e e ch   c e

Fuzzy Multicriteria Decision-Making Approach for Measuring the Possibility of Cloud Adoption for Software Testing

Sikandar Ali [1,2] **Niamat Ullah,** [3] **Muhammad Faisal Abrar,** [4] **Zhongguo Yang,** [5] **and Jiwei Huang** [1,2]

[1] *ep me of omp e c e ce   ech o o , h   e   of e o e m, e   1   , h*
e   e   o o of e o e m   , h   e   of e o e m, e   , e   1   , h
ep me of omp e c e ce   e of e , e 1   ,
ep me of omp e of e   ee , e of   ee   ech o o ,   ,
choo of fo m o c e ce   ech o o , o h h   e   of ech o o , e   , h

Correspondence should be addressed to Jiwei Huang; huangjw@cup.edu.cn

Received 20 October 2019; Revised 27 December 2019; Accepted 9 January 2020; Published 30 April 2020

Guest Editor: Rahman Ali

To reduce costs and improve organizational e ciency, the adoption of innovative services such as Cloud services is the current trend in today's highly competitive global business venture. e aim of the study is to guide the software development organization (SDO) for Cloud-based testing (CBT) adoption. To achieve the aim, this study rst explores the determinants and predictors of Cloud adoption for software testing. Grounded on the collected data, this study designs a technology acceptance model using fuzzy multicriteria decision-making (FMCDM) approach. For the stated model development, this study identi es a list of predictors (main criteria) and factors (subcriteria) using systematic literature review (SLR). In the results of SLR, this study identi es seventy subcriteria also known as in uential factors (IFs) from a sample of 136 papers. To provide a concise understanding of the facts, this study classi es the identi ed factors into ten predictors. To verify the SLR results and to rank the factors and predictors, an empirical survey was conducted with ninety- ve experts from twenty di erent countries. e application value in the industrial eld and academic achievement of the present study is the development of a general framework in-corporating fuzzy set theory for improving MCDM models. e model can be applied to predict organizational Cloud adoption possibility taking various IFs and predictors as assessment criteria. e developed model can be divided into two main parts, ranking and rating. To measure the success or failure contribution of the individual IFs towards successful CBT adoption, the ranking part of the model will be used, while for a complete organizational assessment in order to identify the weak area for possible improvements, the assessment part of the model will be used. Collectively, it can be used as a decision support system to gauge SDO readiness towards successful CBT.

1. Introduction

Over the past decades, software applications are becoming complex, dynamic, component based, and distributed, therefore making software testing a challenging task [1]. Industry reports show that software of 20,000 lines of code might take seven weeks of uninterrupted execution for a thorough regression test [2]. Many software development organizations (SDOs) consider test process automation to shorten development cycle and decrease testing costs [3]. Moreover, it is time-consuming and laborious to conduct some types of testing manually like regression testing, while through test automation, they can be performed e ciently [4]. Once tests have been automated, they can be run repeatedly and quickly. Test automation also proves to be an economical method for the regression testing of software with lengthy maintenance spell [4]. However, this type of testing demands huge investment from an SDO on various testing tools and related resources which may go out of use over a period of time because technology and application advance over time. Since SDO is supposed to retain pace with the industrial changes, SDO nds it hard and unfeasible to upgrade them every time [2].

Due to the essential value-added feature of Cloud Computing (CC), paradigms such as on large resource pool,

self-service on demand, measured service, broad network access, and rapid elasticity application development, deployment, and testing in the CC environments have been greater than before. This results in the reduction of delivery time, increased productivity, and capital and operational cost saving. Besides numerous benefits, it also postures several challenges for the quality of service (QoS) assurance including high-availability, multitenancy issues, elasticity, privacy, and security [5]. CC is regarded by many authors as a paradigm shift in computing, in lieu of an important change in the way computing services are offered, developed, attained, and paid for [6]. In practice, in this model, a Cloud service provider (CSP), serving the public, uses a pool of computing services via multitenancy. Potential clients can acquire and use these services over the Internet with automated user-friendly interfaces. Using the pay-as-you-go model, the service consumption is automatically metered [7].

At present, Cloud services are offered in at least three different models, i.e., Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS). These services are typically organized in one of the four infrastructures which are public, private, hybrid, or community Cloud settings. In the near past, numerous business organizations have shown a growing interest in the adoption of Cloud services to support various corporate functions [8]. CC is one of the five most persuasive technologies across the globe and is ranked the third widely invested significant IT service in 2013 [9]. According to the latest 2018 McKinsey report, over the next three years, organizations will make a fundamental shift from developing IT to utilizing IT [10]. Companies of all sizes are moving to off-premise Cloud services, among a significant move are noticed in large enterprises. The market share of CC is increasing fast, 30% small and medium businesses (SMBs) have already acquired five or more Cloud services while 60% have acquired a minimum of one Cloud service [11]. While cost reduction is often perceived to be the main factor in adopting CC services, the report shows that quality is a key predictor for Cloud adoption, while compliance and security remain key inhibitors for acceptance, specifically for large organizations [10].

As stated above there are three basic service models of CC, i.e., IaaS, PaaS, and SaaS. However, when we want to conduct Cloud testing, there emerges a new service model called Software Testing-as-a-Service (STaaS) or simply TaaS. A new term, known as TaaS or STaaS, has become a hot topic of discussion in different research groups as well as IT businesses and the Cloud industry [12]. The term of "TaaS" was originally coined and properly introduced by Tieto (http://www.tieto.com/) in 2009 in Denmark. Afterward, TaaS has been widely accepted by both industry and academic communities owing to its advantages in utility style service models, scalable testing environment, demanded testing services, and cost reduction [12]. This model has been assimilated into commercial products like Sogeti and Tieto. TaaS in a CC environment is deliberated as a new service and business model where a TaaS service provider creates a software testing environment and provides them as-a-testing service to accomplish testing tasks and activities for SDO

to test application software under-test (SUT) or a web-based system running over a Cloud infrastructure [12]. In connection to software testing, the terms "testing in the Cloud," "Cloud-based testing (CBT)," "Cloud testing," "STaaS," and "TaaS" are now widely used as synonyms.

To cope with the challenges of software testing; employing CC technology would be the best choice. While many organizations are adopting CC with careful assurance, testing seems to be one of the areas where they are willing to be more optimistic to move. Adoption of CC for software testing process is considered to be a safe decision because testing data does not contain any sensitive business information [2]. However, the adoption of CC for software testing is still in the initial stage [13]. The adoption rate of CC depends on the identification of novel factors. Before taking any important adoption decision, the significant factors for Cloud adoption should be ranked. Studies focusing on the adoption of CC for software testing are still lacking. Furthermore, conducting testing in the CC environment is neither cost-effective nor it is the best possible solution to all testing problems. Furthermore, not every type of software is the best candidate for testing in the Cloud [14].

SDO concerned about CBT must be conscious about the characteristics and types of software that are suitable to be tested in the Cloud environment, which enables them to better employ CBT [15]. Unit testing (particularly large regression suites), performance testing, and high-volume automated testing are the ideal choices to be migrated to the Cloud environment [14]. Unlike traditional web-based software testing, TaaS on Cloud infrastructure has several distinct requirements and unique features [12]. Some of the requirements from different perspectives are (1) the applications or SUT must be accessible online. The SUT might be a SaaS application or non-SaaS application. Additionally, this takes into account different levels of testing, for example regression or performance testing; (2) diverse deployment models of the Cloud such as private, public, hybrid, or community are used to host testing platforms and infrastructure; (3) testing of the Cloud itself [16].

Moreover, it is not straightforward to move testing to the Cloud; several quantitative and qualitative factors influence the decision to adopt CC. For example, all the traditional artifacts must be relocated to a new platform while still adhering to the original quality attributes and functionality. Questions like why tests are migrated to the Cloud, which techniques will be suitable, which test needs to be migrated, and when to migrate to the Cloud demand to be answered before proceeding with the CBT [17]. When transferring testing to the Cloud platform, the tools that are used in the course of the testing need to be switched to an innovative Cloud environment while still coinciding with the development environment. These artifacts consist of the test environment (such as testbeds and tools), test plans, testing method, test cases, test results and its documentation, and so on. Equal rewards should be reaped provided that high risks are posed while moving application testing to the CC platform [14].

1.1. Importance. Specifically, this study explores the determinants and predictors of Cloud adoption in the

context of software testing and develops a Cloud testing adoption assessment model (CTAAM) for decision-making towards CC adoption for software testing, specifically known as CBT. CTAAM was developed based on the input from experts' judgment using fuzzy multicriteria decision-making (FMCDM) approach. For the proposed model development, we performed a systematic literature review (SLR) for data collection by applying our customized search strings [18]. From results of the SLR study, we identify seventy influential factors (IFs) from a sample of 136 papers. To validate the SLR findings and to rank the IFs, an empirical survey was conducted in the software testing industry with ninety-five experts from 20 different countries. Based on the collected data, we performed a two-phase analysis based on FMCDM methods. We develop CTAAM model based on the two-phase analysis results. For evaluation of the CTAAM two case studies were conducted in SDO. To rank the IFs, we used our developed FMCDM based framework. For this purpose, we have developed an evaluation scale with 133 items based on previous researches.

1.1. Goal of the Study. This study contributes positively to the academic literature on the Cloud adoption in the context of software testing by identifying various criteria and is the foundation of Cloud adoption for future research in this field. Unlike other researchers, we formulate Cloud adoption as MCDM problem. Since several qualitative and quantitative factors influence decision towards Cloud adoption indicating that adoption is a kind of MCDM problem. Furthermore, unlike other researchers, we have combined the fuzzy set theory with MCDM to better handle vagueness, uncertainty, expert heterogeneity, human biasness, and subjectivity. In detail, in our framework model, the linguistic terms with parallel values in triangular fuzzy numbers (TFNs) format are used to translate the linguistic term. To handle expert heterogeneity, this study makes CTAAM capable of accumulating the decision of several experts by means of various predictors and factors as measurement criteria.

The model will do an organization assessment based on various IFs and predictors as evaluation criteria and can be applied to predict organizational Cloud adoption possibility taking various IFs and predictors as assessment criteria. The developed model can be divided into two main parts ranking and rating. To measure the success or failure contribution of the individual IFs towards successful CC adoption the ranking part of the model will be used while for a complete organizational assessment in order to identify the weak area for possible improvements the assessment part of the model will be used. Collectively, it can be used as a decision support system (DSS) to gauge SDO readiness towards successful CC adoption for CBT. The industrial contribution of the model is that it can be used as an assessment tool for SDO vendors and will indicate their weakness using an extended fuzzy version of the Motorola instrument specifically extended for this study. The models developed through FMCDM approach can handle uncertainty and vagueness in the expert judgments but they cannot identify the weak and strong

areas of an organization. Therefore, our framework model in this study uses Motorola assessment tool [19, 20] in the case study. Similar to our previous studies [21–23], this study also prolonged the Motorola evaluation tool to a fuzzy environment by suggesting the TFNs scale for its three dimensions. The existing Motorola guidelines [19, 20] were appropriate for a 5-point Likert scale only. We update the guidelines to 7-point Likert and introduce TFNs instead of even numbers (0, 2, 4, 6, and 10). We have designed seven point linguistic scales for assessing the importance weights of 133 items. Scale development is another contribution.

Theoretical contribution of the fuzzy study is the design and structure of a general framework to improve multi-criteria decision models. Other researchers and practitioners can use the proposed model structure and procedure as a template to develop FMCDM models and framework for decision-making, organizational capability improvement, and assessment for enterprises, organization, or companies in the other relevant fields. Moreover, the ranking part of the CTAAM can be adopted to design a standalone prioritization or weighting tool and to rank the impact of different risk or success factors while the assessment part of the CTAAM can be adopted to develop mini assessment tool. Collectively, it can be adopted as a DSS.

1.2. Paper Type. The rest of the paper is organized as follows. In the second section, a brief review of the studies conducted in the domain of CBT followed by quick motivation for using fuzzy set theory is presented. Section 3 describes the research methodologies for data collection and analysis. This is then followed by results obtained through ranking and rating in Section 4. Section 5 discusses the results while Section 6 discusses the study limitations. Finally, Section 7 concludes the paper.

2. Background and Motivation

Cloud testing is the process of software testing based on CC technology [5]. Cloud testing can be testing the Cloud itself or TaaS over Cloud. In this study, we focus on testing applications in the Cloud, i.e., TaaS. A group of expert architects and performance testers from the UK leading web load and performance testing company (http://www.CloudTesting.com/) initiated Cloud-based testing service for load and performance testing of website and Web applications [24].

The present literature around CBT is in the form of reports and industrial white papers. However, as the academic research on CBT gains popularity, scientific research has also been growing [22, 25–29]. Researchers examine specific types of testing [28], its architecture [30], applicability through case studies [31], experiments [31], and industrial survey [25]. Other related research, focusing on overviewing [32], surveying [25, 33], and summarizing the existing work [34].

The survey paper [35] poses some questions to the CC research community, which involve suitable solutions to software testing. Fernandes and Gemmer [36] have defined

the CC paradigm and its implications for IT organizations. Bertolino et al. [37] identify various references and buzzwords. Bai et al. [24] suggest new developments in Cloud testing tools like SaaS testing, cross-Cloud testing, real-time results processing and dynamic adaptive testing, dependability testing, service-level-agreement conformance testing, and security and reliability testing. Nachiyappan and Justus [38] give an overview of Cloud testing tools. In a comparative study, they discuss various challenges to CC testing tools. Titinen [39] in their master thesis identify critical factors for software testing tools selection. They also classify testing tools into test management, execution, comparison, framework, and measurement tools. Murthy and Suma [40] conduct a study on CC testing tools, and Devasena et al. [41] propose a load testing tool for Cloud.

However, this research area is still immature [26]. For the time being, a vast majority of the preceding researches addresses the testing of the core architecture of the Cloud. Furthermore, the majority of the preceding work focuses on Cloud service evaluation. Conversely, customization of those services regarding software testing has been largely ignored in the preceding literature. What's more, few studies focus on Cloud adoption [42–48] but they are not conducted form software testing, the only exception are [22, 25]. A recent survey on testing in the Cloud [49] identified lack of a Cloud testing adoption model that guides decision-maker, when and how to move testing to the Cloud, based on various decision factors as a research gap. Therefore, our study will be an attempt to bridge some of the most important research gaps by developing a research model that is adaptable and scalable for SDO.

1 *o* *o* *o* *e*   The decisions on whether to adopt Cloud as a testing tool are essential since incorrect decisions might consequence in a loss in terms of resources and efforts. A shortage of knowledge and information about a certain technology makes it uncertain and vague. Thus, unavoidably, most of the critical decisions are made under nondeterministic conditions while any carelessness may result in the dire consequences. Models and frameworks have been recommended to support the decision-makers for CC adoption, incorporating various aspects such as demand behaviors, business objectives, and quality of services, migration cost, and technical challenges. For instance, an MCDM model for ranking and selecting Cloud services includes six factors, namely, accountability, costs, assurance, agility, performance, and security [50]. Although there are numerous models for decision-making [42–48, 51–55] but they are not conducted form software testing, the only exceptions are [14, 22, 25]. Furthermore, most of them [42–48, 51–53] are not based on the fuzzy set theory. Furthermore, they are proposed for either service selection [46] or Cloud provider selection [46]. Interestingly, none of them is proposed for software testing or TaaS. The only exception is found in [14], in which authors propose a decision framework called SMART-T. SMART-T originated from Service-Oriented Architecture (SOA) Migration, Adoption, and Reuse Technique (SMART) [56] where shows testing. SMART-T

contains three parts: business drivers, technical factors, and operational results. SMART-T is developed based on business drivers and technical factors. Additionally, it is not developed based on the fuzzy set theory. Moreover, we did not find any study from the perspective of CC adoption, handling vagueness, or uncertainty in experts' decision-making.

Our study will contribute to the existing literature on Cloud adoption and software testing. Unlike the preceding researchers, we formulate Cloud adoption as the MCDM problem. Since several qualitative and quantitative factors influence in decision towards Cloud adoption indicating that adoption is a kind of MCDM problem. Compared with the preceding studies, our study has the following significance. First, very few studies on CBT have been conducted. *Second*, no compressive study on Cloud adoption for software can be found in the relevant literature. Therefore, this paper is the initial effort to provide new valuable discernments on the adoption of CC technology for software testing, which is considered to be an emerging research perspective in automated testing. Third, besides the importance of CBT, very limited evidence in the relevant literature can be found that discuss aspects of CBT adoption through systematic theoretical evaluation and models. In fact, the application of operational research models and theories, especially technology adoption theories and the MCDM approaches for Cloud adoption in the context of software testing remains unexplored. Consequently, proposing a Cloud testing adoption assessment model (CTAAM) and linking various influential factors to its core constructs of organizational adoption intention is deliberated as a valuable tool, which is the fundamental innovation of this study. Fourth, previous studies have only utilized MCDM [45–47, 51, 57] approach in ranking the influential factors and its dimension and predicting the organizations' intention to adopt Cloud technologies but they are not conducted for software testing, the only exception is [25]. Besides, we did not find any study, handling vagueness or uncertainty in experts' decision-making as well as causal relations among the identified factors simultaneously from the perspective of CC adoption. To handle expert heterogeneity, this study develops a model capable of aggregating the decision of multiple experts by using multiple predictors and factors as measurement criteria. In our proposed framework, multiple experts can participate in both the ranking survey and rating assessment case study. Additionally, unlike the preceding studies, this study provides the experts with a self-effacing linguistic scale as presented in column 1 of Tables 1 and 2.

pp c o of *e* *eo* *o* *c* *e* *ec* *o*   In real-world decision problems experts typically have to decide with imprecise, vague, or incomplete data [58]. Vagueness exists in the natural language terms, such as good or best, better than, smaller than, considerable, important, significant, not implemented, partially implemented, fully implemented, achieving, achieved, outstanding, qualified, marginally qualified, etc. [59, 60]. Vagueness or fuzziness in data may be due to poor boundaries of scale

TABLE 1: Linguistic scale for ranking of items.

Linguistic terms	Corresponding scale
Not contributing at all	(0.0, 0.0, 0.1)
Contributing sometime	(0.0, 0.1, 0.3)
Slightly contributing	(0.1, 0.3, 0.5)
Not sure	(0.3, 0.5, 0.7)
Moderately contributing	(0.5, 0.7, 0.9)
Strongly contributing	(0.7, 0.9, 1.0)
Extremely contributing	(0.9, 1.0, 1.0)

TABLE 2: Linguistic scale for rating of items.

Linguistic term	Corresponding scale
Very poor implementation	(0.0, 0.0, 0.1)
Poor implementation	(0.0, 0.1, 0.3)
Weak implementation	(0.1, 0.3, 0.5)
Fair implementation	(0.3, 0.5, 0.7)
Marginally implemented	(0.5, 0.7, 0.9)
Fully implemented	(0.7, 0.9, 1.0)
Outstanding	(0.9, 1.0, 1.0)

[61]. All real numbers greater than one is a familiar example of a class of objects where boundaries are not clear or poorly defined [58]. To cope with the fuzziness usually articulated in decision data rising from the qualitative independent judgment of the experts due to the uncertainty, incomplete or vague data, the scale based on crisp numbers might be inadequate to model the real-world MCDM problems [62].

Furthermore, the influential factors are not likewise significant as the influential factors may be quite different regarding their contribution to success [59]. Thus, it is more forthright and realistic for experts to rank the IFs for example "factor F extremely contributing towards CC adoption" rather than to guess "factor "F" 70% contributing towards CC adoption". Converting heterogeneous information to a single form might result in a loss of important information [63].

In practice, decision-making usually requires subjective data provided by the expert [64]. It is obvious that different experts have different preference structures, knowledge levels, and complex decision capabilities based on their experiences [65, 66]; therefore, a good decision-making model must tolerate vagueness or ambiguity in expert judgment [60, 65]. Numerous researchers have recommended incorporating fuzzy set theory [54, 55, 58–67] dispensing subjectivity in decision-making by human experts. Fuzzy statistics in FMCDM methods are represented by fuzzy numbers via fuzzy sets. In fuzzy sets, the degree of association is allocated to objects in their universe of discourse. An object in this approach can fit some sophisticated classes of objects with only a partial association [67].

Fuzzy set theory

Zadeh [67] is the first researcher to incorporate "fuzzy sets theory" and proposed the MCDM approach for subjective decision-making. It has been widely adopted to model uncertainties in human decision-making. It also meritoriously resolves uncertainties in collecting data for MCDM. Several researchers have identified that the subjective vagueness of expert judgment can be dispensed by integrating the fuzzy set theory [59, 60, 65, 66] with MCDM. In a fuzzy set, linguistic terms are used to denote fuzzy variables that are then translated to the corresponding numerical values using the predefined linguistic scale [65, 66].

Membership function

A fuzzy set has a membership function which assigns a score of association to objects within its universe of information between 0 and 1.

Figure 1 shows the coordinates of the three vertices lower, m central, and, upper of $A()$ in a fuzzy set A. Let is a universal set having items { } then, a fuzzy set $\widetilde{A}$ in the universe of information defined by its membership function is as follows:

$$_A(\ ): U \to [0,1], \tag{1}$$

which assigns to each { } a score of association to $\widetilde{A}$ in the interval [0, 1]. Here $_A(\ ) =$ score of membership of in $\widetilde{A}$, assign values in range of 0 to 1, i.e.,

$$\overline{A}(\ ) \in [0,1]. \tag{2}$$

A fuzzy set can also be represented by a continuous membership function $\overline{A}(\ )$.

$$\overline{A}(\ ) = \begin{cases} 0, & \text{if } \quad 1, \\ \dfrac{-1}{m-l}, & \text{if } l \quad m, \\ \dfrac{u-}{u-m}, & \text{if } m \quad u, \\ 0, & \text{if } \quad u, \end{cases}. \tag{3}$$

Linguistic scale

The conventional scaling approaches face difficulties in labeling the criteria that are overly complicated or hard to define [67]. To counter the consequences of fuzziness the linguistic scale is used. Linguistic terms with parallel TFNs offer practical means for dispensing incidents of subjectivity. The appropriateness of substitutes versus criteria and the importance weights of criteria are assessed using a linguistic scale with corresponding fuzzy numbers [59, 60]. Several researchers [21–23, 59, 60, 65, 66] recommend the linguistic scale for describing such situations.

Inspired by the above-reported studies, in this paper, the linguistic terms with parallel TFNs scores are used to reproduce the information. Precisely, we have designed seven Likert linguistic scales for ranking IFs via 133 items. Seven Likert linguistic scales as presented in Table 1 were used to rank the significance of the identified factors via its items found via SLR in phase 2.

Likewise, seven linguistic scales as presented in Table 2 grounded on Motorola evaluation tool are designed. The case study companies were asked to rate the level of
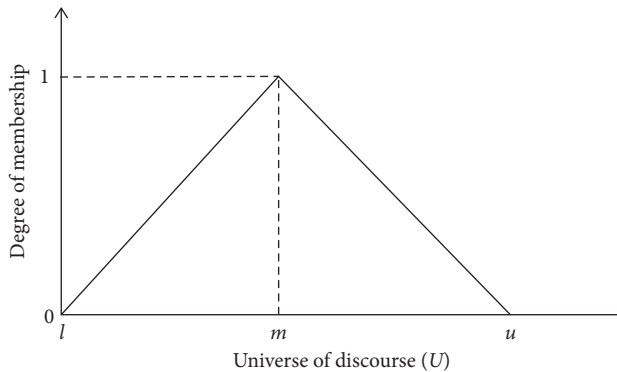
FIGURE 1: Coordinates of Triangular fuzzy numbers.

implementation of items across the three dimensions of Motorola evaluation tool [21], i.e., results, deployment, and approach as presented in Table 3.

3. Research Methodology

In terms of the proposed research model, we have developed it by including empirical research methodology based on a mixture of quantitative and qualitative research techniques for data collection and a two-phase FMCDM approach for analysis as deliberated in the following subsections.

Phase#1: to identify critical influential factors (CIFs), for data collection at the early stage, SLR was used.

Phase#2: to rank the identified CIFs using fuzzy FMCDM and at the same time to validate the SLRs findings, a questionnaire survey was carried out.

Phase#3: based on the inputs from the questionnaire survey and SLR the CTAAM was developed in phase 3.

Phase#4: to assess the CTAAM through the FMCDM approach and at the same time to evaluate its effectiveness in real-world environment case study was executed.

1 Systematic Literature Review is study conducts SLR [68] to find the critical factors of the adoption of CC for software testing. Before conducting the SLR, we design a review plan known as a protocol. We have published the SLR protocol with initial results in a conference paper [18].

1 1 Review Protocol It reduces researchers' bias and enhances review accuracy and repeatability [68]. Particularly, it outlines rationale for the study, search strategies, study questions, literature inclusion/exclusion and quality assessment checklists, and data extraction and synthesis plans [68]. Figure 2 illustrates the main phases of the SLR process.

Protocol of the search In terms of the searching phase, the scope of the study is acknowledged as follows:

Population Primary studies on "Cloud-based testing," "testing in the Cloud," "Cloud testing," "testing-as-a-service," and "TaaS."

Intervention Published primary studies reporting success factors for Cloud-based software testing.

Outcome of relevance List of factors related to the CBT adoption.

Experimental Any forms of empirical study.

The below example shows the aforementioned parts:

[What factors/challenges]   "INTERVENTION"

In [Cloud software testing]   "POPULATION"

Which impacts upon

[Adoption of Cloud]   "OUTCOMES OF RELEVANCE"

KEYWORDS_ABSTRACT_TITLE

(Software OR Application OR program) AND (Test OR Validate OR verification OR maintenance OR accepting) AND (Cloud OR Cloud computing OR TaaS OR Testing as a service) AND (factors OR motivators OR drivers OR elements OR parameters OR characteristics OR adopt decision OR criteria OR "lesson learned")

Search plan: We used an automatic search method for publication selection because manual search is not easy for digital libraries where the numbers of available documents are over several thousands. The final search string is given below.

Literature Selection: In this step, the investigators have to list sources to be searched, inclusion, and exclusion and quality criteria.

Selection of sources: Following the guidelines for SLR [68], we selected seven search engines/databases related to the study subject to run an exhaustive search [68]. Databases are chosen based on the characteristics, such as coverage of indexed articles (journals, conferences, or books), availability of the study, and importance to the software engineering area. Following are the seven databases, we selected.

(1) ACM_digital library http://www.acm.org/
(2) Scholar_Google https://scholar.google.com
(3) IEEE_Xplore-http://ieeexplore.ieee.org/
(4) Cite_Seer-http://citeseer.ist.psu.edu/
(5) Science_Direct-http://www.sciencedirect.com/
(6) Springer_Link-http://link.springer.com/
(7) Wiley_online library-https://onlinelibrary.wiley.com/

Inclusion and Exclusion Criterion: An inclusion and exclusion criterion is defined to verify the importance and usefulness of each study. The relevant criteria are listed in Table 4.

Quality Criteria: The quality checks are listed in Table 5. The checklist will be scored based on three possible answers; yes, partial, and no. If any of the checks are not relevant to any articles, then it will be excluded from the evaluation of that particular study only. Those studies that will not provide the basic information about their research methods will score less than 50% in quality assessment and will be excluded.

Data extraction: Primary reviewers will independently review all papers. In the case of needs for guidance, the secondary reviewers will be approached.

Execution of the review: e review was conducted from June 2017 to September 2018. To support the protocol trial, we used the EndNote X7 tool for the control group of primary studies and paper selection and storage. Moreover, Microsoft Access was used to manage the data extraction and quality assessment.

mp c e To rank the identi ed CIFs using FMCDM and at the same time to validate the SLR's ndings, a questionnaire survey was carried out in the software development industry. Questionnaire survey serves as an appropriate method of gathering tacit quantitative and qualitative facts [69]. e process of a questionnaire survey can be divided into two phases, designing and sampling. In the design phase, a set of questions are designed to be answered by the sample. In the sampling phase, the in-
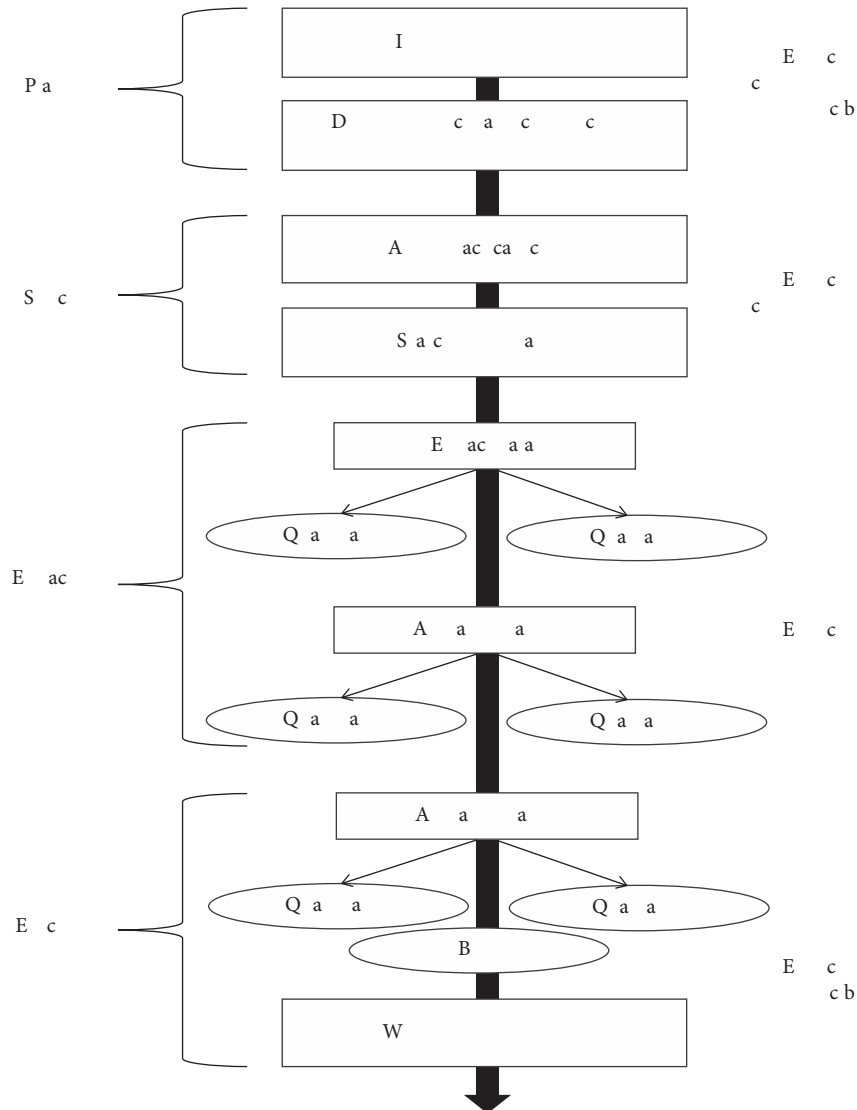
Figure 2: Various stages in the SLR process.

questions and required scripts and web pages came out as results. It is a commercially available online application. It also helps in collecting responses from participants, because it facilitates resending of the link through e-mail in case of no response. e questionnaire design was piloted through six members of our laboratory and necessary changes were made accordingly.

he Survey inquiry is deliberated as a suitable method of gathering tacit qualitative and quantitative data [69]. e questions of the questionnaire are of two types: open-ended, also called subjective, and close-ended, called objective. e subjective questions allow a variety of answers from the respondents, while for objective, only one choice can be chosen from the available choices. is method of data gathering assists in reducing the threat of bias relating to the investigator's prejudices. It encourages the respondent to give her/his view regarding a speci c question [69, 70]. Before a questionnaire, each participant

was sent a questionnaire invitation letter. is letter outlined the main theme of the questionnaire survey, the expected duration, and measures that could be taken to ensure privacy and con dentiality. We concentrate on obtaining the linguistic weight for the in uential factors (items) because not all the factors are equally important for the adoption of CC in connection to software testing.

pp o ch For analysis purposes, we use the FMCDM framework presented in the next section.

1 me o fo e he o of o op o fo of e e is paper aims to increase empirical knowledge of CC adoption for software testing. Recently, the use of MCDM approaches has been increasing in information science and software engineering. In this paper, we successfully adopt FMCDM for the evaluation and assessment of CBT in the software industry. For

TABLE 4: Inclusion criteria/exclusion criteria.

S. no	Inclusion criteria
1	Studies that describe factors for cloud-based software testing (CBT)
2	Studies that describe the evaluation of tools, methods, processes, and techniques for CBT.
3	Studies that describe advice/strength/weakness of the CBT framework
4	Studies that describe lessons learned in adopting CBT tools/services
5	Studies that describe techniques/process for migrating software testing to cloud platform
6	Studies that describe criteria for what and when to shift software testing to cloud platform
7	Studies that describe solutions to the challenges in CBT testing domain
8	Studies that evaluate CBT service provider practical capabilities for automatic software testing
	Exclusion criteria
1	Studies written in a language other than English
2	Studies that do not mention practices/solutions/challenges/tools/methods or techniques in automatic software testing in the Cloud
3	Studies conducted in a domain other than software engineering
4	Studies with less than four pages of contents
5	Studies published in more than one sources
6	Studies which get a low score on the quality checklist

TABLE 5: Quality criteria.

S. no	Quality criteria
1	Is it clear how the factors that influence the adoption are identified?
2	Are the findings supported by extensive data?
3	Are the objectives clearly stated?
4	Are the experimental or observational units adequately described?
5	Is the paper based on based on some standard theory or model?
6	Are negative findings presented?
7	Are the implications clearly stated?
8	Are all the data proposed in the data extraction form possible to extract?
9	If there is a control, what are these variables that may affect study results?
10	Is the study context well defined?
11	Are the testing strategies formally described?
12	Is there a link between interpretation along with conclusion and data?

the stated research model development, we have incorporated a five-phase empirical research methodology using a mix of qualitative and quantitative research techniques for data collection and MCDM approach for model development and evaluation. An FMCDM framework model as shown in Figure 3 was developed to rank and rate the identified alternative criteria by assembling multiple experts in decision-making as deliberated below.

Phase 1: Selection of the relevant factors. To find the influential factors for the adoption of CC for CBT, the review protocol was executed to select papers through five distinct stages. In the first stage, after the selection of the databases, the search string was applied to the selected sources and we got 1,650 research articles for further process. The number of returned papers in each database is shown in Table 6. In the second stage, we read the titles and keywords and apply inclusion criteria. In the third stage, we read the abstracts and conclusions and apply exclusion criteria. At stage-3, we also excluded 29 studies that were repeated across different databases. In the fourth phase, we read the full text of the remaining articles and applied quality criteria along with inclusion/exclusion, which results in the studies to be described in the review. After the data extraction

and execution stage, we extract a list of quotes from the sample of 136 papers, where each quote described a list of influential factors. The primary reviewers in conjunction with the secondary reviewers reviewed these quotes to classify these quotes into different categories. Initially, 82 categories were identified; these were reviewed and merged into 70 different factors.

Phase 2: Selection of the relevant criteria of the main criteria. This study presents the participants with the ease to understand linguistic terms, parameterized by TFNs to precise the individual disagreement or agreement about the importance of identified predictors (main criteria) and factors (subcriteria). We are concerned about finding the ranks because all the criteria (factors and predictors) are not equally important. To rank the predictors, we divided it into small factors called subcriteria.

After data collection through SLR, the identified factors (also called subcriteria in this section) are framed into twelve major variables, ten independent (also called main criteria in this section), one mediator, and one dependent variable. Out of seventy influential factors, eleven were identified to have negative effect on the Cloud adoption, these are term as barriers. The fifty-nine factors are distributed into ten main
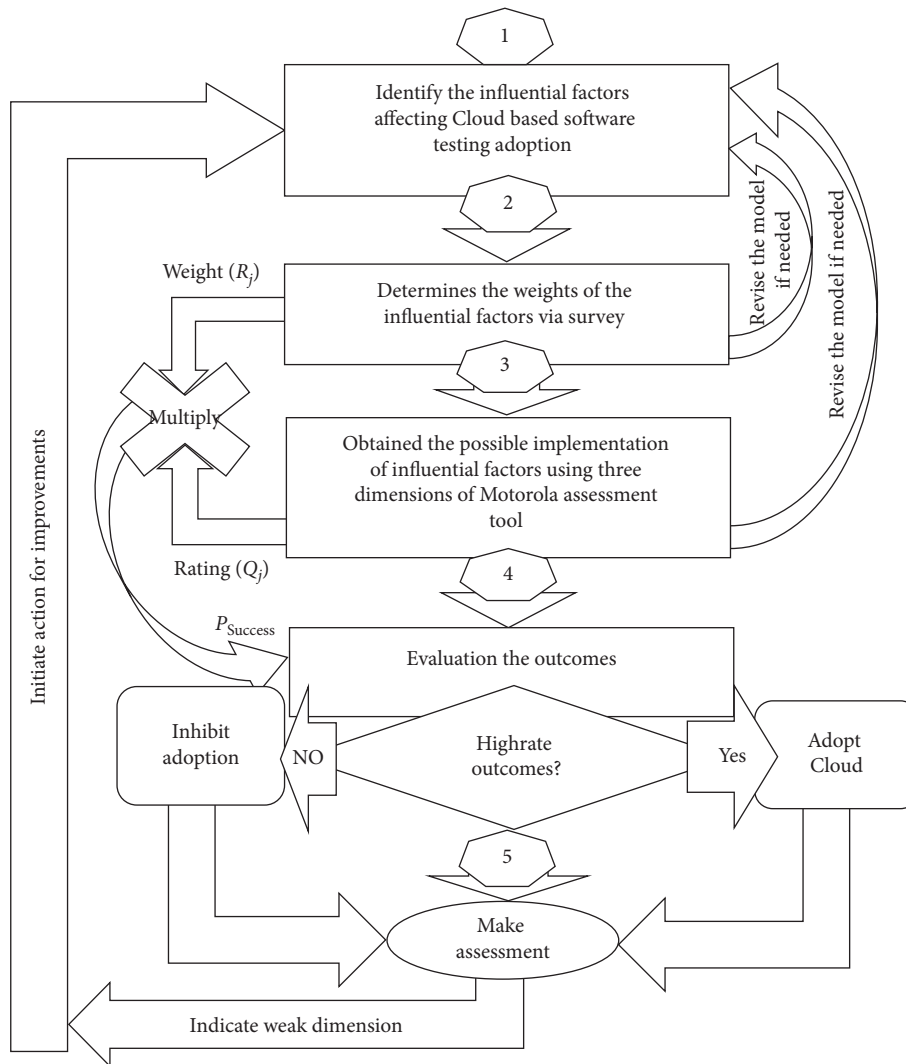
Figure 3: FMCDM model for measuring the possibility of successful Cloud adoption for software testing.

Table 6: Papers included/excluded during different phases.

Search venue	After phase 1	After phase 2	After phase 3	After phase 4
Science direct	177	83	80	14
ACM	123	55	50	17
IEEE xplore	842	250	229	76
Springer link	77	32	21	13
CiteSeer	63	07	06	03
Google scholar	400	58	21	12
Wiley	13	03	03	01
Total	1650	533	464	136

variables that are supposed to positively affect the adoption decision while the 11 barriers collectively form one variable that is supposed to negatively affect adoption decision (Figure 4).

Scale development process. The questions and the items to measure factors for this study have been borrowed from the preceding research on CC adoption. The scales used to collect the survey data about various predictors and factors used for this study are illustrated in Appendix-A in Supplementary materials. The nest means for assuring content validity is to select items from the earlier confirmed scale [71]. Therefore, in this study, we try to borrow items from preceding research to show generalized insights. To further improve the questionnaire design and contents, the questionnaire was sent to two experts in this field and two experts in academia conducting research in the relevant field. Their comments were incorporated accordingly.

I   c   ab   a
F   Q   S
S   c   a a ab   a   ab
S   c   a   c   c
U   ab   a   a
P   c   c   b
P   c   c   c
P   c   c   c

FIGURE 4: A conceptual model of Cloud adoption: predictors and factors.

Based on the outcomes of SLR, we have identified ten main criteria and seventy subcriteria as illustrated in Figure 4. These criteria were measured through one hundred and thirty-three items in an expert-based questionnaire survey. The items are presented in Appendix-A in Supplementary materials.

the experiment the the computation of the sample. To rank the influential factors and validate the findings of SLR, an empirical questionnaire survey was performed. Survey investigation is considered an appropriate method for collecting tacit quantitative and qualitative data [69]. SLR results were used as input to the

questionnaire. The importance weights of main and sub-criteria are assessed through a linguistic scale denoted by TFNs [59, 60]. In this study, we incorporate the fuzzy scores both in the weighting survey (see Table 1) and implementation case study (Table 2). Specifically, based on TFNs as shown in the second column of Tables 1 and 2, we translated the expert opinions for processing through the FMCDM framework.

The specific math of the selected process for computing the significant weights of items is described in the following steps:

Step 1: Generate a decision matrix $\widetilde{A}$ for the significance weights of items ($I_b, b = 1, 2, 3, \ldots, n$). The participants of the survey ($E^a, a = 1, 2, 3, \ldots, m$) are then asked to give their subjective judgment about the significant weight of each item ($\widetilde{}^a_b$ where represents item, represents item's number, and represents an expert's number) through the linguistic variables as given in Table 1, e.g.,

$$\widetilde{A} = \begin{matrix} & E^1 \; E^2 \; E^3 \cdots \; E^m \\ I_1 \\ I_2 \\ I_3 \\ \\ I_n \end{matrix} \begin{bmatrix} \widetilde{}^1_1 & \widetilde{}^2_1 & \widetilde{}^3_1 & \cdots & \widetilde{}^m_1 \\ \widetilde{}^1_2 & \widetilde{}^2_2 & \widetilde{}^3_2 & \cdots & \widetilde{}^m_2 \\ \widetilde{}^1_3 & \widetilde{}^2_3 & \widetilde{}^3_3 & \cdots & \widetilde{}^m_3 \\ & & & & \\ \widetilde{}^1_n & \widetilde{}^2_n & \widetilde{}^3_n & \cdots & \widetilde{}^m_n \end{bmatrix}, \quad a = 1, 2, \ldots, m; b = 1, 2, \ldots, n. \tag{4}$$

where m represents the total number of experts and n represents the total number of items and, $\widetilde{}^a_b = (l\widetilde{}^a_b, m\widetilde{}^a_b, u\widetilde{}^a_b)$ represents the fuzzy weight of the bth items given by ath respondents.

Step 2: As the independent assessment of each expert, vary according to his or her, perception, knowledge, role, and experience of the subject matter. Thus, we calculate the average score to cumulative the fuzzy importance of each item by m experts.

$$\widetilde{}_b = \frac{1}{m} \left[\sum_{a=1}^{m} \widetilde{}^a_b \right], \tag{5}$$

where $\widetilde{}_b = (l_b, m_b, u_b)$ represents the cumulative fuzzy importance weight of the th item

Step 3. The cumulative fuzzy importance $\widetilde{}_b$ is used to calculate the best nonfuzzy performance (BNF) value BNP_{W_b}. BNP_{W_b} can be obtained using

$$\text{BNP}_{W_b} = \frac{[(uw_b - lw_b) + (mw_b - lw_b)]}{3} + lw_b, \tag{6}$$

where BNP_{W_b} denotes the best nonfuzzy performance (BNP) score for the TFN $\widetilde{}_b$ while W_b is the significant weight of the bth item in the crisp numeral format. The local ranks will be obtained through W_b.

Step 4. After the defuzzification of TFNs in step 4, crisp numbers are calculated and normalized through (7) while the items global ranks R_b will be obtained through equation (7):

$$R_b = \frac{W_b}{\sum_{b=1}^{n} W_b}, \tag{7}$$

where R_b denotes the normalized importance of the bth item such that $\sum_{b=1}^{n} R_b = 1$.

Step 5. Since each item measures different aspects of the subcriteria, we incorporate the average score method to cumulative the best fuzzy importance W_f of each subcriteria (factors).

$$W_f = \frac{\sum_{b=1}^{k} W_b}{k}, \tag{8}$$

where W_b represents the BNP weight of each item in the particular factor (subcriteria), k represents the total number of items measuring that factors while $\sum_{b=1}^{k} W_b$ is the aggregate BNP weight of all the items in that factor.

Computing the significant weight of the predictor

Step 6. Using W_b (BNP weight of items), we can obtain W_p, BNP weight of predictor using

$$W_p = \sum_{b=1}^{h} W_b, \tag{9}$$

where W_b represents the BNP weight of each item in the particular predictor, while h represents the total number of items in that predictor. We also obtain and normalize the crisp number for each predictor and each factor within predictor using equations (10) and (11):

$$R_f = \frac{W_f}{\sum_{f=1}^{h} W_f}, \tag{10}$$

where R_f represents the normalized significance weight of the bth factor such that $\sum_{f=1}^{h} R_f = 1$.

$$R_p = \frac{W_f}{\sum_{b=1}^{n} W_b}, \tag{11}$$

where $\sum_{b=1}^{n} W_b = W_t$ sum of the weight of all items and R_p represents the normalized significance weight of the bth predictor such that $\sum_{b=1}^{n} R_b = 1$.

The results of the case study. To assess the CTAAM through MCDM approach and at the same time evaluate its effectiveness in real-world

environment, two case studies were executed. Case study can provide adequate evaluation in real-world industrial environments [72]. Similar to the questionnaire survey, here we also translate the expert's judgment to TFNs format but unlike them, we use the extended version of the Motorola evaluation tool [21] to evaluate the SDO readiness towards Cloud adoption for CBT. For the assessment of influential factors and to find the implementation of various dimensions in the assessment case study, we also incorporate the fuzzy set theory. Seven linguistic scales as illustrated in Table 2 based on the dimension of Motorola evaluation tool design for case companies to rate factors across the three dimensions of Motorola evaluation tool [73] such as approach, deployment, and results as presented in Table 3.

he e   e of he of   e e
o fo   o   op o   e procedures for calculating the degree of implementation of the items in particular organizations are described in the following steps:

(i) Step 1: Generate three matrices $\widetilde{B}_A$, $\widetilde{B}_D$, and $\widetilde{B}_R$ for the rate of implementation of the items ($I_b, b = 1, 2, 3, \ldots, n$). A, , and   denotes the three dimensions of the Motorola evaluation tool as presented in Table 3.   e respondents of the survey ($E^a, a = 1, 2, 3, \ldots, m$) are asked to give their expert judgment about the degree of implementation of the items in their respective organizations following the guidelines of Motorola evaluation tool as illustrated in Table 3, by using the linguistic scale as presented in Table 2, e.g.,

$$
\widetilde{B}_A =
\begin{array}{c}
\ \ E^1 \quad E^2 \quad E^3 \quad \cdots \quad E^m \\
\begin{array}{c}
I_1 \\ I_2 \\ I_3 \\ \\ I_n
\end{array}
\left[
\begin{array}{ccccc}
\widetilde{1}_1 & \widetilde{2}_1 & \widetilde{3}_1 & \cdots & \widetilde{m}_1 \\
\widetilde{1}_2 & \widetilde{2}_2 & \widetilde{3}_2 & \cdots & \widetilde{m}_2 \\
\widetilde{1}_3 & \widetilde{2}_3 & \widetilde{3}_3 & \cdots & \widetilde{m}_3 \\
 & & & & \\
\widetilde{1}_n & \widetilde{2}_n & \widetilde{3}_n & \cdots & \widetilde{m}_n
\end{array}
\right]
\end{array}, \quad a = 1, 2, \ldots, m; \, b = 1, 2, \ldots, n,
$$

$$E^1 \quad E^2 \quad E^3 \quad \cdots \quad E^m$$

Table 7: An example of the predictor (main criteria) ranking using factor (subcriteria).

Expert	M2S1	M2S2	M2S3	M2S4	M2S5	M2S6 (1)	M2S6 (2)	M2S7 (1)	M2S7 (2)	M2S8
E1	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
E2	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)
E3	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
E4	(0.1, 0.3, 0.5)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E5	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.9, 1.0, 1.0)
E6	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E7	(0.1, 0.3, 0.5)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)
E8	(0.0, 0.0, 0.1)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
E9	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)
E10	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E11	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.9, 1.0, 1.0)	(0.3, 0.5, 0.7)	(0.9, 1.0, 1.0)
E12	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E13	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E14	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)
E15	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E16	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
E17	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
E18	(0.7, 0.9, 1.0)	(0.1, 0.3, 0.5)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E19	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E20	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.3, 0.5, 0.7)	(0.9, 1.0, 1.0)
E21	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E22	(0.1, 0.3, 0.5)	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E23	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)
E24	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E25	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E26	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)
E27	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E28	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
E29	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E30	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)
E31	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)
E32	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)
E33	(0.5, 0.7, 0.9)	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)
E34	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)
E35	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
E36	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)
E37	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)
E38	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)
E39	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
E40	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E41	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)
E42	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)

TABLE 7: Continued.

An example of factors ranking (subcriteria)

Expert	M2S1	M2S2	M2S3	M2S4	M2S5	M2S6 (1)	M2S6 (2)	M2S7 (1)	M2S7 (2)	M2S8
E43	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)
E44	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)
E45	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.9, 1.0, 1.0)
E46	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)
E47	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)
E48	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.1, 0.3, 0.5)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)
E49	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)
E50	(0.1, 0.3, 0.5)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)
E51	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.9, 1.0, 1.0)	(0.1, 0.3, 0.5)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)
E52	(0.5, 0.7, 0.9)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)	(0.9, 1.0, 1.0)
E53	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)	(0.7, 0.9, 1.0)	(0.9, 1.0, 1.0)
wj	(0.41, 0.60, 0.79)	(0.57, 0.76, 0.92)	(0.48, 0.68, 0.86)	(0.54, 0.73, 0.88)	(0.50, 0.70, 0.86)	(0.53, 0.71, 0.86)	(0.62, 0.81, 0.95)	(0.65, 0.82, 0.95)	(0.50, 0.69, 0.85)	(0.65, 0.80, 0.89)

the respective organizations, then it is easy to estimate the possibility of success P_{sucess} based on items using equation (17) as well as based on factors by equation (18)

$$P_{\text{sucess}} = R_b \times Q_b, \tag{17}$$

$$P_{\text{sucess}} = R_f \times Q_f. \tag{18}$$

From the possibility of success, we can easily compute the possibility of failure.

h e : e me   o o o   e me   oo . In the result of the evaluation, in case of a low outcome (less than 70%), the model will pinpoint weak in uential factors and predictors for further improvements while in case of high rate outcomes it will direct adoption and will signpost towards areas for enhancements.

e f he e e fo   e h ceme e BNP weights of items are used to compute the BNP weights of factors and predictors. If I_f represents the implementation of factor and I_p represents the implementation of predictor, then it can be found using (19) and (20), respectively:

$$I_f = \frac{Q_f}{k}, \tag{19}$$

where   represents the total number of items in that factor $Q_f = \sum_{b=1}^{k} Q_b$ is the aggregate BNP implementation of all items implements that factor.

$$I_p = \frac{Q_p}{h}, \tag{20}$$

where h represents the total number of items in that predictor and $Q_p = \sum_{b=1}^{h} Q_b$ is the aggregate BNP implementation of all the items that measure that predictor.

4. Results and Analysis

is section demonstrates the outcomes of our study.   is study proposes a technology acceptance model for CBT, based on the factors identi ed through the SLR and questionnaire survey, using the codi ed knowledge of experts and an intelligence technique for decision-making regarding CBT.   e outcomes presented here are obtained based on the analysis of collected data in the rst phase, i.e., ranking and rating of these data through the industrial survey and case study using FMCDM.

1 e   e   he e h o h mp c e is section presents the results obtained by employing FMCDM techniques to analyze the data collected through SLR. We have identi ed 133 items to measure the ten predictors called the main criteria using 70 subfactors via FMCDM.

1 1   e h of he c o   c e One hundred thirty-three items as shown in Appendix-A in Supplementary materials, were used in our scale to rank

predictors and factors through a questionnaire survey. Ninety- ve experts participate in the weighting process through a questionnaire survey.   e weight calculation processes are as follows:

(1) ese experts are asked to give their subjective judgment about the items using the linguistic scale presented in Table 1.

(2) e linguistic evaluations are then transcribed into the corresponding TFN as shown in Table 7 while taking predictor M2: "perceived performance expectancy" as an example.

(3) Because the perceptions of each expert are di erent due to their role, industrial experience, quali cation, etc., equation (5) is used to get the synthesized aggregate TFN ($\tilde{}_b$) as listed in Supplementary Table 2 column 4.

(4) en, defuzzi cation of the TFN is carried out to obtain BNP in a crisp format using equation (6).   e outcomes are shown in Supplementary Table 2 (BNP_WJ).   e BNP value is used in the further calculation for global ranking R_b and local ranking as shown in Supplementary Table 2 column 7.

(5) e crisp number obtained in step 4 is normalized and the normalized importance R_b of items are obtained by using equation (7) which are further used to nd an overall rank for each item.   e outcomes are presented in Supplementary Table 2 column 8 (second last column). R_b is used to obtain a global rank of each item as shown in Supplementary Table 2 column 9 (last column). As some factors and barriers are measured using more than one item, the individual item scores of a factor in BNF format are then converted to an average score in the same BNF format using equation (8) as shown in Supplementary Table 2 (last row). Referring to equation (8), the BNP weight of the factor (subcriteria) W_f is an average of the BNP weight of all the items in that respective factor.

Using W_f, we obtained the local rank of the factors (subcriteria) within predictors (main criteria) as shown in Supplementary Table 2 second last column (W_f).

1   e h of he e c o   e Referring to equations (8) and (9), BNP weight of the factor (subcriteria), W_f is the average of the BNP weight of all the items in that respective factor while the BNP weight of the predictor and W_p is the sum of the BNP weights of all the items in that predictor. We rst calculate W_f by using equation (8), then we nd the importance (rank) of factor within predictor (local weight) using (W_f).   e overall rank R_f of a factor was obtained by dividing W_f by sum of the weight of all factors, i.e., $\sum_{f=1}^{n} W_f$ refers to equation (10), which are 50.2752 in our study as given in Supplementary Table 2 (last row). For ranking predictors, we used the weight of the normalized importance of each predictor as calculated by equation (11). Ranks of the predictors are presented in

Table 7, where R_p is the normalized weight of the predictor and $W_t = \sum_{b=1}^{n} W_b$ shows the total weight, W_t is the sum of the BNP weight W_b of all the items and is accessible from Supplementary Table 2 (last row).

Case study of the Case Company

To obtain the actual level of implementation of the influential factors using different items in the SDO organizations, a case study is carried out. The case study is a suitable method for providing enough evidence in the real-world industrial environment.

To be more certain and confident in our assessment, two distinct case studies were carried out at two distinct SDO companies. Companies are chosen because they provided particularly detailed descriptions of different SDO activities. If an item or factor has a strong implementation in the company, then the likelihood of success in the adoption of Cloud for future projects increases. The possible implementation of factor (subcriteria) concerning to each predictor (main-criteria) is calculated as follows:

Step 1: The participants in the case study were requested to provide their independent views about the extent of implementation of each item in their respective organizations from the three dimensions of Motorola by choosing linguistic terms shown in Supplementary Table 3 incorporating the Motorola guidelines given in Table 3.

Step 2: The linguistic terms are then transcribed into corresponding TFN. An example of M2 (PPE): "perceived performance expectancy" are shown in Table 8.

Step 3: Three-dimensional scores in TFN format are then converted to an average score in the same TFN format using equation (13) as shown in Table 8.

Step 4: To aggregate the subjective judgments of the participants on the assessment of factors (because the perception of each expert is different due to their role, experiences, and education level, etc.). Equation (14) is used to get the synthesized TFN as listed in Table 8 (last row).

Step 5: Then defuzzification of the TFN is carried out to obtain BNP in the crisp format using equation (15) as shown in Supplementary Table 3.

Assessment of the Success of Cloud adoption

Once we have an importance weight R_b and possible implementation Q_j of items, then it is easy to calculate the possibility of successful adoption using (19). The possibility of successful adoption for Company-A is shown in Table 9 (last column) while that of Company-B is shown in Table 10.

The overall success value is equal to the sum of the success value of all items. Success value 0.5 indicates a 50% chance of both success and failure. Once we get success value, then the possibility of failure can be calculated using the common probability method.

The possibility of success of Cloud adoption based on the score of implementation for company-A is given in Table 9. It is clear from Table 9 that company-A has not implemented any of the predictors (main-criteria). Furthermore, $_{Success}$ is less than 70%. Our model did not suggest company-A to adopt Cloud at this time because there is a 50% chance of both success and failure. To increase the chances of adoption, Company-A must focus on the perceived business concerns which are not implemented. Similarly, based on the results presented in Table 10 Company-B, has implemented seven predictors; however, $_{Success}$ is still less than 70%. Based on the implementation score, Company-B is relatively more secure if moving to the Cloud environment. For complete implementation, Company-B must focus on the weak factors under perceived business concerns (M10), which will further increase trust and will stimulate Company-B to adopt the Cloud.

5. Summary and Discussion

The objective of this study is to prioritize the predictors as well as to rank the overall subcriteria. Firstly, the ten main criteria are ranked as CC adoption "Feasibility planning and risk analysis, 0.11"; "Perceived degree of Cloud resource utilization, 0.1168"; "Perceived effort expectancy, 0.1141"; "Organization competency and capacity, 0.1128"; "Perceived degree of trust, 0.111"; "Perceived business concerns, 0.1020"; "Organizational dynamics and business drivers, 0.0999"; "External stimuli, 0.0924"; "Perceived performance expectancy, 0.0745"; and "Perceived economic benefits of Cloud adoption, 0.057."

According to the rank of predictors, we find that feasibility planning and risk analysis, perceived degree of Cloud resource utilization, and perceived effort expectancy are the top three predictors of Cloud adoption in the context of software testing. It means, currently testing in the Cloud does not apply to all types of software. Furthermore, most of the testing techniques cannot be applied in the Cloud environment. Additionally, currently testing in the Cloud is not fully automated and testers are left with the laborious tasks of writing scripts and configuring the test environment. Utilization of Cloud resources is also limited due to the viability and suitability of different types of software to be tested and types of tests performed in the Cloud environment. Furthermore, human testers are still needed to conduct testing. This might be the reason that "perceived economic" benefit of Cloud adoption has low weight and ranks low. Besides, perceived performance expectancy is the second least significant predictor in our study. This might be the reason that the adoption of CC in the domain of software testing is very slow compared with that in other fields. The following studies support the results:

(i) The first step in moving to the Cloud is the deep understanding of CC resources and how they relate to their needs. It is also imperative that SDO realizes and analyzes the CC value-added features and goes on only if there is a growing need for this approach [56].

TABLE 8: Corresponding TFNs of evaluation in case organization taking "Perceive performance expectancy" is an example.

M2 (PPE)		Perceived performance expectancy									
		Factors/items									
Expert	Dimension	M2S1	M2S2	M2S3	M2S4	M2S5	M2S6 (1)	M2S6 (2)	M2S7 (1)	M2S7 (2)	M2S8
E1	A	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
	D	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
	R	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.1, 0.3, 0.5)	(0.0, 0.1, 0.3)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)
E2	A	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)
	D	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)
	R	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.1, 0.3, 0.5)	(0.1, 0.3, 0.5)
E3	A	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)
	D	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)
	R	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
E4	A	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)
	D	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
	R	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.0, 0.0, 0.1)
E5	A	(0.0, 0.1, 0.3)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
	D	(0.0, 0.0, 0.1)	(0.0, 0.1, 0.3)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)
	R	(0.0, 0.0, 0.1)	(0.0, 0.1, 0.3)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.0, 0.0, 0.1)
E6	A	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)
	D	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.0, 0.0, 0.1)
	R	(0.0, 0.0, 0.1)	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.3, 0.5, 0.7)	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.1, 0.3, 0.5)	(0.1, 0.3, 0.5)	(0.0, 0.0, 0.1)
E7	A	(0.3, 0.5, 0.7)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.7, 0.9, 1.0)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)
	D	(0.0, 0.0, 0.1)	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.7, 0.9, 1.0)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)
	R	(0.0, 0.0, 0.1)	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.1, 0.3, 0.5)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.3, 0.5, 0.7)	(0.5, 0.7, 0.9)	(0.5, 0.7, 0.9)	(0.1, 0.3, 0.5)
$\bar{q}_{ib} = (l\bar{q}_{ib}, m\bar{q}_{ib}, u\bar{q}_{j})$		.1, .34, .2	.4, .6, 1	.49, .69, 1	.31, 1, 1	.3, .6	.3, .6	.41, .61, 1	.41, .61, 1	.31, 1, 1	.2, .44, .63

(ii) To find an answer to the question "when to migrate testing to the CC platform?" one must check the characteristics of SUT, perceived support of the testing methods in the CC environment, preferences, nature, and dependency of test execution [74]. If the majority of these subfactors cannot operate in the Cloud environment, then Cloud adoption is considered feasible; otherwise, a decision to stop the migration can be made [74].

(iii) Before going to adopt the Cloud testing service, you have to look at ways through which testing is conducted and testing services the service providers offer, in order to plan best to maximum utilize Cloud platform for their testing [25].

(iv) TaaS can be exclusively embraced by the industry if it supports more types of testing such as agile development via continuous regression testing [56].

(v) Perceived ease of use contributes significantly towards a willingness to adopt CC services [45]. The outcomes show that convenience and ease of use are the key motivators for Cloud services adoption [45].

(vi) Automation could encourage the adoption of a service. On the contrary, if a service requires tremendous human effort, this might negatively affect adoption [75].

(vii) Despite the great importance of Cloud-based testing, it is extremely complex and at present lacks proper tools to support it [76].

(viii) Extra efforts are required to deploy application and setup test environment in the Cloud [77].

(ix) For simple test cases, the use of CC might increase the task cost and system complexity [78].

(x) Automation encourages the adoption of the CBT technique. On the contrary, if a tool or service requires huge efforts, this might limit its adoption [75].

(xi) To improve quality and reduce the cost of test automation is the most important means [79].

(xii) According to a topical survey [80], presently 28 % of test cases are automated whereas the executives demand to raise this amount shortly. Employing test automation mainstream in the software industry, more and more companies will adopt CBT.

(xiii) The authors of [56] suggest that future work on TaaS needs to focus on entirely automatic test cases. They examine the currently available tools and services and find that only BlazeMeter can offer fully automated performance and load testing [56].

The ranks of the subcriteria in connection to "CC adoption" for software testing are given in Supplementary Table 2. Among the 70 subcriteria, top ten criteria are support for multidirectional testing, 0.018547; top management support, 0.018471; interface usability, 0.017637;

standardization; 0.016031; reduced employee cost, 0.017257; pricing and deployment models, 0.016830; employee support for CBT, 0.016723; resources availability, 0.016637; perceived IT risks, 0.013693; and service automation, 0.016500.

Ten least significant factors in decreasing order of weights from 61 to 70 are regulation requirements, 0.012336; improve developer-tester collaboration, 0.011930; risk analysis, 0.011784; competitor's pressure, 0.011492; security, 0.011027; incompatibility, 0.008024; performance uncertainty and outage, 0.007983; legal concerns, 0.005898; insufficient QoS guarantee via SLA, 0.004413; and concerns over confidentiality, 0.004020.

Priyadarshinee et al. [71] rank the factors through fuzzy AHP. They identify organizational risks, sharing and collaboration, confidentiality and integrity, and security and privacy as top factors. According to them, security and privacy, sharing and collaboration, confidentiality, and integrity are the top four factors that have a major effect on the adoption of CC. Competitive pressure is ranked in the middle by us (and it is ranked 20th out of 51 by them), but unlike them, we have relatively low weight for security and privacy. The reason might be that testing data is not sensitive; therefore, organizations are not worried about the security of data.

(i) Migrating the testing to the CC environment is anticipated to be a safe decision because the testing will not contain any sensitive business data [2].

(ii) Migration will have a nominal impact on corporate work as usual deeds [26].

(iii) For organizations having no concerns for security, software testing can be one of the suitable areas that can be moved to the Cloud platform [49].

However, the privacy of the personal information and test results still concerns to SDO. Assurance of privacy, disaster recovery, and back up inspire the confidence of the client in Cloud adoption for software testing.

(i) Since testing does not contain sensitive information, the security concerns are related to the privacy of resources and confidentiality of test results [81].

(ii) Before adopting Cloud for software testing, one should decide whether the Cloud is safe for their sensitive information [35].

(iii) An acceptable testing service essentially addresses concerns related to multitenancy as well as to the safety of uploaded material and confidentiality of test results [82].

(iv) All tenants' data in Cloud are stored in a shared infrastructure using a shared single instance, hence making it difficult to ensure data confidentiality [83].

To assess our proposed framework, we have executed two case studies in the SDO planning to adopt Cloud for software testing shortly. At the end of the case study, a focus group session was conducted among the participants to

TABLE 9: Evaluation of predictor in case organization (Company-A).

Name of predictor	p	# items	p	Success
M01: perceived economic benefits of Cloud adoption	4.7540	9	0.5282	
M02: perceived performance expectancy	5.4460	10	0.5446	
M03: perceived effort expectancy	8.1333	14	0.5810	
M04: perceived degree of Cloud resource utilization	8.6905	14	0.6208	
M05: organizational dynamics and business drivers	7.4556	12	0.6213	50.3122 approx. 50.3%
M06: perceived degree of trust	8.2571	16	0.5161	
M07: external stimuli (pressure + support)	5.0429	12	0.4202	
M08: organizational competency and readiness	8.3413	13	0.6416	
M09: feasibility planning and risk analysis	9.8943	16	0.6184	
M10: perceived business concerns	8.0921	18	0.4496	

TABLE 10: Evaluation of predictor in case organization (Company-B).

Name of predictor	p	# items	p	Success
M01: perceived economic benefits of Cloud adoption	6.174	09	0.6860	
M02: perceived performance expectancy	6.996	10	0.6996	
M03: perceived effort expectancy	10.031	14	0.7165	
M04: perceived degree of Cloud resource utilization	10.325	14	0.7375	
M05: organizational dynamics and business drivers	9.4556	12	0.7880	61.26407 approx. 61.3%
M06: perceived degree of trust	8.2571	16	0.5161	
M07: external stimuli (pressure + support)	7.0429	12	0.5869	
M08: organizational competency and readiness	9.1930	13	0.7072	
M09: feasibility planning and risk analysis	11.843	16	0.7402	
M10: perceived business concerns	10.921	18	0.6067	

obtain feedback and the model was revised accordingly. Case study method proves the most powerful instrument for validation in empirical software engineering [72].

In this study, the Motorola assessment instrument and the TFN scale are used in a fuzzy environment. Our contribution is the introduction of TFN scores and the updated guidelines for a 7-point Likert scale. The existing Motorola guidelines are suitable for a 5-point Likert scale only; we have added two more rows, one at the start and the other at the end to make it convenient for measurement on a 7-point Likert scale. The same instrument is also used for the evaluation of SOVRM [84], SPIIMM [354], and SOPM [20].

This tool has been used in CMMI to evaluate the organizations' present position comparable to the CMM and to recognize weak areas that require improvements and management attention [73].

Evaluation results regarding factors using the Motorola assessment instrument are presented in Table 3. According to Motorola assessments, factors having an average evaluation or implementation score greater than 0.7 will be considered implemented while scores less than 0.7 will be marked as weakness. For instance, in predictor perceived economic benefits (M1), two factors such as market place establishment (MIS6) and access to global markets (M1S7) are not implemented properly. The reason might be that Company-A, which does not want these two goals to be achieved through CC adoption.

(i) The biggest challenge is to ensure global availability all the time [56].

(ii) Currently, there is no comprehensive model to test the mobile CC applications. Furthermore, they state that the new Cloud testing methods are restricted to the applications with special features and can only support limited types of tests [85].

All factors in perceived performance expectancy (M2) have scores less than 0.7 except better service availability and reliability (M2S3). It means Company-A is optimistic only about Cloud reliability and availability. It can be observed from Table 3 that "improves developer-tester collaboration" (M2S1) gets a very low implementation score, i.e., lower than 35%. It means regarding Company-A, adoption of CC will not improve developer and tester communication to a high level. Additionally, "better usability and easy deployment" gets a score (56%) which is found very high in the other Cloud adoption study such as [45, 86]. The reason might be that as compared with other Cloud services, testing is not fully automated. The following literature proves the claim:

(i) CC adoption is neither feasible and risk-free all the time nor an automatic or easy process [13].

(ii) The use of CC might increase the task cost and system complexity in case of simple tests [78].

(iii) According to a well-established French testing company called Sogeti, merely 28% of the test cases are presently automated, whereas the executives demand to raise this amount soon. Utilizing test automation in the software industry more and more companies will adopt CBT [80].

(iv) Future work on TaaS needs to focus on entirely automatic test cases. They have examined the currently available tools and services and found that

only BlazeMeter can offer fully automated performance and load testing [56].

In connection to the perceived degree of Cloud resource utilization (M4), we got a relatively low implementation of factors "varieties of service utilization" (M4S3), "support for Global software development" (M4S5), and "support for ubiquity and mobile testing" (M4S6).

(i) Not all types of testing are suitable to be performed in the Cloud. For instance, high-volume automated testing, unit testing (particularly large regression suites), and performance testing are ideal choices to be migrated to the Cloud environment [14].

(ii) TaaS can be exusively embraced by the industry if it supports more types of testing such as agile development via continuous regression testing [56].

(iii) The biggest challenge is to ensure global availability all the time [56].

(iv) Currently, there is no comprehensive model to test the Cloud mobile applications. Furthermore, they state that the new Cloud testing methods are restricted to applications with special features and can only be used in some types of tests [85].

6. Conclusion

This study proposes that software testing should be moved to the Cloud. Additionally, this study explores the determinants and predictors of Cloud adoption for software testing. For this purpose, this study explores the technical aspect of software testing in the Cloud through the FMCDM approach and further develops a technology acceptance model. An expert system based on the codified knowledge of an expert and artificial intelligence techniques such as MCDM is applied to help organizations decide Cloud adoption for software testing. The proposed model includes various predictors and determinants that will guide software development organizations (SDO) towards Cloud adoption for software testing. The model is developed using a five-phase analytic approach. Data are collected via SLR and questionnaire-based survey while analysis is made through fuzzy multicriteria decision-making (FMCDM) approach.

The results of this study suggest that Cloud service providers need to provide support to SDO who are presently working in the CBT domain, develop a cooperative environment for the CC adoption, and eliminate any ambiguity regarding this technology. Besides, the CC providers should clarify the advantages of Cloud services over in-house services. Furthermore, they must focus on concerns such as locality and privacy faced by the SDO and resolve those problems by forming an unrelenting relationship with the clients. The Cloud providers have to assure the SDO that their testing data are safe and accessible upon request around the clock. They should increase their reliability to get confidence, faith, and trust of the SDO about their services. A trial chance may be provided to the SDO to try the Cloud

services out before its actual implementation. The provider government has to play a significant role in building trust regarding government intrusion among the SDO. The Client needs to implement CBT and it is very off-beamed to think that CBT is meant for larger companies only.

7. Threats to Validity

The empirical survey was based on the results of SLR; this two-phase framework ensures content validity. Construct validity is concerned with measurement scale whether the measurement scales represent the attributes being measured. The attributes of this research study were taken from a considerable amount of previous research [25, 42–48] and experiencing a systematic literature review [20, 87, 88]. The respondents of the survey confirm the relevance of the attributes selected. Further, the inner reliability of survey responses was assessed using Cronbach's alpha coefficient, which is 0.89 (>0.70), which demonstrates the reliability of data and scale. To internal validity, the SLR findings were used as input for the design of the questionnaire.

The empirical study part of this research engaged participants mainly from the Asian countries only. However, to lessen population prejudice, contributors from other countries such as North America were also invited to include diverse perspectives. Fifty experts voluntarily participated in this exploratory study and there were no previous bonds between the participants and researchers. Contributors were informed that their participation is entirely voluntary and they can withdraw at any time during any stage of the survey if they want. However, to ensure external validity and to diminish any possible bias, the ninety-five contributors were chosen from 20 different countries. Besides, most of the participants had worked in a range of small, medium, and large multinational organizations. Although we cannot claim that all the contributors from these 20 countries would agree with us, we believe that they provide a descriptive sample. In an empirical survey-based research, it is hard to obtain a fully representative sample and to deal with them in an entirely objective fashion [89]. To overcome these limitations, only those participants were included who are involved in test automation. The claim/relevant expertise of the participants were verified by inculcating some open-ended questions in the questionnaire, which were difficult to answer by an ordinary tester or manager etc.

This situation might create difficulties when contributors' judgments may be inaccurate or factors supposed to have a significant interrelation for adoption may not be significant at all. However, similar to other opinion-based empirical research studies [25, 43–48], we have full confidence that the findings of this research are based on the data that have been collected from the relevant participants who have been involved and have vastly diversified experience in SDO.

Data Availability

The data used to support the study are provided in the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

Supplementary Materials

All the necessary data are provided in Supplementary Tables 1 to 3. (*Supplementary Materials*)

References

[1] M. Madan, M. Dave, and A. Tandon, "Challenges in testing of cloud based application," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, no. 1, pp. 28–31, 2016.

[2] M. S. N. Murthy and V. Suma, "Software testing and its scope in CLOUD: a detailed survey," in *Proceedings of the 1st International Conference on Computational Intelligence and Informatics*, pp. 269–273, Bengaluru, India, February 2017.

[3] O. Taipale, J. Kasurinen, K. Karhu, and K. Smolander, "Trade-off between automated and manual software testing," *International Journal of System Assurance Engineering and Management*, vol. 2, no. 2, pp. 114–125, 2011.

[4] V. Garousi, M. V. Mäntylä, A. Betin-Can, and S. Mirshokraie, "When and what to automate in software testing? A multi-vocal literature review," *Information and Software Technology*, vol. 76, pp. 92–117, 2016.

[5] G. N. Iyer, J. Pasimuthu, and R. Loganathan, "PCTF: an integrated, extensible cloud test framework for testing cloud platforms and applications," in *Proceedings of the 13th International Conference on Quality Software*, pp. 135–138, Dallas, TX, USA, May 2013.

[6] S. Marston, Z. Li, S. Bandyopadhyay, and A. Ghalsasi, "Cloud computing-the business perspective," in *Proceedings of the 44th Hawaii International Conference on System Science*, pp. 1–11, Kauai, HI, USA, January 2011.

[7] M. Armbrust, I. Stoica, M. Zaharia et al., "A view of cloud computing," *Communications of the ACM*, vol. 53, no. 4, pp. 50–58, 2010.

[8] J. Luftman, H. S. Zadeh, B. Derksen, M. Santana, E. H. Rigoni, and Z. Huang, "Key information technology and management issues 2011-2012: an international study," *Journal of Information Technology*, vol. 27, no. 3, pp. 198–212, 2012.

[9] L. A. Kappelman, E. R. Mclean, J. N. Luftman, and V. Johnson, "Key issues of IT organizations and their leadership: the 2013 SIM IT trends study," *MIS Quarterly Executive*, vol. 12, no. 4, p. 7, 2013.

[10] A. Elumalai, I. Starikova, and S. Tandon, *IT as a service: from build to consume*, McKinsey and Company, New York, NY, USA, 2018.

[11] J. Avrane-Chopard, T. Bourgault, A. Dubey, and L. Moodley, "Big business in small business: cloud services for SMBs," *Telecommunications, Media, and Technology Practice*, McKinsey and Company, New York, NY, USA, 2014.

[12] J. Gao, X. Bai, W. Tsai, and T. Uehara, "Testing as a service (TaaS) on clouds," in *Proceedings of the 7th IEEE International Symposium on Service-Oriented System Engineering*, pp. 212–223, San Francisco Bay, CA, USA, March 2013.

[13] H. Srikanth, M. Cashman, and M. B. Cohen, "Test case prioritization of build acceptance tests for an enterprise cloud application: an industrial case study," *Journal of Systems and Software*, vol. 119, pp. 122–135, 2016.

[14] T. Parveen and S. Tilley, "When to migrate software testing to the cloud?" in *Proceedings of the Third International Conference on Software Testing, Verification, and Validation Workshops*, pp. 424–427, Paris, France, April 2010.

[15] Z. Ganon and I. E. Zilbershtein, "Cloud-based performance testing of network management systems," in *Proceedings of the 14th IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks*, pp. 1–6, Pisa, Italy, June 2009.

[16] L. Riungu-Kalliosaari, "Empirical study on the adoption, use and effects of cloud-based testing," Doctoral Dissertation, Lappeenranta University of Technology, Lappeenranta, Finland, 2014.

[17] D. Kumar and K. K. Mishra, "The impacts of test automation on software's cost, quality and time to market," *Procedia Computer Science*, vol. 79, pp. 8–15, 2016.

[18] S. Ali and N. Ullah, "Critical influential factors for software testing-as-a-service adoption: preliminary findings from systematic literature review," in *Proceedings of the International Conference on Frontiers of Information Technology, Communication, and Computer Engineering*, pp. 1–6, Swat, Pakistan, July 2019.

[19] M. Diaz and J. Sligo, "How software process improvement helped Motorola," *IEEE Software*, vol. 14, no. 5, pp. 75–81, 1997.

[20] S. Ali and S. U. Khan, "Software outsourcing partnership model: an evaluation framework for vendor organizations," *Journal of Systems and Software*, vol. 117, pp. 402–425, 2016.

[21] S. Ali, H. Li, S. U. Khan, Y. Zhao, and L. Li, "Fuzzy multi attribute assessment model for software outsourcing partnership formation," *IEEE Access*, vol. 6, pp. 55431–55461, 2018.

[22] S. Ali and H. Li, "Moving software testing to the cloud: an adoption assessment model based on fuzzy multi-attribute decision making algorithm," in *Proceedings of the 8th International Conference on Software and Computer Applications*, pp. 382–386, Tokyo, Japan, April 2019.

[23] R. A. Khan, M. Y. Idris, S. U. Khan et al., "An evaluation framework for communication and coordination processes in offshore software development outsourcing relationship: using fuzzy methods," *IEEE Access*, vol. 7, pp. 112879–112906, 2019.

[24] X. Bai, M. Li, B. Chen, W. T. Tsai, and J. Gao, "Cloud testing tools," in *Proceedings of the 6th IEEE International Symposium on Service Oriented System Engineering*, pp. 1–12, Irvine, CA, USA, December 2011.

[25] L. Riungu-Kalliosaari, O. Taipale, K. Smolander, and I. Richardson, "Adoption and use of cloud-based testing in practice," *Software Quality Journal*, vol. 24, no. 2, pp. 337–364, 2016.

[26] I. C. Priyanka, I. Chana, and A. Rana, "Empirical evaluation of cloud-based testing techniques," *ACM SIGSOFT Software Engineering Notes*, vol. 37, no. 3, pp. 1–9, 2012.

[27] C. Jia, Y. Cai, Y. T. Yu, and T. H. Tse, "5W + 1H pattern: a perspective of systematic mapping studies and a case study on cloud software testing," *o of em of e*, vol. 116, pp. 206–219, 2016.

[28] W.-T. Tsai and G. Qi, "Integrated fault detection and test algebra for combinatorial testing in TaaS (Testing-as-a-Service)," *m o o e c ce eo*, vol. 68, pp. 108–124, 2016.

[29] S. Zein, N. Salleh, and J. Grundy, "A systematic mapping study of mobile application testing techniques," *o of em of e*, vol. 117, pp. 334–356, 2016.

[30] T. Banzai, H. Koizumi, R. Kanbayashi et al., "Design of a software testing environment for reliable distributed systems using cloud computing technology," in *ocee of he e o mpo m e omp he*, pp. 631–636, Melbourne, Australia, May 2010.

[31] R. M. Ashfaq, "A new testing framework for cloud-based application," *o of of e ee e e em*, vol. 2, no. 3, pp. 157–163, 2017.

[32] P. Jamshidi, A. Ahmad, and C. Pahl, "Cloud migration research: a systematic review," *c o o o omp*, vol. 1, no. 2, pp. 142–157, 2013.

[33] P. Harikrishna and A. Amuthan, "A survey of testing as a service in cloud computing," in *ocee of he e o o fe e ce o omp e omm c o fo m c*, pp. 1–5, Coimbatore, India, 2016.

[34] A. A. Ahmad, P. Brereton, and P. Andras, "A systematic mapping study of empirical studies on software cloud testing methods,," in *ocee of he e o o fe e ce o of e , e ec omp o*, pp. 555–562, Prague, Czech Republic, July 2017.

[35] A. F. Mohammad and H. McHeick, "Cloud services testing: an understanding," *oce omp e c e ce*, vol. 5, pp. 513–520, 2011.

[36] J. Fernandes and F. Gemmer, "Application testing on the cloud: smart testing for agile enterprises," 2020, https://www.oracle.com/technetwork/oem/s317292-app-testing-in-cloud-181639.pdf.

[37] A. Bertolino, L. Nautiyal, and Preeti, "Annotated buzzwords and key references for software testing in the cloud,," in *ocee of he e o o fe e ce o omp , omm c o om o*, pp. 893–900, Noida, India, May 2017.

[38] S. Nachiyappan and S. Justus, "Cloud testing tools and its challenges: a comparative study," *oce omp e c e ce*, vol. 50, pp. 482–489, 2015.

[39] M. Tiitinen, "Key factors for selecting software testing tools," Master Degree in Business Informatics Master Business Informatics, Helsinki Metropolia University of Applied Sciences, Helsinki, Finland, 2013.

[40] M. S. N. Murthy and V. Suma, "A study on cloud computing testing tools," in *c f c e o cee of he h o e o of omp e oc e of*, vol. I, pp. 605–612, Springer, Cham, Switzerland, 2014.

[41] M. S. Geetha Devasena, V. Krishna Kumar, and R. Kingsy Grace, "LTTC: a load testing tool for cloud," in *ocee of e o o fe e ce o omm c o e o*, pp. 689–698, Singapore, 2017.

[42] M. O. Alassa , A. Alharthi, R. J. Walters, and G. B. Wills, "A framework for critical security factors that in uence the decision of cloud adoption by Saudi government agencies," *e em c fo m c*, vol. 34, no. 7, pp. 996–1010, 2017.

[43] N. Alkhater, R. Walters, and G. Wills, "An empirical study of factors in uencing cloud adoption among private sector organizations," *e em c fo m c*, vol. 35, no. 1, 2018.

[44] Y. Alshamaila, S. Papagiannidis, and F. Li, "Cloud computing adoption by SMEs in the north east of England," *o of e p e fo m o eme*, vol. 26, no. 3, pp. 250–275, 2013.

[45] P. Gupta, A. Seetharaman, and J. R. Raj, " e usage and adoption of cloud computing by small and medium businesses," *e o o of fo m o eme*, vol. 33, no. 5, pp. 861–874, 2013.

[46] P.-F. Hsu, S. Ray, and Y.-Y. Li-Hsieh, "Examining cloud computing adoption intention, pricing mechanism, and deployment model," *e o o of fo m o eme*, vol. 34, no. 4, pp. 474–488, 2014.

[47] S. K. Sharma, A. H. Al-Badi, S. M. Govindaluri, and M. H. Al-Kharusi, "Predicting motivators of cloud computing adoption: a developing country perspective," *omp e m eh o*, vol. 62, pp. 61–69, 2016.

[48] R. D. Raut, B. B. Gardas, M. K. Jha, and P. Priyadarshinee, "Examining the critical success factors of cloud computing adoption in the MSMEs by using ISM model," *e o of h ech o o eme e e ch*, vol. 28, no. 2, pp. 125–141, 2017.

[49] K. Incki, B. Ari, and H. Sozer, "A survey of software testing in the cloud,," in *ocee of he h e o o fe e ce of e ec e*, pp. 18–23, Gaithersburg, MD, USA, August 2012.

[50] R. R. Kumar and C. Kumar, "A multi criteria decision making method for cloud service selection and ranking," *e o o of m e omp e e ce*, vol. 9, no. 3, pp. 1–14, 2018.

[51] P. Priyadarshinee, R. D. Raut, M. K. Jha, and B. B. Gardas, "Understanding and predicting the determinants of cloud computing adoption: a two staged hybrid SEM - neural networks approach," *omp e m eh o*, vol. 76, pp. 341–362, 2017.

[52] S. Nazir, S. Shahzad, M. Nazir, and H. U. Rehman, "Evaluating security of software components using analytic network process," in *ocee of he 11 h e o o fe ce o o e of fo m o ech o o*, pp. 183–188, Islamabad, Pakistan, December 2013.

[53] S. Nazir, S. Anwar, S. A. Khan et al., "Software component selection based on quality criteria using the analytic network process," *c pp e*, vol. 2014, Article ID 535970, 12 pages, 2014.

[54] S. Nazir, M. A. Khan, S. Anwar, H. Khan, and M. Nazir, "A novel fuzzy logic based software component selection modeling," in *ocee of he 1 e o o fe ce o fo m o c e ce pp c o*, pp. 1–6, Suwon, South Korea, May 2012.

[55] S. Nazir, S. Shahzad, S. Mahfooz, and M. Nazir, "Fuzzy logic based decision support system for component security evaluation," *e e o o of fo m o ech o o*, vol. 15, no. 2, pp. 224–231, 2018.

[56] S. Tilley and B. Floss, "Hard problems in software testing solutions using testing as a service (TaaS)," vol. 2, no. 1, pp. 1–103, 2014.

[57] W. Kim, S. D. Kim, E. Lee, and S. Lee, "Adoption issues for cloud computing," in *ocee of he 11 h e o o fe ce o fo m o e o e e p p c o e ce*, pp. 3–6, Kuala Lumpur, Malaysia, December 2009.

[58] P. Prodanovic, *e e ho p e pe ec o*, Department of Civil and Environmental Engineering Facility for Intelligent Decision Support, University of Western, Ontario, Canada, 2001.

[59] T. H. Chang and T. C. Wang, "Using the fuzzy multi-criteria decision making approach for measuring the possibility of successful knowledge management," *fo m o c e ce*, vol. 179, no. 13, pp. 355–370, 2009.

[60] A. Sangaiah and A. angavelu, "An exploration of FMCDM approach for evaluating the outcome/success of GSD projects," *e ope , o of ee*, vol. 3, pp. 419–435, 2013.

[61] G. Li, G. Kou, C. Lin, L. Xu, and Y. Liao, "Multi-attribute decision making with generalized fuzzy numbers," *o of he pe o e e ch oc e*, vol. 66, pp. 1793–1803, 2015.

[62] S. Guo and H. Zhao, "Fuzzy best-worst multi-criteria decision-making method and its applications," *o e e e em*, vol. 121, pp. 23–31, 2017.

[63] G. Li, G. Kou, and Y. Peng, "A group decision making model for integrating heterogeneous information," *c o o em , , e e c*, vol. 48, no. 6, pp. 1–11, 2016.

[64] G. Kou, Y. Peng, and G. Wang, "Evaluation of clustering algorithms for nancial risk analysis using MCDM methods," *fo m o c e ce*, vol. 275, pp. 1–12, 2014.

[65] M.-S. Kuo and G.-S. Liang, "A soft computing method of performance evaluation with MCDM based on interval-valued fuzzy numbers," *pp e of omp*, vol. 12, no. 1, pp. 476–485, 2012.

[66] Y.-J. Wang, "Applying FMCDM to evaluate nancial performance of domestic airlines in Taiwan," *pe em h pp c o*, vol. 34, no. 3, pp. 1837–1845, 2008.

[67] L. A. Zadeh, "Fuzzy sets," *fo m o o o*, vol. 8, no. 3, pp. 338–353, 1965.

[68] C. C. Kitchenham, "Guidelines for performing systematic literature reviews in software engineering," Ver. 2.3 EBSE Technical Report, Durham University, Durham, UK, 2007.

[69] T. C. Lethbridge, S. E. Sim, and J. Singer, "Studying software engineers: data collection techniques for software eld studies," *mp c of e ee*, vol. 10, no. 3, pp. 311–341, 2005.

[70] J. W. Creswell, *e e ch e : e, e, e e ho pp o che*, Sage Publications, ousand Oaks, CA, USA, 2013.

[71] P. Priyadarshinee, R. D. Raut, M. K. Jha, and S. S. Kamble, "A cloud computing adoption in Indian SMEs: scale development and validation approach," *e o of h ech o o eme e e ch*, vol. 28, no. 2, pp. 221–245, 2017.

[72] R. K. Yin, *e e e ch: e e ho*, Sage Publications, ousand Oaks, CA, USA, 2009.

[73] M. K. Daskalantonakis, "Achieving higher SEI levels," *of e*, vol. 11, no. 4, pp. 17–24, 1994.

[74] S. Tilley and T. Parveen, "SMART-T: migrating testing to the cloud," *of e e he o : o ec o , p e ef omp e c e ce*, pp. 19–35, Springer, Berlin, Germany, 2012.

[75] A. O. Portillo-Dominguez, M. Wang, J. Murphy, and D. Magoni, "Automated WAIT for cloud-based application testing," in *ocee of he e e h e o o fe ce o of e e , e c o o o hop*, pp. 370–375, Cleveland, OH, USA, March 2014.

[76] A. Gambi, W. Hummer, and S. Dustdar, "Automated testing of cloud-based elastic systems with AUToCLES," in *ocee of he h e o o fe ce o om e of e ee*, pp. 714–717, Palo Alto, CA, USA, November 2013.

[77] M. Arslan, U. Qamar, S. Hassan, and S. Ayub, "Automatic performance analysis of cloud based load testing of web-application & its comparison with traditional load testing," in *ocee of he h e o o fe ce o of e ee*, pp. 140–144, Firenze, Italy, May 2015.

[78] Z. Yang, M. Xiao, X. Zhao, D. Feng, L. Zhang, and H. Song, "Cloud computing and exploration of its application to test eld and PHM," in *ocee of he o o c em e h eme o fe ce*, pp. 1–4, Chengdu, China, October 2016.

[79] H. Muccini, A. D. Francesco, and P. Esposito, "Software testing of mobile applications: challenges and future research directions," in *ocee of he h e o o hop o om o of of e e*, pp. 29–35, Zurich, Switzerland, June 2012.

[80] Capgemini and Sogetti, "World quality report 2014-2015," 2020, https://www.sogeti.com/explore/reports/world-quality-report-2019.

[81] J. Zhou, S. Li, Z. Zhang, and Z. Ye, "Position paper: cloud-based performance testing: issues and challenges," in *o cee of he 1 e o o hop o o op c o e ce*, pp. 55–62, Prague, Czech Republic, April 2013.

[82] L. Ciortea, C. Zam r, S. Bucur, V. Chipounov, and G. Candea, "Cloud 9," *pe em e e*, vol. 43, no. 4, pp. 5–10, 2010.

[83] Q. Gao, W. Wang, G. Wu, X. Li, J. Wei, and H. Zhong, "Migrating loaR}dh¨©[82]S,)-, of (x7©2C7©l©S(x ((C7©nsrxB7©Sc9,"S