors will govern the outcomes of those actions. The likelihood to achieve the desired project outcomes depends on the stakeholders' values and the factors linked to the desired behavior, thereby accounting for the difficulty in predicting behavior in any complex project. However, a project where every stakeholder understands and is committed to factor-based behavior will enable a system to achieve the desired project outcomes under a far higher likelihood. On the other hand, projects that are not grounded in success factors will result in a wide variety of project stakeholders' personal interpretations on how to behave.

Further, in systems thinking theory, CSFs enable the anticipation and rationalization of temporary collective behavior of a project system, by accounting for the interaction that exists between factors. Beyond simply identifying CSFs, there is also a need to incorporate their interaction in predicting project outcomes.

Finally, adopting an open-systems approach in project management includes generic constructs such as equifinality [38], which refers to a system being able to achieve the same final state from dissimilar initial conditions and do so in diverse manners (e.g. [51] - [53]). According to Child [54], equifinality implies that strategic choice is available to decision-makers when creating organizations to achieve high performance - particularly in complex organizations. Thus, this construct is necessary to effectively manage projects in non-linear unpredictable environments [38]. For the purposes of this study, the authors use equifinality to explore the phenomenon where the same level of performance is achievable from multiple contexts, using different sets of factors.

C. ARTIFICIAL NEURAL NETWORKS

ANNs are biologically inspired computational models mimicking the information-processing structure of the human brain through a similar network. In general, ANNs are considered to be powerful non-procedural and adaptable prediction tools, consisting of a number of interconnected neurons with linear and nonlinear activation functions [55]. In order to predict a project's system behavior, ANNs use an iterative training process to learn linear and nonlinear relationships between project system elements. One of the most important advantages of ANN applications is their ability to capture the underlying patterns of available datasets and model complex relationships between input and output variables, while not having a complete understanding of the complexity of functional relationships between variables. Even though prediction models are based on large amounts of data, the researchers often have to deal with much smaller data sets and create the most accurate model from the available data [56]. The generic nature of the methodology and lack of any limitation to the type of dataset or the number of input and output variables makes an ANN approach highly applicable in software projects.

The structure of an ANN consists of an input layer, an output layer, and one or more hidden layers. The input layer introduces the data to the network, while the output layer is the final layer that has a set of values to represent the network's output. The hidden layers are placed between both the input and the output layer to perform the calculations and produce internal representations of input patterns. ANNs are typically configured based on the choice of the number of the input and output nodes, the number of hidden layers, the choice of activation functions, and other design parameters. A trial and error approach is generally applied in order to achieve optimal architecture. Such parameters change throughout the experimental phase to select the optimum ANN architecture, taking into account an acceptable compromise between predictive accuracy and processing time. In nonlinear input-output mapping, the ability of hidden neurons is considered necessary to extract higher-order statistics. Using only a few neurons in the hidden layer might disable networks in order to capture the nonlinear trends in the dataset, thereby resulting in the lower predictive accuracy of the model. Nonetheless, too large a number of hidden neurons could result in the time-consuming process of training and ultimately over-fitting the training dataset.

IEEE *Access*

III. LITERATURE REVIEW

Hitherto research reported in the literature involving the application of ANNs to project success concerns the objectives of project outcome prediction and CSF identification, as found in [57] - [65]. The main challenge is that an accurate prediction of project success has proven difficult due to the complex relationship between project success and CSFs, resulting in the application of multiple-input single-output models, where output variables are simple success measures. However, multiple studies have shown that success needs to be understood as a multidimensional construct in software projects to appreciate the complex dynamics and impacts within them [1]. In order to better comprehend the complexity that exists in software project systems and to address project success, there is a necessity to rise beyond a single output model and to bring forth multiple output models to predict success in multiple dimensions of software project success.

In the study of software projects, ANNs are generally recognized for their ability to produce reasonably accurate predictions in situations where there are nonlinear relationships between inputs and outputs [29]. As a non-parametric method, ANNs are easy to use because they extract implicit knowledge from past experience without involving project managers in complex subjective judgments [61]. However, software development is filled with such judgments. For example, the success of software projects clearly depends, to a significant extent, on the quality of the communication, yet communication quality is rarely measured and based on easily available data. While the assessment of such factors is the subject of CSF studies, the evaluation of CSFs on software projects with regard to ANNs can help to overcome the disadvantages of using only easily available data collected from past projects. In addition, the use of CSFs as input variables can add considerable value and insight into the modeling process of software project management. The key to modeling usable project outcome prediction models is to move beyond the limits of easily available data and to conceive of information as it relates to key areas of activity in which favorable results are absolutely necessary for project success.

Although a large number of studies categorize CSFs into key areas, suggesting alternative frameworks (e.g. [10], [12], [48], [66]), there is no general categorization of CFS for software projects [67]. To illustrate, many authors have categorized CSFs for either agile or traditional software projects. Authors such as Chow and Cao [66] focused their research on agile software projects and categorized 109 CSFs into five categories: organizational factors, people-related factors, process factors, technical factors, and project factors. In addition, Ahimbisibwe et al. [12] identified 37 CSFs for both agile and traditional software development projects and grouped them into organizational, team-related, customer-related, and project factors.

Sudhakar [10] proposed a conceptual model examining 80 CSFs, composed of seven categories of factors: communication, team, organization, environment, technical, project management, and product. Sudhakar [10] also recognized that CSFs of one category affects another, suggesting that, in addition to relationships between CSFs and project success, there are also relationships between the CSFs themselves. Based on the findings in the literature (e.g. [10], [12], [66]), the study presented here categorizes CSFs into five groups (Table 1): customer factors, technical factors, project management factors, organization factors, and team factors.

TABLE I
LIST OF CRITICAL SUCCESS FACTORS

Factor category	Critical Success Factors
Customer factors	Customer involvement
	Customer support
	Customer leadership
	Customer experience
Technical factors	Requirements
	Specification change
	Development methodology
	Documentation
	Testing
Project management factors	Planning
	Control
	Constraints
Organization factors	Top management support
	Clear vision
	Organizational culture
Team factors	Internal project communication
	Team-member competencies
	Training and education
	Turnover impact
	Motivation

Factors, such as customer involvement, customer experience, and customer support [12], are all variables related to project customer characteristics. Customer involvement has been reported by numerous empirical studies in software development projects to be one of the most significant factors that influence project success (e.g. [66], [68] - [71]). Furthermore, customer experience as a predictor of success refers to customer familiarity with this type of application and their understanding of the problem they wish to solve using a software development project (e.g. [12], [69], [70], [72], [73]). Studies have argued that customer leadership characteristics refer to the acceptable level of a business domain, conceptual skills, and personal characteristics that could increase the chances of project success ([12], [74]).

Project management CSFs as predictor variables refer to adequacy or flexibility in planning and controlling practices used in a project. In general, planning positively influences the process' performance ([68], [70]); however, innovative projects such as software projects, require flexibility, in which there are focal goals to decide on a course of action [38] - the same applies to task control and coordination. Hence, if there is low uncertainty in a project, traditional

approaches planning, monitoring, and controlling the project's scope, resources, budget and milestones should work well. However, under circumstances of complexity and uncertainty, responding to change is one of the key adequacy principles of project management, which exclusively needs flexibility in planning and control practices. On the other hand, the predictability of project success is connected to the availability of data on a project's status that allows for project managers to track a project's progress and arrive at management decisions correctly [21]. In this context, a predictor of success is a variable that takes into account items such as adequacy of monitoring and reporting progress as well as the availability of data.

The literature considers adequate development methodology, documentation, and testing to be determiners of success in both plan-driven and agile methodologies as relates to the characteristics of the development environment [75]. In addition, technical factors as predictors of project success include variables that are related to requirements. Clear requirements and specifications are the most important predictor of project success [48]. Predictors of risk, on the other hand, may include such items as having clear and detailed requirements, both project members and customers equally understanding and sharing a commitment to the requirements, as well as appropriately managing specification changes [21].

Team factors contain variables pertaining to communication, competency level, training and education, turnover, and motivation. The success of software development projects is argued to depend greatly on effective communication and feedback (e.g. [12], [48], [66], [70]). Internal project communication is defined as a practice that increases information exchange and cohesion among development team members [12]. In general, internal project communication enhances information and knowledge sharing between the members of the project team thereby increasing team performance. Ali et al. [76] reported collaboration, coordination, and cooperation to be highly correlated with the strategic aspects of project success, such as new business opportunities. Together with maintaining commitment and motivation of project teams (e.g. [48], [66], [77], [78]), team capability, competences and skills are reported as the CSFs and backbone of the software industry (e.g. [66], [69], [73], [75], [78]). Finally, staff turnover is argued to influence project operations (e.g. [79], [80]), particularly its strategic aspects through the loss of future potential of already capable team members and the higher costs incurred when hiring replacements.

According to Howell et al. [81], since parent organizations play a dominant role in the project's environment, it is reasonable to view organizational variables as predictors indicating project behavior. The literature review shows that top management support is a crucial factor for the success of software development projects and, as such, is a significant predictor of project performance (e.g. [12], [82]-[85]). Further, multiple authors suggest that organizational culture has a positive effect on project success, especially in agile project management (e.g. [69], [71], [78]). Another essential factor is a clear vision and mission [78]. Kaufman et al. [86] assert as such, stating that an organization's objectives are crucial factors in strategic planning and thinking in project management.

Based on the literature, our study continues CSFs long place in software projects. The study aims to use CSFs as success predictors to create project outcome prediction models through the conjoint use of expert judgment and non-parametric methods such as ANNs.

IV. METHOD

The method applied in the study included conjoint use of expert judgment and non-parametric method to create a multi-organization data set and perform experiments on evaluating CSFs on software projects using a set of ANN models, MISO and MIMO. The study designed a method to answer its two research questions of how the MISO model performs when identifying overall project success vs. failure as well as how the MIMO model performs when identifying success vs. failure in multiple project-success dimensions.

As to create multi-organization derived data, the study has applied a survey-based empirical approach to collect data on past software projects, as well as a key-informant approach to identify the software-project management professionals to match the criteria of the study; i.e., experts who possess sufficient experience as project managers on software projects. As the most crucial element of the study was accessibility to software project professionals, convenience sampling as a type of non-probability sampling method was deemed best to apply.

Using project management practices from a country-by-country perspective has been shown to be extremely valuable in the literature for creating project outcome prediction models [87]. Therefore, in order to access software project management professionals who possess sufficient experience in carrying out software projects, the study used a database limited to software project managers that carried out projects for software companies headquartered in Serbia, created by the Project Management Centre at the University of Belgrade. In addition, with more than 2,500 active software companies [88] and ranking as 12th out of the 131 world's economies in the export of ICT services (% of total trade) [89], Serbia is among the most globally relevant countries for knowledge sharing in ICT and makes it a fruitful source to carry out such research.

In order to address the research questions and perform experiments on evaluating CSFs on software projects, the study proposed an ANN-based approach to perform:
1) a binary classification task to discriminate successful projects from failed ones;

2) a multi-task binary classification to discriminate success from failure in multiple project-success dimensions.

Accordingly, the study created a set of multilayer neural networks, MISO and MIMO, wherein input layers correspond to inputs of the project system–CSFs, and output layers representing the output of the project system – project success.

In order to establish a basis to draw a comparison between MISO and MIMO, the approach was designed to use the same CSFs as input variables and output constructs created from project success dimensions (PMS, PS, and SS). To use average values in creating output variables, the study proposed Cronbach's alpha coefficient to validate project success dimensions. For the binary classification tasks, output constructs of both MISO and MIMO were labeled as binary class outputs. The study employed k-means clustering, therefore, in order to ensure a balanced dataset in terms of distribution between the two classes, as well as to construct a single output model (MISO) based on all three dimensions of project success. As a result, MIMO output variables were labeled using success criteria, while the same for MISO used clustering results. Furthermore, the study used these binary-class output variables constructed for MISO, to perform the evaluation of input variables through a two-step evaluation process. First, by identifying whether input variables (CSFs) are non-discriminatory towards successful and failed projects, followed by identifying whether there is a nonlinear relationship between inputs (CSFs) and outputs (project success) using threshold values for maximum correlation and minimum significance.

In order to create an ANN design for both MISO and MIMO, the study utilized a trial and error approach. Accordingly, a series of experiments needed to be performed in order to test the training algorithms, network architectures, preprocessing methods, transformation functions, as well as validation methods. The objective was to evaluate multiple ANN designs in terms of accuracy and processing time, then to decide on those most optimal for MISO and MIMO.

A. DATA COLLECTION APPROACH
Some general information on the data collection approach used in the study is here provided.

1) QUESTIONNAIRE
The authors used a comprehensive survey questionnaire to collect information specific to CSFs and performance measures on past software projects. As suggested by Sekaran [90], the first section of the questionnaire covered question-items related to CSFs (Table 1), identified from the critical literature review of our study. The items were measured using a five-point Likert scale (1 - strongly disagree, to 5 - strongly agree), as the authors used a measurement scale created by Ahimbisibwe et al. [12].

The second section of the questionnaire was used to measure performance for the three dimensions of project success: 1) project management success - PMS, 2) product success - PS, and 3) strategic success - SS. The list of the 27 performance measures grouped by project success dimensions is presented in Table 2. These items were also measured on a five-point Likert scale (1 - strongly disagree to 5 - strongly agree).

2) PARTICIPANTS
The aim of the data collection was to accumulate information on how software projects were executed using the practitioners' experience as pertinent for the actuality of projects, as suggested by Cicmil et al. [91]. In order to obtain the needed information, the key informant approach was used and project managers were targeted as appropriate respondents. Such an approach is more time-intensive but allows for researchers to ask additional questions to clarify all aspects of the questionnaire.

Owing to the fact that the most critical element was accessibility, a non-probability sampling approach was applied. With the participating project managers being personally contacted, almost a 100% return was achieved. The average working experience of the project managers involved in this study was 14 years and almost half of the participants possessed more than 20 years of experience in the field.

TABLE II
LIST OF PERFORMANCE MEASURES [1]

Project Management Success	Product Success	Strategic Success
Project Cost	Benefits	Return on investment
Project time	Satisfied objectives	Revenue measures
Full Scope	Satisfied project requirements	New business opportunities
Milestones	Stakeholder Satisfaction	New markets
Functionality	Client satisfaction	Derived products
Number of defects	Product/result usable	Competitive advantage
Use of resources	Product/result in use	New or expanded core competency/ capability
Agreed scope changes	Product/result useful	Improved processes
Change requests	Fulfilled expectations	Enhanced reputation

3) SAMPLE

Data on 47 software projects were collected from 35 separate private firms by involving projects of a wide range of contextual and developmental characteristics. Consistent with prior findings on prediction models based on multi-organizational data sets [87], the sample organizations included in this study were both small and large in scale, ranging from 5 employees in the smallest to more than 5,000 in the largest. As concerns their organizational capacities, the targeted firms run both small and large-scale projects, as reflected in the projects' team size, duration, and budget. A breakdown of the companies and projects is provided in Table 3. As may be expected, smaller-sized organizations are more likely to operate smaller-scale projects (2-5 team members) under budgets not exceeding USD 100,000, whereas larger-sized organizations operate larger-scale projects that have over 50 team members, are implemented in a timeframe between 24 and 36 months (or exceeding this timeframe) and are worth at least one million USD but may also be more than 10 million USD.

TABLE III
SAMPLE INFORMATION

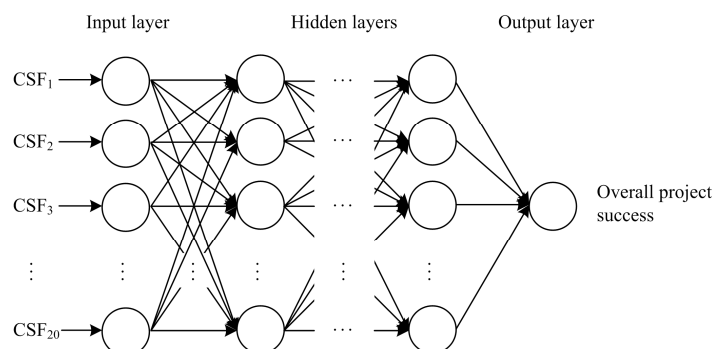|	Software company size				
	<10	11-50	51-100	101-500	> 500
Project budget (USD)					
< $100,000	6	6	5	2	0
$100,000 - $1M	4	4	3	4	3
$1M - $10M	0	2	1	3	2
>$10M	0	0	0	1	1
Project duration (in months)					
< 6 months	2	2	1	1	0
6-12 months	3	4	4	3	0
12-24 months	3	4	2	3	3
24-36 months	1	1	2	1	2
> 36 months	1	1	0	2	1
Development team size					
2-5	4	1	3	2	0
6-10	5	7	5	1	1
11-20	1	2	1	2	3
21-50	0	2	0	3	1
>50	0	0	0	2	1

FIGURE 1. A MISO model is composed of an input layer where the multiple inputs are CSFs, an output layer where the single output represents the overall project success and includes one or more hidden-layers in between.
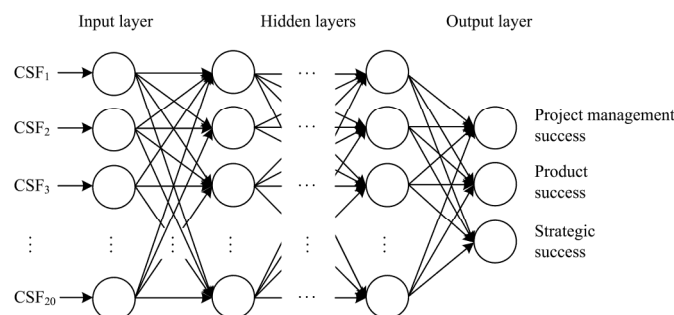

FIGURE 2. A MIMO model is composed of an input layer where the multiple inputs are CSFs, an output layer where the multi outputs represent project-success dimensions and includes one or more hidden-layers in between.

B. ARTIFICIAL NEURAL NETWORK APPROACH

1) MIMO AND MISO MODELS

To create project outcome prediction models, the authors used two ANN models: 1) MISO is a multilayer neural network architecture composed of a multiple-input and single-output configuration (Fig 1); and 2) MIMO is a multilayer architecture composed of multiple-input and multiple-output configurations (Fig 2). While, in the both models, each neuron in the input layer represents an input in the project system, such as a CSF, each neuron in the output layer corresponds to the output of the project system, such as the project success class. Further, for the MIMO model, it was decided to create output variables

corresponding to each dimension of the success construct, and for the MISO model, to use the overall success as a construct corresponding to a single output variable.

2) PROJECT SUCCESS

In order to validate all three dimensions of project success (PMS, PS, and SS), Cronbach's alpha value was calculated for each, whereby a set of 9 specific dimension item-questions were tested to find if they do measure the same construct (Table 2). The PMS showed a satisfactory internal consistency of α= 0.936. PS did so also at α= 0.965 as well as the SS with α= 0.951. Considering that internal consistency is high, dimension scores are calculated as an arithmetic mean of question-items for dimensions of success (Fig 3).
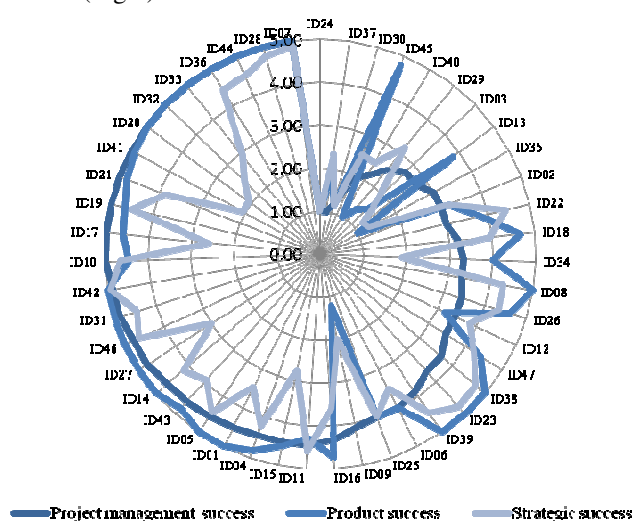
FIGURE 3. The scores represent the relative positions of multiple success dimensions as comparative measures for each project (ID1-ID47), showing that projects may vary in their success scores in project-success dimensions.

Further, K-means clustering was applied in order to take into account the three project-success dimensions in order for every project to decide on overall project success. The results for initial and final centers are presented in Table 4.

TABLE IV
THE RESULTS OF K-MEANS CLUSTERING

	Initial cluster centers				Final cluster centers			
	Cluster				Cluster			
	1	2	3	4	1	2	3	4
PMS	4.3	1.9	1.0	5.0	4.3	2.7	2.2	4.9
PS	5.0	4.8	1.0	5.0	4.7	3.9	1.3	4.9
SS	4.9	2.5	1.0	2.1	4.3	2.2	1.9	2.6

While cluster one consists of projects that have the best scores for all three project-success dimensions (PMS, PS, and SS), cluster four consists of projects with scores higher than 4.00 for two out of three dimensions (PMS and PS); the projects in these two clusters could be considered successful. The other clusters two and three consists of projects of scores lower than 4.00 for all three project-success dimensions; accordingly, these two clusters consist

of projects that could be considered to be "failed" projects.

3) LABELING OUTPUT VARIABLES – MIMO

In order to label output variables for MIMO, success criteria were applied based on scores for each success dimension, wherein scores greater than 4.00 were labeled as 1 (success), if-else 0 (failure).

4) LABELING OUTPUT VARIABLES – MISO

In order to label output variables for the MISO model, projects were assigned to clusters based on success scores in all three dimensions, wherein projects with scores greater than 4.00 in at least two dimensions out of three were labeled as 1 (successful projects), if-else 0 (failed projects).

5) INPUT VARIABLES

In order to determine the effectiveness of a selected set of input variables, we performed a two-step evaluation process as presented in Fig 4.
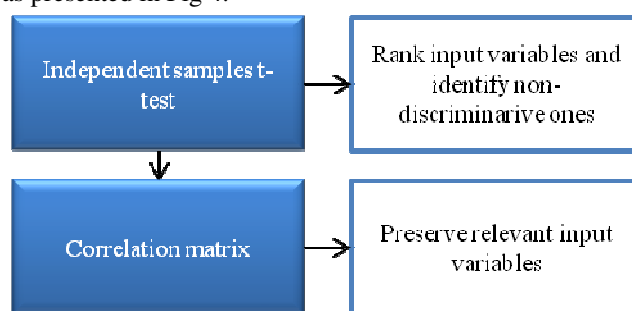
FIGURE 4. The two-step statistical analysis for input variable selection.

An independent sample t-test to each input variable was applied to test whether the data means are equal in the two classes of projects labeled as 1 (successful) and 0 (failed). The labels were obtained from the overall success classification of projects as either successful or failed, as presented above for the MISO model. A criterion of 0.05 p-values was used to rank input variables and identify non-discriminative ones.

A correlation matrix was applied to explore threshold values for maximum correlation and minimum significance. The correlation threshold included a higher bound that is the highest allowed correlation coefficient with respect to classes (0.6 in this study). The variables with the lowest p-value were considered relevant to be preserved among highly correlated variables.

5) ANN DESIGN

All models were back-propagation feedforward networks. The optimal ANN design for both MIMO and MISO models was determined through experiments to evaluate the training algorithms, network architectures, preprocessing methods, and transformation functions, as well as the Leave-p-out method (LPO) as related to their accuracy (the ratio of correct predictions) and processing time. While not all results of the ANN design task were significant, their overall direction showed trends that might be insightful

regarding parameters more likely to impact the model performance in similar settings. Figures 5 through 12 visually illustrate the performance values, in terms of minimum, median and maximum values of accuracy and processing time, of separate ANN designs for both the MIMO and MISO models.

For both the MIMO and MISO models, two training algorithms were implemented: 1) Resilient Backpropagation Algorithm-RPA and 2) Scaled Conjugate

Gradient Algorithm-SCGA. For the MIMO model, in terms of the maximum value of accuracy and minimum processing time, Fig 5 shows that RPA outperforms SCGA. For the MISO model, Fig 6a shows that both training algorithms have similar maximum values of accuracy, but there was a significant difference in the minimum and median values in favor of SCGA. Conversely, RPA is significantly more efficient in comparison to SCGA in terms of processing time (Fig 6b).
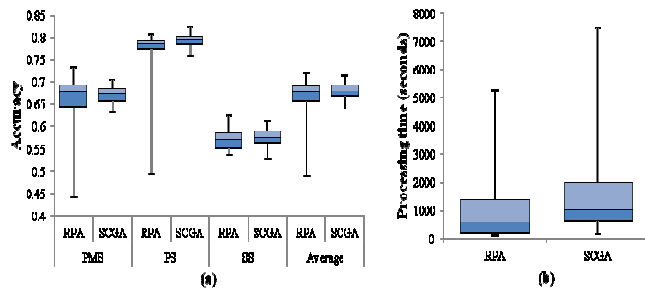
FIGURE 5. MIMO model - the performance of the Resilient Backpropagation Algorithm (RPA) and Scaled Conjugate Gradient Algorithm (SCGA) in a) accuracy and b) processing time.
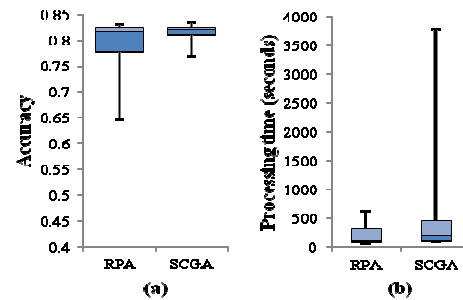
FIGURE 6. MISO model - the performance of the Resilient Backpropagation Algorithm (RPA) and Scaled Conjugate Gradient Algorithm (SCGA) in a) accuracy and b) processing time.
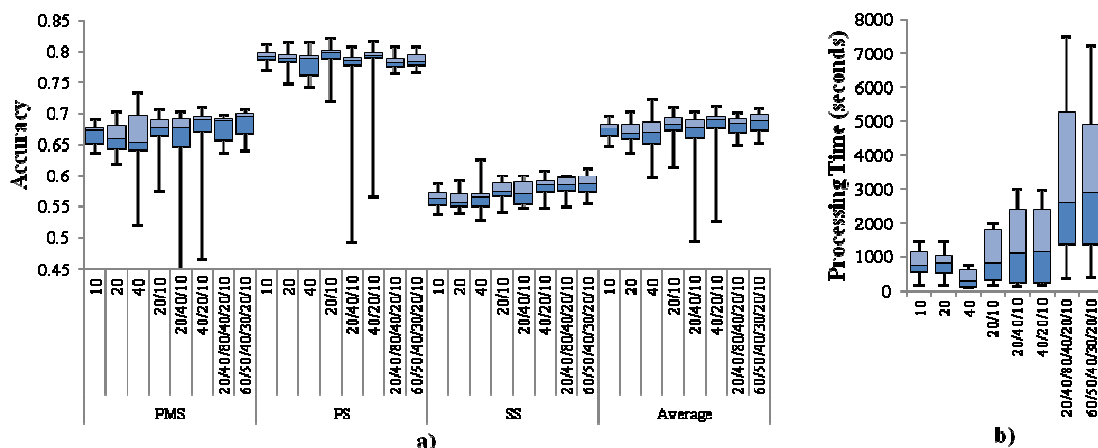
FIGURE 7. MIMO model - the performance of the multiple ANN architectures in a) accuracy and b) processing time.

For both MIMO and MISO models, the experiments were set to simulate gradual network-structure growth by adding new network elements and included models with one-, two-, three- and six-hidden-layer configurations: 1) the one-layer network included experiments with 10, 20, and 40 neurons; 2) the two-layer network included a 20/10 configuration; 3) the three-layer network included two types of configurations - 20/40/10 and 40/20/10; 4) the six-layer-network included experiments with 20/40/80/40/20/10 and 60/50/40/30/20/10 configurations. Fig 7 shows the best performing network for the MIMO model to be a one-layer network containing 40 neurons in a

hidden layer, followed by a three-layer network with a 40/20/10 configuration. For the MISO model, there was no significant difference in the maximum value of accuracy (Fig 8), apart from the minimum values of the accuracy of the six-layer-networks that were considerably lower in comparison to others. Finally, for both the MIMO and MISO models, the number of neurons is indicated to influence processing time.

In addition to the experiments set using raw data, experiments were set with two types of normalization [0, 1] and [-1, 1] for the preprocessing methods of the MIMO model (Fig. 9). Concerning both accuracy and

preprocessing time in comparison to the data of [0, 1] normalization and raw data, the preprocessing method of [-1, 1] normalization shows better performances for all three criteria. For MISO, the authors applied only [-1, 1] normalization.

Both the MIMO and MISO models used tansig, purelin, and satlin transformation functions to create a combination of hidden and output layers; for instance, tansig-satlin combines tansig for hidden layers and satlin for the output

layer. For the MIMO model, in terms of both the maximum value of accuracy and processing time, satlin-satlin and tansig-satlin outperform tansig-purelin (Fig 10). For the MISO model, using multiple combinations of transformation functions yield intriguing results. Fig 11 shows that both satlin-satlin and tansig-satlin combinations outperform the combination of tansig-purelin in relation to their accuracy, particularly their processing time.
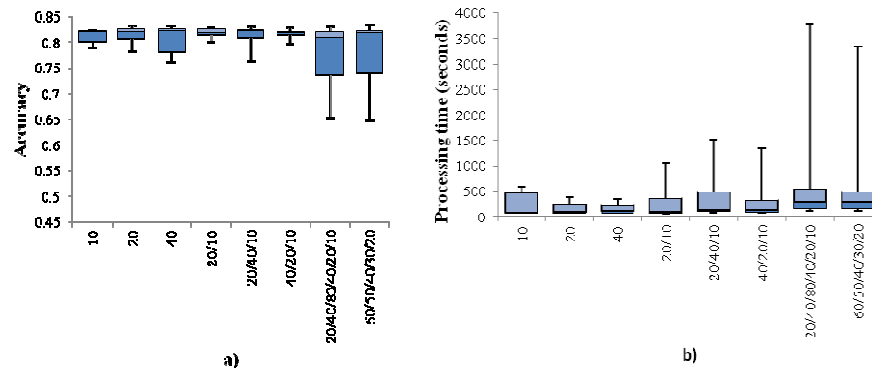

FIGURE 8. MISO model - the performance of the multiple ANN architectures in a) accuracy and b) processing time.
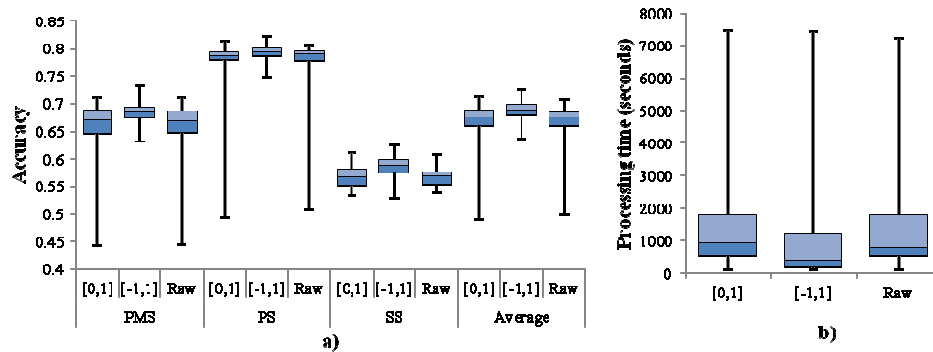

FIGURE 9. MIMO model - the performance of [0, 1] and [-1, 1] normalization, as well as raw data (i.e., data with no normalization) in a) accuracy and b) processing time.
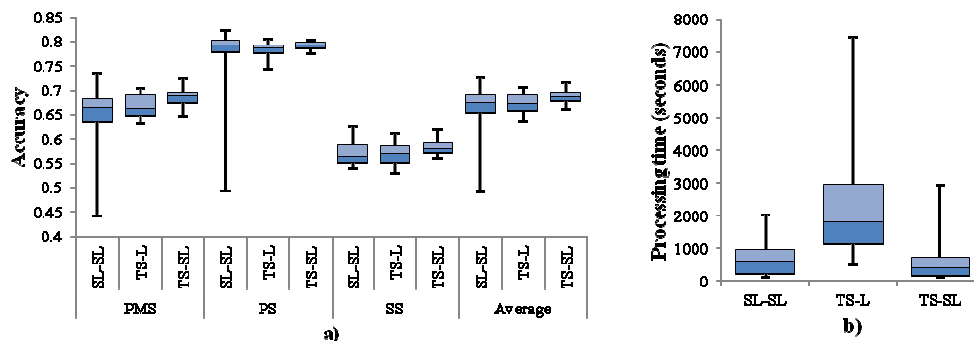

FIGURE 10. MIMO model - the performance of SL-SL (satlin-satlin), TS-L (tansig-purelin), and TS-SL (tansig-satlin) in their a) accuracy and b) processing time.
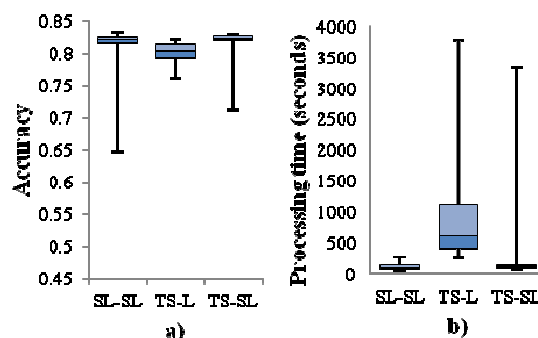
FIGURE 11. MISO model - the performance of SL-SL (satlin-satlin), TS-L (tansig-purelin), and TS-SL (tansig-satlin) in their a) accuracy and b) processing time.
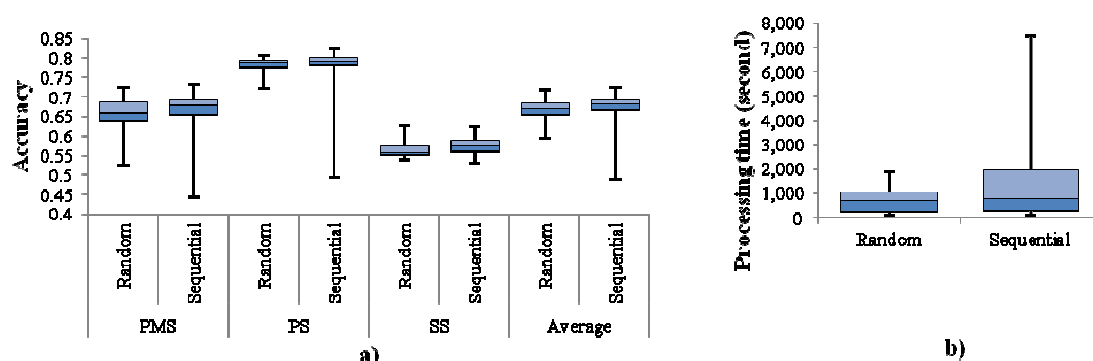

FIGURE 12. MIMO model - the performance of the sequential and random LPO (p=7) in its a) accuracy and b) processing time.

The LPO method was applied by removing seven samples (p=7) sequentially or randomly from the complete set as the test set and using the remaining as the training set. For the MIMO model, in terms of the maximum value of accuracy, the random method was better than the sequential (Fig 12), even if slightly less time-efficient. For the MISO model, only the sequential LPO method was applied.

V. RESULTS AND DISCUSSION

A. PERFORMANCE EVALUATION

In order to report results on MISO and MIMO performances, the study uses the following metrics: 1) Accuracy (ACC); 2) False-positive ratio (FPR); 3) Precision (PREC); 4) Recall (REC); 5) F- Score; and 6) Area under the ROC curve (AUC). While accuracy is a better performance measure to reflect on positive and negative classes correctly predicted, it is still unable to provide information on how the model performs on negative classes incorrectly predicted as positive classes (as provided by FPR). Further, precision and recall are preferable metrics when predicting positive classes is important, yet the F-Score is a better metric when incorrectly predicted classes are too costly; particularly when the benefit of correctly predicted classes is less important than the cost of incorrectly predicted classes. To

complete a thorough performance evaluation, the study uses AUC to provide information on the quality of the model's predictions. All these metrics are equally important to indicate how well models perform.

TABLE V
THE BINARY CLASSIFICATION TASKS – MISO AND MIMO.

	ACC	FPR	PREC	REC	F-SCORE	AUC
MISO	0.83	0.13	0.87	0.88	0.88	0.84
MIMO	0.72	0.26	0.74	0.79	0.76	0.75
PMS	0.73	0.24	0.76	0.82	0.79	0.79
PS	0.80	0.13	0.87	0.87	0.87	0.81
SS	0.62	0.42	0.58	0.67	0.62	0.64

Research question 1: "How does the MISO model perform when identifying overall project success vs. failure?"

In order to provide an answer to this question, a binary classification task was performed and the results are presented in Table 5. With an accuracy of 0.83, the results show that MISO is highly capable of predicting overall success vs. failure in software projects. With a precision of 0.87, the probabilities that the model correctly predicts positive classes (success) are more than acceptable, whose sensitivity shows a value of recall at 0.88. The model's satisfactory performances are confirmed by the F- Score (0.88) and FPR (0.13), demonstrating there to be a low probability of false alerts to be raised by failed projects that are incorrectly identified as successful. Certainly, the high

performance of tradeoffs between sensitivity and specificity in a binary classifier is confirmed by the AUC of 0.84, which demonstrates the model's overall high predictive quality.

Research question 2: "How does the MIMO model perform when identifying success vs. failure of a project-success dimension?"

In order to provide an answer to this question a multi-task binary classification was performed, the results of which are presented in Table 5. With an acceptable average accuracy of 0.72, the result showed that MIMO performs well when predicting success vs. failure in multiple project success dimensions. However, with an accuracy of 0.73 in PMS, 0.80 in PS, and 0.62 in SS, the results also do indicate there to be a significant difference between performances in predicting success vs. failure in multiple project success dimensions. Its probability to correctly predict success in PMS and PS is fairly high, with a precision of 0.76 for PMS and an excellent precision of 0.87 for PS, but unacceptably low for predicting outcome in the SS dimension. The sensitivity of the MIMO showed reasonably good performances at an average recall of 0.79, as well as the overall quality of the models' predictive capability with an average AUC of 0.75. However, the results for the FPR show a high probability of incorrectly classifying failure as success with a fall-out of 0.42 in the SS dimension, which is better, but still a high fall-out of 0.26 in the PMS dimension in comparison to the 0.13 in the PS dimension. Finally, with a reasonable F-Score of 0.76, the results show that MIMO does well on average in terms of the test's accuracy.

B. COMPARISON WITH PREVIOUS METHODS

One of the main challenges of project managers is the early prediction of project outcomes. If project managers are able to anticipate the success or failure in disparate project success dimensions, they would be able to better decide on key areas of action and better prepare for the future. Accordingly, this study used ANNs to improve project success and project success prediction with regard to complex relationships existing between input and output variables of the software project system. As a result, ANN models showed satisfactory performance when modeling nonlinear relationships existing between CSFs and an overall project success based on the multidimensional construct (MISO) or multiple project success dimensions in software projects (MIMO).

In comparison to the traditional parametric method using historical project data to create linear models based on regression analysis, ANNs do not depend on data distribution or assume a fixed structure of a model [62]. These models, owing to their robust nature, are more accurate than linear models are (e.g. [61], [65]). Furthermore, where there are nonlinear relationships between input and output variables, as well as where there

is less information available on the relationships between variables, parametric methods fail to provide an accurate prediction.

In addition to parametric methods, expert judgment is another traditional method used to predict project outcomes. In this paper it is found that expert judgment could be used for variable evaluation, adding value to the quality of data collected in key project management areas. Although the complexity of software projects makes using expert judgment difficult to accurately predict the future, the expert judgment itself may be used in conjunction with other methods, such as has been used as an adjustment factor in parametric models [65] or to create Bayesian models [92]. Further, metrics selection based on the expert judgment was a method used in one of the first applications of artificial intelligence to project success, the model of which was created by applying a Bayesian classifier to estimate the project outcome (e.g. [17], [93]). Therefore, the conjoint use of expert judgment and non-parametric methods is already recognized in the literature as it brings value to software project management through interaction between project managers' complex subjective judgments and ANNs as non-procedural and highly adaptable prediction tools capable to learn linear and nonlinear relationships between project system elements.

C. ADVANTAGES OF THE METHOD

The application of the MISO and MIMO models allowed the authors to address the criticism of closed boundaries through the multidimensional construct of project success. The authors argue that software development projects are characterized by high levels of boundary complexity and that project success is determined by the diverse interests of both the internal and external stakeholders [38]. Hence, the output variables for both the MISO and MIMO models were based on a project-success multidimensional construct that included the three project-success dimensions of PMS, PS, and SS (e.g. [1], [10]). Further, the application of the MISO and MIMO models allowed the authors to use equifinality instead of contingency theory and to address the criticism of the linear causality present in conventional project management theory through nonlinear relationships between CSFs and project success. Equifinality implies the availability of strategic choice to project managers to effectively manage projects in complex and unpredictable environments [38]. The diversity of CSFs used as input variables proves that a project system is able to achieve the same final state, success or failure, from distinct initial conditions and using multiple combinations of CSFs.

The majority of research on project-outcome prediction models thus far has focused on creating appropriate methods to identify the characteristics or similarities of software projects used as predictors of project success (e.g. [17], [19], [94]-[96]). This study selected CSFs, as predictors of project success, from organizational,

customer, team, technical, as well as project management perspectives and used arguments provided by Sudhakar [10] that CSFs are interconnected, both directly and indirectly affecting project success. However, due to the complexity of project systems, existing models of CSFs that have been researched as linear models are unable to address factor interactions, owing to complex functional relationships tending to invalidate the assumptions of most statistical methods [38]. Therefore, rather than a few measurable variables in linear relationships, a network of CSFs is a more appropriate design to comprehend the behavior of the project system owing to the numerous interactions and dependencies among system elements in complex project systems. In such networks, CSF interaction is determined by the causality between multiple factors and as interaction is oriented from one factor to another, the precedence of the relationship between factors is crucial when defining the nature of CSF interactions, as well as the dynamism of CSFs [64].

The literature review shows that companies use AI to deliver value in various business domains (e.g. [97]-[99]). However, due to the complexity of the interaction between humans and AI (e.g. [100]-[102]), some aspects of AI application still need a human expert to reflect on the problem domain [103]. Therefore, the study finds that the conjoint use of ANNs and CSFs obtained through the expert judgment method contribute to performance improvements in terms of decision-making in software projects.

D. LIMITATIONS

The output variables for both the MISO and MIMO models were based on a project-success multidimensional construct that included three project-success dimensions PMS, PS, and SS (e.g. [1], [10]). However, one challenge facing this study is that obtained datasets in practice are often imbalanced, which results in lower performance of classification tasks for underrepresented classes in the training data (e.g. [104], [105]). For this reason, k-means clustering was applied to create a more balanced distribution between two classes for the MISO model, labeled as successful or failed projects. In contrast, the authors decided to apply success-score criteria to the MIMO model in order to label project-success dimensions (scores > 4.00 = 1 - success, if-else 0 - failure). Consequently, one limitation for this model was the SS dimension's imbalance in the distribution between the two classes, yielding a worse performance of classification tasks than for the PMS and PS. A solution to this problem could be advanced sampling techniques that generate new data in under-represented classes based on current data [105].

In order to create a prediction model by capturing the nonlinear relationships between variables, input-variable analysis for the MISO model was achieved through the application of an independent sample t-test and correlation matrix to select input variables by exploring threshold values for maximum correlation and minimum significance. By doing so, the study overcomes the linearity assumption of statistical methods based on the selection of the most significant factors, while disregarding "less significant ones" and their relationships with factors previously identified as significant, as well as includes factors that could both, directly and indirectly, affect project success. However, the same input-variable analysis process for the MIMO model was not applied, as it was based on input variables previously selected for the MISO model. A separate input-variable analysis for the MIMO model would require further exploring relationships between CSFs and project-success dimensions, possibly resulting in the necessity to utilize different sets of input variables for every project-success dimension and to create three separate MISO models.

E. THREATS TO VALIDITY

This study tests the variables that contribute to the project success proposed in the literature to construct a theory related to predicting the project outcome of software projects. Hence, this subsection includes a construct, conclusion, as well as internal and external validity analysis to assess how well a research method identifies concepts of interest and what the implications are for the measures used, its findings, and conclusions. In order to make sure that there is a relationship between theory and investigation, the authors performed a comprehensive review of the literature on CSFs for software projects, finding that the majority of the identified CSFs were covered by the survey's questionnaire. In addition, in order to validate all three dimensions of project success (PMS, PS & SS), the Cronbachs' alpha value was calculated for each.

All participants were asked to respond to at least one questionnaire on finished projects in which they had been project managers. In order to prevent hypothesis guessing, those surveyed remained uninformed about the main objectives of the study. Since the models in this study are tested using data on already finished projects, the relevance of such models could be identified as a threat to its validity. Therefore, the authors plan to test the models using on-going projects at the time of research in future studies.

In order to make sure that there is a relationship between the method and the outcome, all participants were selected due to their sufficient experience in software projects. Basic instructions to answer the questions were provided to ensure reliability and consistency in the responses.

A threat to internal validity is that participants are prone to positivity bias thereby being less critical when reporting any negatives in successful projects or being too critical when reporting any negatives in failed projects. The respondents were therefore asked to provide information on projects under no specific instructions to choose a successful or failed project.

Regarding external validity, this study sample was gathered through a convenience sample; all the participants from Serbia were software project management professionals holding positions in the industry. Particular considerations of external validity were a reflection of the practitioners sampled; as such, the authors strove to best guarantee that participants were involved in professional software development across several industries, project sizes, and project types, including in-house and outsourced development projects. Apart from access to the final study results, there were no other incentives offered. Finally, this study shows that the models are general enough to provide acceptable predictive results even though they were built with project data from multiple companies and projects that are both in-house and outsourced.

VI. CONCLUSION

In conventional project management theories, when faced with practical problems, the focus tends to be placed on statistical models based on few measurable variables in a linear relationship, which thereby fails to take into account that projects are complex, multi-loop, and nonlinear systems. However, the consistently high rate of failure in software development projects leaves one to question these statistical methods.

This study, therefore, advocates that software development projects are open systems of interconnected CSFs which produce a project system's behavior. Results from this study confirm that CSFs are capable of doing so. Furthermore, it is found that ANN-based project-outcome prediction models (MISO and MIMO) may be created by using nonlinear functional relationships between CSFs and project success. The same results of the study indicate that equifinality should be embedded in project management theories to help understand that projects are capable of achieving the same final state from disparate initial conditions. A systems-thinking approach, therefore, is able to contribute to balancing the interests of project-success dimensions from multiple stakeholders by implementing causality and equifinality in contemporary project management methods.

Implications for project management theory and future research could be summarized in conclusions drawn from the results. Firstly, in complex systems, such as software projects, results show that equifinality (flexible project management strategy based on multiple combinations of CSFs) and boundary management (causal relationships between CSFs and project-success dimensions) may be highly significant to a successful practice and more significant than contingency fit models. Secondly, even though the emphasis on nonlinear relationships between CSFs and project success leads to a better understanding of the behavior of the software project system, these arguments should not be perceived as the exclusion of linear models or conventional project management

methods. This study challenges only the emphasis placed on linear models and other conventional project management theories such as contingency theory, proposing that these should be improved by introducing constructs such as equifinality and causality. Research from other project management fields, such as risk project management, has recognized systems thinking to be an approach that allows projects to be more successful by better understanding the complexity and causal connectedness between risks [37]. Further research is needed to quantify the interaction of CSFs and explore possible propagation of specific factors within a project success network in order to identify the most influential factors in the project system as a whole.

Therefore, this study challenges the theory of conventional project management, unable to appreciate the complex nature of software development [106]. The findings of this study provide evidence running counter to contingency fit models of CSFs for software development projects [12]. The authors make distinct contributions to the theory of software project management (e.g. [1], [10]) by providing systems thinking approach, which may prove to lead to a more successful practice.

Overall, this study encourages further research into the field of software projects to discover how systems thinking could be better applied by software project managers. There is quite strong support for the idea that a combination of CSFs and artificial intelligence tools, such as ANNs could add more value in software project management, especially through better understanding complex relationships existing between CSFs and multiple dimensions of project success. Since this study does confirm the value of such an approach, it will deliver greater confidence to software project managers in their continuing to investigate project outcome prediction models through the interaction of humans and AI.

REFERENCES

[1] D. Dalcher, "Rethinking Success in Software Projects: Looking Beyond the Failure Factors," in *Software Project Management in a Changing World,* G. Ruhe and C. Wohlin (eds). Berlin, Heidelberg, Germany: Springer, 2014, pp. 27-49.

[2] T. Clancy, "The Standish group report - CHAOS report 2014," The Standish Group International, Inc., Dennis, MA, 2014. [Online] Available: https://www.projectsmart.co.uk/white-papers/chaos-report.pdf.

[3] K. Jugdev, J. Thomas and C. Delisle, "Rethinking project management – Old truths and new insights," *Int. Proj. Manag. J.,* vol. 7, no. 1, pp. 36-43, 2001.

[4] R. Turner, M. Huemann, F. Anbari, and C. Bredillet, *Perspectives on Projects.* London, UK, and New York, USA: Routledge, 2010.

[5] T. Dille and J. Söderlund, "Managing inter-institutional projects: the significance of isochronism, timing norms and temporal misfits" *Int. J. Proj. Manag.,* vol. 29, no. 4, pp. 480-490, May 2011, 10.1016/j.ijproman.2011.02.007.

[6] P. Svejvig and P. Andersen, "Rethinking project management: A structured literature review with a critical look at the brave new world," *Int. J. Proj. Manag.,* vol. 33, no. 2, pp. 278–290, Feb. 2015, 10.1016/j.ijproman.2014.06.004.

[7] P. Naur and B. Randell, "Software engineering - Report on a conference sponsored by the NATO Science Committee," NATO

Scientific Affairs Division, Brussels, Belgium, 1969. [Online] Available: http://homepages.cs.ncl.ac.uk/brian.randell/NATO/nato1968.PDF

[8] R. Daniel, "Management information crisis," *Harv. Bus. Rev.,* vol. 39, no. 5, pp. 111-121, Sept./Oct. 1969.

[9] J. F. Rockart, "Chief executives define their own data needs," *Harv. Bus. Rev.*, vol. 57, no. 2, pp. 81-93, Mar. 1979.

[10] G. P. Sudhakar, "A model of critical success factors for software projects," *J. Enterp. Inf. Manag.,* vol. 25, no. 6, pp. 537-558, Oct. 2012, 10.1108/17410391211272829.

[11] J. Söderlund, "Pluralism in project management: Navigating the crossroads of specialization and fragmentation," *Int. J. Manag. Rev.,* vol. 13, no. 2, pp. 153-176, Oct 2010, 10.1111/j.1468-2370.2010.00290.x.

[12] Ahimbisibwe, R. Cavana, and U. Daellenbach, "A contingency fit model of critical success factors for software development projects: A comparison of agile and traditional plan based methodologies," *J. Enterp. Inf. Manag.,* vol. 28, no. 1, pp.7-33, Feb. 2015, 10.1108/JEIM-08-2013-0060.

[13] M. Emes and W. Griffiths, "Systems thinking: How is it used in project management?" Association for Project Management, Apr. 2018. [Online]. Available: https://www.apm.org.uk/media/17308/systems-thinking_final.pdf

[14] S. Berlin, T. Raz, C. Glezer and M. Zviran, "Comparison of estimation methods of cost and duration in IT projects," *Inf. Softw. Technol.*, vol. 51, no. 4, pp. 738-748, Apr. 2009, 10.1016/J.INFSOF.2008.09.007.

[15] C. Lopez-Martin and A. Abran, "Applying expert judgment to improve an individual's ability to predict software development effort," *Int. J. Softw. Eng. Knowl. Eng.,* vol. 22, no. 4, pp. 467–483, Jun. 2012, 10.1142/S0218194012500118.

[16] W. M. Han, "Discriminating risky software project using neural networks," *Comput. Stand. Inter.*, vol. 40, pp. 15-22, Jun. 2015, 10.1016/j.csi.2015.01.001.

[17] S. Abe, O. Mizuno, T. Kikuno, N. Kikuchi, and M. Hirayama, "Estimation of project success using Bayesian classifier," in *Proc. ICSE'06*, China, 2006, pp. 600.

[18] O. Mizuno, T. Hamasaki, Y. Takagi, and T. Kikuno, "An empirical evaluation of predicting runaway software projects using Bayesian classification," in *PROFES 2004 Lect. Notes Comput. Sci.*, vol. 3009, Kansai Science City, Japan, 2004, pp. 263-273.

[19] F. Reyes, N. Cerpa, A. Candia-Vejar, and M. Bardeen, "The optimization of success probability for software projects using genetic algorithms," *J. Syst. Softw.*, vol. 84, no. 5, pp. 775-785, May 2011, 10.1016/j.jss.2010.12.036.

[20] M. Y. Cheng, Y. W. Wu and C. F. Wu, "Project success prediction using an evolutionary support vector machine inference model," *Automat. Constr.*, vol. 19, no.3, pp. 302-307, May 2010, 10.1016/j.autcon.2009.12.003

[21] Y. Takagi, O. Mizuno and T. Kikuno, "An empirical approach to characterizing risky software projects based on logistic regression analysis," *Empir. Softw. Eng.,* vol. 10, no. 4, pp. 495–515, Oct. 2005, 10.1007/s10664-005-3864-z.

[22] J. Wen, S. Li, Z. Lin, Y. Hu and C. Huang, "Systematic literature review of machine learning based software development effort estimation models," *Inf. Softw. Technol.,* vol. 54, no. 1, pp. 41-59, Jan. 2012, 41–59, 10.1016/j.infsof.2011.09.002

[23] S. K. Sehra, Y. S. Brar, N. Kaur and S. S. Sehra, "Research Patterns and Trends in Software Effort Estimation," *Inform. Softw. Technol.*, vol. 91, pp. 1-21, Nov. 2017, 10.1016/j.infsof.2017.06.002.

[24] D. Dvir, A. Ben-David, A. Sadeh and A. J. Shenhar, "Critical managerial factors affecting defense projects success: A comparison between neural network and regression analysis," *Eng. Appl. Artif. Intell.,* vol. 19, no. 5, pp. 535–543, Aug. 2006, 10.1016/j.engappai.2005.12.002.

[25] Y. R. Wang and G. E. Gibson, "A study of preproject planning and project success using ANNs and regression models," *Automat. Constr.*, vol. 19, no. 3, pp. 341-346, May 2010, 10.1016/j.autcon.2009.12.007.

[26] M. Paliwal and U. A. Kumar, "Neural networks and statistical techniques: A review of applications," *Expert Syst. Appl.*, vol. 36, no. 1, pp. 2-17, Jan. 2009, 10.1016/j.eswa.2007.10.005.

[27] S. Chulani, B. Boehm and B. Steece, "Bayesian analysis of empirical software engineering cost models," *IEEE Trans. Softw. Eng.*, vol. 25, no. 4, pp. 573-583, Jul./Aug. 1999, 10.1109/32.799958.

[28] Y. Li, V. Rakesh and C. K. Reddy, "Project success prediction in crowdfunding environments," in *Proc. 9th ACM Int. Conf. Web Search and Data Mining,* San Francisco, CA, USA, 2016, pp. 247-256.

[29] H. Park and S. Baek, "An empirical validation of a neural network model for software effort estimation," *Expert Syst. Appl.*, vol. 35, no. 3, pp. 929–937, Oct. 2008, 10.1016/j.eswa.2007.08.001.

[30] Idri, A. Zakrani and A. Zahi, "Design of Radial Basis Function Neural Networks for Software Effort Estimation," *IJCSI*, vol. 7, no. 4, pp. 11-17, Jul. 2010.

[31] González-Carrasco, R. Colomo-Palacios, J. L. López-Cuadrado and F. J. García- Peñalvo, "SEffEst: effort estimation in software projects using fuzzy logic and neural networks," *Int. J. Comput. Intell. Syst.*, vol. 5, no. 4, pp. 679-699, Aug. 2012, 10.1080/18756891.2012.718118.

[32] B. Nassif, D. Ho and L. F. Capretz, "Towards an Early Software Estimation Using Log-linear Regression and a Multilayer Perceptron Model," *J. Syst. Softw.*, vol. 86, no. 1, pp. 144-160, Jan. 2013, 10.1016/J.JSS.2012.07.050.

[33] P. Pospieszny, B. Czarnacka-Chrobot and A. Kobyliński, "An effective approach for software project effort and duration estimation with machine learning algorithms," *J. Syst. Softw.*, vol. 137, pp. 184-196, Mar. 2018, 10.1016/j.jss.2017.11.066.

[34] E. Papatheocharous and A. S. Andreou, "Software cost estimation using artificial neural networks with inputs selection," in *Proc. ICEIS2007*, Funchal, Madeira – Portugal, 2007, pp. 398–407.

[35] F. de Barcelos Tronto, J. D. S. da Silva and N. Sant'Anna, "An investigation of artificial neural networks based prediction systems in software project management", *J. Syst. Softw.,* vol. 81, no. 3, pp. 356–367, Mar. 2008, 10.1016/j.jss.2007.05.011.

[36] Attarzadeh and S. H. Ow, "Software development cost and time forecasting using a high performance artificial neural network model," in *Proc. ICICIS 2011*, Cairo, Egypt, 2011, pp. 18-26.

[37] L. A. Vidal, "Thinking project management in the age of complexity: particular implications on project risk management," Ph.D. dissertation, Bus. Admin., Ecole Centrale Paris, Paris, France, 2009.

[38] M. Kapsali, "Systems thinking in innovation project management: A match that works," *Int. J. Proj. Manag.*, vol. 29, no. 4, pp. 396-407, May 2011, 10.1016/j.ijproman.2011.01.003.

[39] J. Sheffield, S. Sankaran and T. Haslett, "Systems thinking: Taming complexity in project management," *On the Horizon*, vol. 20, no. 2, pp. 126-136, May 2012, 10.1108/10748121211235787.

[40] R. G. Taylor, "Systems thinking for project management: implications for practice and education," *Acta Structilia*, vol. 17, no. 1, pp. 79-103, Jan. 2010.

[41] H. Kerzner, *Project Management Best Practices: Achieving Global Excellence*. New York, NY, USA: John Wiley & Sons, Inc, 2018.

[42] G. Prabhakar, "What is project success: A literature review," *Int. J. Bus. Manag.*, vol. 9, no. 3, pp. 3-10, Jan. 2008, 10.5539/ijbm.v3n9p3.

[43] Baccarini, "The logical framework method for defining project success," *Proj. Manag. J.,* vol. 30, no. 4, pp. 25-32, Dec. 1999, 10.1177/875697289903000405.

[44] C.V. Bullen and J. F. Rockart, "A Primer on Critical Success Factors," in Center for Information Systems Research Working Paper No. 69, Cambridge, MA: Sloan School of Management, M.I.T., 1981.

[45] Y. P. Freund, "Critical success factors," *Planning Rev.*, vol. 16, no. 4, pp. 20-23, Apr. 1988, 10.1108/eb054225.

[46] C. Boynton and R. W. Zmud, "An assessment of critical success factors," *MIT Sloan Manag. Rev.,* vol. 25, no. 4, pp. 17-27, Jul. 1984.

[47] R. A. Khan and K. Spang, "Critical success factors for international projects," in *Proc. 6th IEEE IDAACS*, Prague, Czech Republic, 2011, pp. 879-883.

[48] M. H. Nasir and S. Sahibuddin, "Critical success factors for software projects: A comparative study," *Sci. Res. Essays*, vol. 6, no. 10, pp. 2174-2186, May 2011, 10.5897/SRE10.1171.

[49] P. Lech, "Causes and remedies for the dominant risk factors in enterprise system implementation projects: The consultant's perspective,". SpringerPlus, vol. 5, Art. No. 238, Feb. 2016, 10.1186/s40064-016-1862-9.

[50] L. Von Bertalanffy, "The history and status of general systems theory," *Acad. Manag. J.*, vol. 15, no. 4, pp. 407-426, Dec. 1972, 10.5465/255139.

[51] L. Von Bertalanffy, *General system theory*. NY, USA: Braziller, 1968, pp 131-133.

[52] D. Katz and R. L. Kahn, *The social psychology of organizations*. NY, USA: Wiley, 1978.

[53] C. Gresov and R. Drazin, "Equifinality: Functional Equivalence in Organization Design," *Acad. Manage. Rev.*, vol. 22, no. 2, pp. 403–428, Apr. 1997, 10.5465/amr.1997.9707154064.

[54] Child, "Organizational structure, environment and performance: The role of strategic choice," *Soc.*, vol. 6, no.1, pp.1-22, 1972.

[55] S. Saha, S. Das, A. Acharya, A. Kumar, S. Mukherjee, I. Pan and A. Gupta, "Identification of nonlinear systems from the knowledge around different operating conditions: A feed-forward multi-layer ANN based approach," in *Proc. 2nd IEEE PDGC 2012*, Solan, Himachal Pradesh, India, 2012, pp. 413-418.

[56] M. Last, "The uncertainty principle of cross-validation," in *Proc. IEEE GrC*, Atlanta, Georgia, USA, 2006 pp. 275-280.

[57] M. Bisi, N. K. Goyal, *"Software development efforts prediction using artificial neural network,"* *IET Softw.*, vol. 10, no. 3, pp. 63–71, Oct. 2016. DOI: 10.1049/iet-sen.2015.0061

[58] R. Poonam, J. Sonal, *"Enhanced Software Effort Estimation Using Multi Layered Feed Forward Artificial Neural Network Technique,"* *Procedia Comput. Sci.*, vol. 89, pp. 307-312, Dec. 2016. DOI: 10.1016/j.procs.2016.06.073

[59] Nassif, M. Azzeh, L. F. Capretz, D. Ho, *"Neural network models for software development effort estimation: a comparative study,"* *Neural Comput. Appl.*, vol. 27, no. 8, pp. 2369-2381, Nov. 2016. DOI: 10.1007/s00521-015-2127-1

[60] X. Li, Q. Jiang, M. K. Hsu, Q. Chen, *"Support or Risk? Software Project Risk Assessment Model Based on Rough Set Theory and Backpropagation Neural Network,"* *Sustainability,* vol. 11, no. 17, 4513. Nov. 2019. DOI:10.3390/su11174513

[61] F. Costantino, G. D. Gravio, F. Nonino, *"Project selection in project portfolio management: An artificial neural network model based on critical success factors,"* *Int. J. Proj. Manag.* vol. 33, no. 8, pp. 1744-1754. Nov. 2015. DOI: 10.1016/j.ijproman.2015.07.003

[62] Kaushik, D. K. Tayal, K. Yadav, A. Kaur, *"Integrating firefly algorithm in artificial neural network models for accurate software cost predictions,"* *J. Softw. Evol. Proc.*, vol. 28, pp. 665–688, Jun. 2016. DOI: 28. 10.1002/smr.1792.

[63] López-Martína, A. Abran, *"Neural networks for predicting the duration of new software projects,"* *J. Syst. Soft.* vol. 101. pp. 127 – 135. Mar. 2015. DOI: 10.1016/j.jss.2014.12.002

[64] A. M. G. Rodríguez, R. G. R. Morales, A. P. Vergara, Y. T. Casañola, J. P. F. Rodríguez and Y. M. Zamora, *"Evolutionary Artificial Neural Network for success forecast in Software Process Improvement,"* 2018 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Gudalajara, Mexico, 2018, pp. 1-6, DOI: 10.1109/LA-CCI.2018.8625244.

[65] D. Magaña Martínez, J. C. Fernández-Rodríguez, *"Artificial Intelligence applied to project success: a literature review,"* *Int. J. Artif. Intell. Interact. Multimed,* vol. 3, no. 5, pp. 77-82, Jan. 2015. DOI: 10.9781/ijimai.2015.3510.

[66] T. Chow and D. B. Cao, "A survey study of critical success factors in agile software projects," *J. Syst. Softw.*, vol. 81, no. 6, pp. 961-971, Jun. 2008, 10.1016/j.jss.2007.08.020.

[67] J. Fortune and D. White, "Framing of project critical success factors by a systems model," *Int. J. Proj. Manag.*, vol 24, no 1, pp. 53-65, Jan. 2006, 10.1016/j.ijproman.2005.07.004

[68] P. Yetton, A. Martin, R. Sharma and K. Johnston, "A model of information systems development project performance," *Infor. Syst. J.*, vol. 10, no. 4, pp. 263–289, Mar. 2003, 10.1046/j.1365-2575.2000.00088.x.

[69] S. C. Misra, V. Kumar and U. Kumar, "Identifying some important success factors in adopting agile software development practices," *J. Syst. Softw.*, vol. 82, no. 11, pp. 1869-1890, Nov. 2009,10.1016/j.jss.2009.05.052.

[70] L. Jun, W. Qiuzhen and M. Qingguo, "The effects of project uncertainty and risk management on is development project performance: A vendor perspective," *Int. J. Proj. Manag.*, vol. 29, no. 7, pp. 923-933, Oct. 2011, 10.1016/j.ijproman.2010.11.002.

[71] J. Sheffield and J. Lemétayer, "Factors associated with the software development agility of successful projects," *Int. J. Proj. Manag.*, vol. 31, no. 3, 459-472, Apr. 2013, 10.1016/j.ijproman.2012.09.011.

[72] J. A. Livermore, "Factors that significantly impact the implementation of an agile software development methodology," *JSW*, vol. 3, no. 4, pp. 31-36, Apr. 2008.

[73] J. Jiang and G. Klein, "Software development risks to project effectiveness," *J. Syst. Softw.*, vol. 52, no. 1, pp. 3-10, May 2000, 10.1016/S0164-1212(99)00128-4.

[74] L. Mohammad and B. Al-Shargabi, "Agile software methodologies: Employee, customer and organization factors," in *Proc. Int. Conf. Technol. Bus. Manage.*, Dubai, UAE, 2011, pp. 1201-1205.

[75] B. Boehm and R. Turner, "Using Risk to Balance Agile and Plan-Driven Methods," *Lett. IEEE Comput. Soc.*, vol. 36, no. 6, pp. 57-66, Jun. 2003, 10.1109/MC.2003.1204376.

[76] A. Sikandar, H. Li, K. Siffat Ullah, Z. Yang and L. Zhu, "Success Factors for Software Outsourcing Partnership Management: An Exploratory Study Using Systematic Literature Review," *IEEE Access*, vol. 5, pp. 23589-23612, Oct. 2017, 10.1109/ACCESS.2017.2764946.

[77] T. Clancy, "The Standish group report - CHAOS report 1995," The Standish Group International, Inc., Dennis, MA, 1995. [Online] Available: http://www.inf.ufpr.br/urban/2019-1_205_e_220/205e220_Ler_ver_para_complementar/StandishGroup__ChaosReport.pdf.

[78] J. Wan and R. Wang, "Empirical research on critical success factors of agile software process improvement," *J. Softw. Eng. Appl.*, vol. 3, no. 12, pp. 1131-1140, Dec. 2010, 10.4236/jsea.2010.312132.

[79] N. Fenton, W. Marsh, M. Neil, P. Cates, S. Forey and M. Tailor, "Making resource decisions for software projects," in *Proc. ICSE04 - Int. Conf. Softw. Eng.*, Edinburgh, Scotland, UK, 2004, pp. 397-406, 10.1109/ICSE.2004.1317462.

[80] X. Wang, C. Wu and L. Ma, "Software project schedule variance prediction using Bayesian Network," in *Proc. ICAMS2010*, Bucharest, Romania, 2010, pp. 26 - 30.

[81] D. Howell, C. Windahl and R. Seidel, "A project contingency framework based on uncertainty and its consequences," *Int. J. Proj. Manag.*, vol. 28, no. 3, pp. 256-264, Apr. 2010, 10.1016/j.ijproman.2009.06.002.

[82] Y. J. Jung, J. W. Wang and S. Wu, "Competitive strategy, TQM practice, and continuous improvement of international project management," *Int. J. Qual. Reliab. Manag.*, vol. 26, no. 2, pp. 164-183, Jan. 2008, 10.1108/02656710910928806.

[83] R. Imreh and M. Raisinghani, "Impact of agile software development on quality within information technology organisations," *J. Emerg. Trends Comput. Inf. Sci.*, vol. 2, no. 10, pp. 460-475, Oct. 2011.

[84] Z. Mansor, S. Yahya and N. H. Arshad, "Towards the development success determinants charter for agile development methodology," *Int. J. Inf. Technol. Eng.*, vol. 2, no. 1, pp. 1-7, Jan./Jun. 2011.

[85] S. Dyck and T. A. Majchrzak, "Identifying Common Characteristics in Fundamental, Integrated, and Agile Software Development Methodologies," in *2012 45th Hawaii Int. Conf. on Syst. Sciences*, Maui, HI, 2012, pp. 5299-5308, 10.1109/HICSS.2012.310.

[86] R. Kaufman, H. Oakley-Browne, R. Watkins and D. Leigh, *Strategic Planning for Success: Aligning People, Performance and Payoffs*. San Francisco, CA, USA: Pfeiffer, 2003, pp. 39-56.

[87] N. Cerpa, M. Bardeen, C. Astudillo, and J. M. Verner, "Evaluating different families of prediction methods for estimating software project outcomes," *J. Syst. Softw.*, vol. 112, pp. 48–64, Feb. 2016, 10.1016/j.jss.2015.10.011.

[88] M. Matejić and M. Šolaja. *"ICT in Serbia at a glance 2020,"* Vojvodina ICT Cluster, Serbia. May 9, 2020. [Online]. Available:

https://vojvodinaictcluster.org/wp-content/uploads/2020/05/ICT-in-Serbia-At-a-Glance-2020.pdf

[89] S. Dutta, B. Lanvin and S. Wunsch-Vincent. "Global Innovation Index 2020 – Who Will Finance Innovation?," Cornell University, INSEAD and the World Intellectual Property Organization (WIPO). ? Ithaca, Fontainebleau and Geneva. 13th edition, Sep. 2, 2020. [Online]. Available: https://www.wipo.int/edocs/pubdocs/en/wipo_pub_gii_2020.pdf

[90] U. Sekaran and R. Bougie, *Research-Methods-For-Business-A-Skill-Building-Approach*. Chichester, WS, UK. John Wiley & Sons, 2006, pp. 51-69.

[91] S. Cicmil, T. Williams, J. Thomas and D. Hodgson, "Rethinking project management: Researching the actuality of projects," *Int. J. Proj. Manag.*, vol. 24, no. 8, pp. 675–686, Nov. 2006, 10.1016/j.ijproman.2006.08.006.

[92] L. Gingnell, U. Franke, R. Lagerström, E. Ericsson, J. Lilliesköld, *"Quantifying Success Factors for IT Projects—An Expert-Based Bayesian Model,"* *Inf. Syst. Manag.,* vol. 31, pp. 21–36, Oct. 2014. DOI: 10.1080/10580530.2014.854033.

[93] N. Fenton, P. Hearty, M. Neil, L. Radliński. *"Software Project and Quality Modelling Using Bayesian Networks,"* in *Artificial Intelligence Applications for Improved Software Engineering Development: New Prospects,* 1st ed., New York, USA, IGI Global, 2009, ch 1, pp. 1-25.

[94] J. D. Procaccino, J. M. Verner, S. Overmyer and M. E. Darter, "Case study: Factors for early prediction of software development success," *Inform. Softw. Tech.*, vol. 44, no. 1, pp. 53-62, Jan. 2002, 10.1016/S0950-5849(01)00217-8.

[95] Wohlin and A. A. Andrews, "Evaluation of three methods to predict project success: A case study," in *Proc. 6th Int. Conf. PROFES 2005*, Oulu, Finland, 2005, pp. 385-398.

[96] N. Cerpa, M. Bardeen, B. A. Kitchenham and J. M. Verner, "Evaluating logistic regression models to estimate software project outcomes," *Inf. Softw. Technol.*, vol. 52, no. 9, pp. 934–944, Sept. 2010, 10.1016/j.infsof.2010.03.011.

[97] T. H. Davenport, R. Ronanki, *"Artificial intelligence for the real world,"* *Harv. Bus. Rev*, vol. 96 no. 1. pp. 108–116. Jan-Feb 2018.

[98] P. Mikalef, M. Boura, G. Lekakos, J. Krogstie, *"Big data analytics and firm performance: Findings from a mixed-method approach,"* *J. Bus. Res,* vol. 98, pp. 261–276. May 2019. DOI: 10.1016/j.jbusres.2019.01.044

[99] M. Schrage, D. "Kiron, Leading with next-generation key performance indicators," MIT Sloan Manag. Rev, vol 16. Jun 2018. DOI: 10.13140/RG.2.2.28573.05600

[100] F. Caputo, V. Cillo, E. Candelo, Y. Liu, *"Innovating through digital revolution: The role of soft skills and Big Data in increasing firm performance,"* *Manag. Decis,* vol. 57. no. 8, pp. 2032–2051. Dec 2019. DPI: 10.1108/MD-07-2018-0833.

[101] Keding, C. *"Understanding the interplay of artificial intelligence and strategic management: four decades of research in review,"* *Manag. Rev. Q.* Feb 2020. DOI: 10.1007/s11301-020-00181-x

[102] A. Miklosik, M. Kuchta, N. Evans, S. Zak, *"Towards the adoption of machine learning-based analytical tools in digital marketing,"* *IEEE Access.* vol. 7, pp. 85705–85718. Jul 2019. DOI: 10.1109/ACCESS.2019. 2924425.

[103] F. Borges, M. Laurindo, R. Spínola, C. Gonçalves, Mattos, *"The strategic use of artificial intelligence in the digital era: Systematic literature review and future research directions,"* *Int. J. Inform. Manage.* Sept. 2020. DOI: 10.1016/j.ijinfomgt.2020.102225

[104] C. Chen, A. Liaw and L. Breiman, "Using random forest to learn imbalanced data," University of California, Berkeley. Tech. rep. Department of Statistics, 2004. url: http://www.stat.berkeley.edu/users/chenchao/666.pdf.

[105] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Trans. Knowl. Data Eng.*, vol. 21, no. 9, pp. 1263-1284, Jun 2009, 10.1109/TKDE.2008.239.

[106] J. S. Reel, "Critical success factors in software projects." *IEEE Softw.*, vol. 16, no. 3, pp. 18-23, May/Jun 1999, 10.1109/52.765782

ZORICA M. MITROVIĆ received both her B.Sc in Organizational Sciences (2009) and her M.Sc in project management (2013) from the University of Belgrade, Faculty of Organizational Sciences, Serbia. She is currently pursuing her Ph.D. degree in information systems and quantitative management at the same University.

From 2010 to 2011, she was a Research Trainee with the Institute Mihailo Pupin, Science and Technology Policy Research Centre. Since 2011, she has been a Teaching Assistant at the Faculty of Organizational Sciences, University of Belgrade, in the Department of Management and Specialized Management Disciplines. Her research and teaching interests lie in the fields of project, performance and strategic management. She is an author and co-author of 11 scientific papers in domestic and international journals, as well as over 40 papers published from the proceedings of domestic and international conferences.

ALEKSANDAR M. RAKIĆEVIĆ received both his B.Sc. degree in information systems and his M.Sc. degree in electronic business and system control, from the University of Belgrade, Faculty of Organizational Sciences, Belgrade, Serbia. He is currently pursuing the Ph.D. degree in system control at the same University.

He is a Teaching Assistant at the Faculty of Organizational Sciences, University of Belgrade, at the Department for System Control. His research interests include systems, control and computational intelligence and their application in finance and management.

Mr. Rakićević is a member of IEEE societies for Computational Intelligence, Control Systems and Systems, Man, and Cybernetics and the European Society for Fuzzy Logic and Technology.

DEJAN Č. PETROVIĆ is a full professor at the Faculty of Organizational Sciences, University of Belgrade. He received the B.Sc. degree in industrial engineering from University of Belgrade Faculty of Organizational Sciences in 1992, M.Sc. degree in business development University of Belgrade Faculty of Organizational Sciences in 1997, and Ph.D. degree in organizational sciences University of Belgrade Faculty of Organizational Sciences in 2003.

He is also visiting professor at several universities in the country and abroad. His research interests include management, project management, strategic management, project appraisal and change management. He has been a consultant for a wide variety of public and private organizations, and he has published books and articles in refereed journals.

MARKO M. MIHIĆ received the B.Sc. degree in organizational sciences in 2001, the M.Sc. degree in business administration in 2005 and the Ph.D. degree in project management in 2008, all from University of Belgrade, Faculty of Organizational Sciences.

He is a Full Professor and Head of Department of Management and Specialized Management Disciplines at Faculty of Organizational Sciences and Visiting Professor at Faculty of Administration, University of Ljubljana, Slovenia. He is a Director of Centre for professional certification of managers at Faculty of Organizational Sciences and one of the founders of Centre for Microsoft Project established in cooperation with Microsoft Serbia. He has published 14 books and chapters and over 200 peer-reviewed papers. His research interest includes Project Management, Strategic Management, Cost-Benefit Analysis, Change Management and Leadership.

Prof. Mihić is Certified Lecturer in Strategic Management and Leadership at The Chartered Management Institute and member of following professional institutions: Academy of Management and Project Management Institute.

IEEE *Access*

Multidisciplinary : Rapid Review : Open Access Journal

JOVANA Đ. RAKIĆEVIĆ received the B.Sc. degree in information systems and technologies in 2011 and the M.S. degree in management of development in 2013, both from the University of Belgrade, Faculty of Organizational Sciences, Belgrade, Serbia. She is currently pursuing the Ph.D. degree in information systems and quantitative management at the same University. Since 2011 she has been a Teaching Assistant at the Faculty of Organizational Sciences, University of Belgrade, at the Department for Technology, Innovation and Development Management. After about 10 years of teaching in the field of technology and innovation management, 2 years ago Jovana started an intensive extracurricular work with undergraduate students on building their knowledge and skills in technology entrepreneurship. Her research and teaching interest is in the fields of technology and innovation management and technology entrepreneurship, technology commercialization and transfer, and entrepreneurial ecosystem support. She is an author and co-author of 5 chapters in international monographs, 10 scientific papers in domestic and international journals, and over 40 papers in proceedings of domestic and international conferences.

Mrs. Rakićević is a member of the Triple Helix Association and the European Society for Fuzzy Logic and Technology.

ELENA T. JELISIĆ received the B.S. degree in information systems and technologies in 2011 and the M.S. degree in information systems in 2013, both from the University of Belgrade. She is currently pursuing the Ph.D. degree in information systems at the same University. Since 2011 she has been engaged as a Teaching Assistant at the Faculty of Organizational Sciences, University of Belgrade, at the Department of information systems. Her research interests include software development and database design. Recently, her research is significantly focused on integration and interoperability of enterprise services and applications, specifically on investigating mechanisms that could ensure complete semantic interoperability.