

Software Process Improvement Assessment for Cloud Application Based on Fuzzy Analytical Hierarchy Process Method

Surjeet Dalal[1], Akshat Agrawal[2(✉)], Neeraj Dahiya[1], and Jatin Verma[3]

[1] SRM University Sonipat, Gurgaon, India
profsurjeetdalal@gmail.com,
neerajdahiya.cse@gmail.com
[2] Amity University Haryana, Gurugram, India
akshatag20@gmail.com
[3] G.B. Pant Engineering College, Delhi, India
vjatin83@gmail.com

Abstract. The miracles of ordinary programming development face exceptional difficulties which relate to Software process development. Main objective of this method is to make the prioritization-based intelligent plan for Software Process Improvement (SPI). Attainment elements utilizing the Fuzzy Analytical Hierarchy (AHP) Process technique. A special phase that was seen became additionally assessed through using a non-useful gathering with use of SPI. Inside subsequent stage, a multi-degree dynamic AHP tools became applied to fashion out and increase the perceptive depiction of the perceived segments and their requests. The repercussions of the Fuzzy AHP method are novel during this evaluation zone because it has been fairly utilized in cloud software improvement. This paper proposes the radical approach with use of Fuzzy AHP in the examination of Global Software Development (GSD) and SPI, which enables with expelling the untidiness and shortcoming inside the assessment of the procedure development phases related to cloud application headway.

Keywords: Fuzzy AHP · Software process improvement · Cloud computing

1 Introduction

These days, cloud computing is perceived as a predominant registering model in IT frameworks, empowering adaptable, universal, on-request and practical access to a wide pool of shared assets. It is commonly seen that huge and differing associations have embraced the cloud worldview for their data frameworks, getting a charge out of the accompanying two key highlights: low expenses by discharging them from the weight to contribute on equipment foundations and programming licenses, and diminished operational multifaceted nature, as associations can concentrate on the nature of their items and administrations as opposed to on the administration of complex IT frameworks [1]. A typical trait of every one of these applications is that they are information intensive, with the information being produced consistently and

© Springer Nature Switzerland AG 2020
O. Gervasi et al. (Eds.): ICCSA 2020, LNCS 12252, pp. 989–1001, 2020.
https://doi.org/10.1007/978-3-030-58811-3_70

originating from heterogeneous sources, for example, sensors or logical gadgets. Moreover, information generation rates can fluctuate altogether, and the applications may regularly need to process information in an opportune way empowering framework to take remedial/vital operational activities or respond at circumstances desperately (Fig. 1).



Fig. 1. Feature of Cloud computing environment [3]

Hence, such applications make frequently utilize computational assets. We will allude to this sort of utilizations as persistent, information stream applications (CDFA), including a wide scope of uses, for example, logical work processes, pipelines, gushing applications, or whatever other concentrated application where information conditions and simultaneousness play a significant angle [2].

Portability in the cloud can allude to two extraordinary yet interlinked angles:

Legacy programming's modernization planned for misusing current cloud-based advancements.

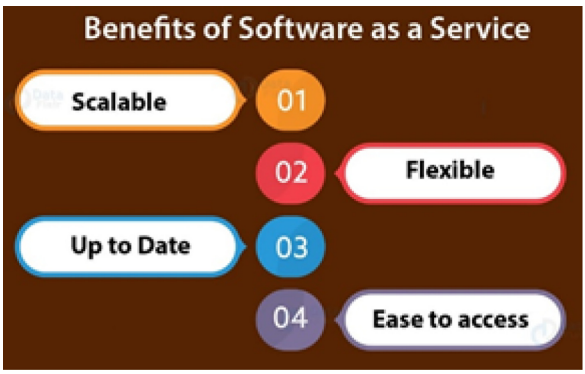

Fig. 2. Benefits of SaaS [1]

Portability of cloud-prepared applications among various cloud stages and suppliers (Fig. 2).

To continue with; we have recognized the purposes of contention which emerge while an application is ported starting with one cloud stages then onto the next. We have distinguished those parts of an application which should be tended to contrastingly in cloud stages. In this area we talk about the accompanying barely any potential clash focuses programming dialects and structures, stage explicit administrations and stage explicit design records [19].

Programming language or potentially systems - Each cloud stage has support for certain language, renditions and structures [4]. The programming dialects and systems that an application has been fabricated will be a significant concern while porting it to another stage. For instance, while Google App Engine (GAE) offers help for Python and Java, Amazon bolsters Java, .NET, PHP, Python and Ruby.

Platform explicit administrations – The time taken for application improvement can be radically decreased by utilizing API's. Rather than programming all of usefulness starting from the earliest stage, they can incorporate it into their application by official to the individual stage APIs [5].

1.1 SaaS App Development

Software-as-a-service (SaaS) applications have certainly become standard even among undertakings. In any case, building up an extraordinary SaaS application isn't trifling. It requires solid comprehension of multi-occupancy, configurability, security, apportioning to construct a genuine SaaS application that can be utilized to serve various sorts of customers.

The most widely recognized issue around the sending and ease of use of utilizations is the failure to test right off the bat in the development cycle intended to address those issues. In Fig. 3, we see that in the truth of undertakings, it is regularly restrictive to give engineers and analyzers situations proportionate to creation from the mix test stages. Surely, for arrangements on physical machines, this speaks to an expense of assets excessively high [7]. This implies the genuine operational design is not tried until the application comes to the pre-creation and creation conditions later in the cycle. This paper suggested approach based on Fuzzy Analytical Hierarchy Process which is applied for purpose of Software Process Improvement (SPI) assessment for Cloud application which emphasis on application deployment as well as migration of the application from one cloud to another.

2 Related Work

G. Fylaktopoulos et al. (2018) depicted the CIRANO stage, a separated Integrated Development Environment (IDE) for cloud-based applications. The proposed stage was attempted to help Model-Driven Development (MDD) and gathering facilitated exertion, empowering the fast improvement of forefront applications in the cloud. The paper introduced at the principal stage the front line in the field of cloud IDEs and delineates the structure, utilization, and nuances of the CIRANO stage. The essential



Fig. 3. Software-as-a-service (SaaS) applications [6]

featured of the proposed stage were presented in two relevant examinations concerning the headway of an application without any planning and porting of a current application. The paper discussed the revelations in relationship with existing instruments and proposed extensions of the phase as future work [8].

Polona Štefanič et al. (2019) expressed that time-fundamental applications, for instance, early notification systems or live event broadcasting, introduced explicit difficulties. They had hard cutoff focuses on Quality of Service necessities that must be kept up, paying little mind to orchestrating changes and moving apexes of weight. In this manner, such applications must modify deftly on-demand, in this way ought to be fit for reconfiguring themselves, nearby the fundamental cloud establishment, to satisfy their necessities. Programming building instruments and ways of thinking right presently did whatever it takes not to help such a perspective. In this paper, they portrayed a structure that had been planned to meet these objectives, as a significant part of the EU SWITCH adventure. SWITCH offers a versatile co-programming structure that gave a pondering layer and an establishment condition, which could help to both show and support the presence example of time-fundamental cloud neighborhood applications. They delineated the building, structure, and execution of the SWITCH parts and depicted how such instruments were applied to three time-fundamental authentic use cases [9].

Fahad Almudarra et al. (2015) discussed issues in making convenient cloud applications using deft programming progression strategies. Sha-Mo-Cloud, a cross-breed cloud framework was made using open cloud organizations for taking care of knowledge including pictures and chronicles got from customer's phones. Customers can move, get the chance to, manage, and share a singular substance-using web interface almost like a versatile application. Adaptable Cloud Computing has been an endless imaginative work zone. Various frameworks and middleware are developed starting late furnishing responses for utilizing appropriated figuring resources on resource-constrained mobile phones. A scarcely any number of works present sound structuring parts for becoming such programming. To spare customer security and

modified data, some a part of the framework was made during a private cloud condition. This add like manner reports the structuring portions of the headway of ShaMoCloud. The item improvement bunch as lately made similar applications using the Team Software Process. The work presented here altogether examines the experiences of the ShaMoCloud progression bunch with four tantamount endeavors that did start late [10].

Philip Churchr et al. (2015) discussed issues in making convenient cloud applications using deft programming progression methods. Sha-Mo-Cloud, a crossbreed cloud framework was made using open cloud organizations for taking care of knowledge including pictures and chronicles got from customer's mobile phones. Customers could move, get the chance to, administer, and share a singular substance-using web interface almost like an adaptable application. Adaptable Cloud Computing had been an endless imaginative work zone. Various frameworks and middleware had been developed starting late offering responses for utilizing disseminated registering resources on resource-constrained PDAs. A barely any number of works introduced sound structuring segments for becoming such programming. To spare customer security and altered data, some a part of the framework was made during a private cloud condition. This work moreover revealed the structuring portions of the headway of ShaMoCloud. The item advancement bunch as lately made near applications using the Team Software Process. The work presented here completely examines the experiences of the ShaMoCloud progression bunch with four equivalent endeavors that did start late [11].

Muhammad Shiraz et al. (2013) clarified that latest enhancements in PDA development had made PDAs as the future figuring and organization get to contraptions. Customers planned to run computational focused applications on Smart Mobile Devices (SMDs) correspondingly as stunning fixed PCs. In any case paying little mind to all the types of progress starting late, SMDs were still low potential figuring contraptions, which were constrained by CPU prospects, memory cutoff, and battery lifetime. Flexible Cloud Computing was the latest sensible response for diminishing this debilitating by expanding the organizations and resources of computational fogs to SMDs on demand premise. In MCC, application offloading was discovered as an item level response for developing application getting ready limits of SMDs. The current offloading figuring's offload computational concentrated applications to remote servers by using differing cloud models. A troublesome piece of such figuring's was the establishment of spread application planning stage at runtime which requires additional preparing resources on SMDs. This paper reviewed existing Disseminated Application Processing Frameworks (DAPFs) for SMDs in the MCC territory. The objective was to highlight gives what is progressively, hard to existing DAPFs in making, realizing, additionally, executing computational concentrated flexible applications inside MCC space [12]. It proposed topical logical arrangement of current DAPFs, reviews current offloading frameworks by using topical logical grouping, and researches the recommendations and essential pieces of ebb and flow offloading structures. Further, it analyzed shared qualities and deviations in such structures on the reason vital boundaries, for instance, offloading expansion, migration granularity, partitioning approach, and development plan. Finally, they put forth open examine issues in circled application planning for MCC that stay to be tended to.

Keiichi Shiohara et al. (2014) expressed that Web application progression was confronted with a growing system multifaceted nature, new specific necessities, dynamic changing of client terminal devices besides, security issues actuated by new parting strategies. During this novel condition, giving new procedures to suitably making Web applications in negligible exertion gets focal. This brought new specific troubles of developing new data showing and new methodology in the arrangement, database mapping, page progress control, endorsement, meeting the administrators, programming, and data and yield interface plan particular districts, as Web application improvement requires those strategies. At an equivalent time, because the significant progression of data and correspondence development (ICT) has been made, ICT propels were carrying exciting changes to our lives and business. Thus, there was a growing prerequisite for creating Web applications for business additionally, ordinary a day presence use. Developing past examination likewise, applications, this paper introduced that data models for creating Web applications might be built using spreadsheets considering a cloud application improvement stage, called FOCAPLAS. This spreadsheet-based making approach introduced their challenges on new frameworks of both the info showing and programming for Web application improvement [13, 14].

Fotis Gonidis et al. (2013) expressed that Cloud application stages picked up reputation and might change the way organization-based cloud applications were made including the utilization of stage basic organizations. A phase central assistance was considered as a bit of programming, which gave certain convenience and is when in doubt offered by methods for a web API, for instance, email, portion, confirmation organization. Regardless, the duplication and upgrade of stage major organizations and the available providers increased the test for the application originators to arrange them and deal with the heterogeneous providers' web APIs. Along these lines, another strategy of making applications should be grasped in which fashioner's impact distinctive stage basic organizations openly from the target application stages. To this end, this paper introduced an improvement framework whose goal was to engage the consistent coordination of the stage organizations and to allow the predictable use of the strong providers by relieving the heterogeneities among them [15].

Rear Benfenatk et al. (2014) described a nonexclusive methodology for self-loader improvement of cloud-based business applications. This could be used by non-IT experts, for example, colleagues, who set off a business application progression by simply communicating its necessities in the wording of business functionalities and constraints, QoS boundaries, likewise, her/his tendencies. From these functionalities and objectives, Linked USDL essentials reports are thus created. These records gave the reason to the cloud organization exposure and dispatch the customized improvement of cloud business applications. We present the essential enhancements in their model [16–18].

3 Problem Formulation

During the past decade, the software industry has been increasingly more worried about Software process improvement (SPI). Therefore, we have seen an expansion of models and activities all asserting to improve the probability of prevailing with SPI activities. SPI has its underlying foundations in quality administration and is intently identified with "second era" authoritative advancement draws near, explicitly to hierarchical learning. Understanding the "quality unrest" is, along these lines, a significant essential for comprehension SPI.

Software process improvement activities bolster the association in accomplishing its key business objectives more adequately, where the business objectives could be conveying programming quicker to the market, improving quality and decreasing or dispensing with squander [19, 20]. The goal is to work more astute and to fabricate programming better, quicker also, less expensive than contenders. Software process improvement makes business sense, and it gives an arrival on speculation. Software process improvement (SPI) is a drawn-out excursion, which is made agreeable by numerous methods.



Fig. 4. Steps in process improvement [9]

The most predominant and favored arrangement is an information-driven technique with which programming advancement associations are testing. To see and feel of information and its administration, it has gotten fundamental to have normalized information the board apparatus (KMT) that involves determinations like-procurement, portrayal, sharing, and conveying. Albeit a few devices and strategies are accessible for overseeing information to take care of area issues, it is felt in the information society that no standard KM devices exist that would encourage SPI. In this bit of usage work, the creators plot the highlights that are considered noteworthy to execute a KMT that drives the excursion of SPI. Four procedure zones are picked, and four subsystems are

distinguished in covering these procedure zones [20, 21]. A progression of studies directed among associations requiring the help of a KMT in making a definitive SPI activity is likewise talked about with intricate and noteworthy outcomes. Ramifications of this work request the participation of programming advancement organizations with the exploration network in finding a superior way to deal with their improvement program [22–24].

This section expands and coordinates models from earlier inquire about by playing out an observational examination of the key components for accomplishment in SPI. Cloud selection for a wide range of programming is expanding, and most associations no longer consider cloud framework as less secure than on-premises arrangements; rather, they consider it to be the main conceivable elective when versatility, cost, and time to esteem are significant variables. Late penetrates of security by government organizations have made cloud clients justifiably apprehensive, yet cloud suppliers are offering more options for the area of the cloud servers and improving security using private encryption keys that are held by the client as opposed to the supplier. It's additionally imperative to consider the relative hazard for the kind of data being put away in the cloud condition: With this approach, the models and examination put away in the open cloud may speak to some important licensed innovation, yet infrequently incorporate any actually recognizable data about the client association's clients [23].

4 Fuzzy Analytical Hierarchy Process Method

There are obtrusive multi-rule dynamic strategies, paying little brain to, Analytic Hierarchy Process (AHP) has been taken into consideration because the foremost elementary structure. The AHP technique required the theoretical records of the authorities in multi-degree dynamic issues. Arising subsequent are three center strides of utilizing the AHP approach. Making of the chain of criticalness structure of the irregular desire issue. Figure weight of measures at every diploma of the chain of criticalness with the help of a pairwise alliance. Figure the standardized burdens to pick out the final positions. The old-style AHP cannot influence a pioneer's peculiarity additionally, ambiguousness identified with surveying the necessities of various measures. As such, the woolen hypothesis mixes the AHP framework known as cushioned AHP that is used to pick capably right and fulfilling selections constantly and faulty issues. This approach will be utilized for each quantitative and exceptional data within the MCDM issues. Triangular warm numbers are applied all through this approach for slanting closer to reviews of the requirements and affect need the immensity of unequivocal models making use of the degree appraisal procedure [24, 25]. During this assessment, we have utilized the padded AHP proposed by Chang which affords incessantly correct and predictable results as disconnected to the standard AHP technique (Fig. 5).

The Fuzzy AHP approach has been applied to form the proclaimed fulfillment factors of SPI and its key classes.

A. Graphical levels of intensity version of the apparent sections and their classes.

In this examination, the graphical astounding structure turned into made established upon the gadget gave through Ramasubu and conversations with the review masters

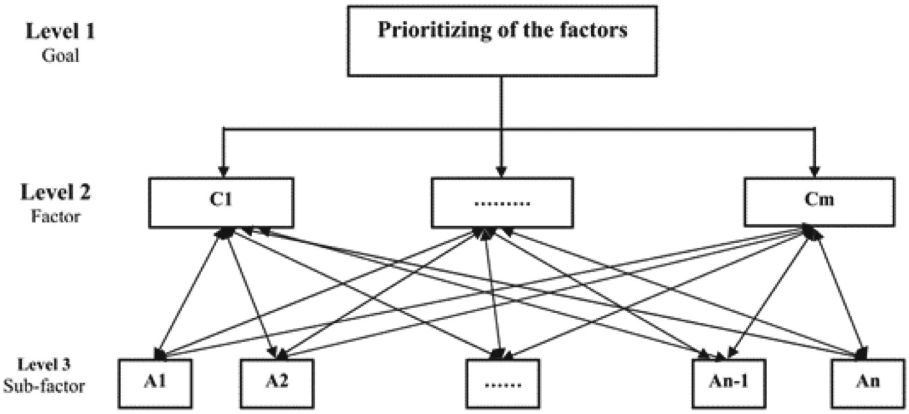

Fig. 5. Fuzzy AHP decision hierarchy [1]

(Fig. 4). The key goal of this assessment is gathered on the very fine degree of the chain of the noteworthiness of association (degree 1), while the delineations and their thriving sections at stage 2 and degree 3 autonomously.

B. examine development for inclining in the direction of the achievement factors the apparent achievement elements and their social affair turned into diagramed utilizing the survey examine method [26, 27]. The overview changed into made to amass statistics from the SPI masters to altogether chance underwrite the conspicuous achievement elements and their classes within the space of GSD. During this evaluation, the expertise became accrued from seventy-three directors together with prodigies, experience supervisors, programming engineers/engineers, programming sketchers, analyzers, instructive aces, and so on. The diagram human beings were welcomed making use of email and clean online life structures, which include ResearchGate, LinkedIn, and Facebook. The people have been the key masters that have been really pulled into the structure development experiences. The online information was amassed from latest timeline and for that reason the full-scale records collection takes one and half month. We have accrued a sum of seventy-five reactions given by means of the diagram people. The evaluation reactions have been explored by using the focal author of the paper and discovered six missing reactions. The missing reactions had been considering 75 reactions that had been considered at some point of this evaluation. We applied a 5-point Likert scale, and consequently the reactions have been referenced as: "very concur", "concur", "authentic", "veer off" and "mainly discredit this idea". It's great to believe the fair point inside the structure examination scale, considering the manner that the nonattendance of a neural choice could oblige the respondents to offer an unequal response (for example tremendous or however negative) [25–28].

C. Pilot trying out of the evaluate: - A pilot appraisal of the evaluation tool become driven by taking into consideration the substance and key factors of the audit. The examination turned into reviewed by using 5 experts identified with the SPI and GSD experiences, which includes the aspect organizing teachers from the metropolis. The going with facts turned into checked within the reactions of the pilot study. Responses were sketched bent test the consistency many of the respondents. Responses have been poor right down to understand any solicitation abandoned a vital open door by the respondents. The instrument was additionally amassed in subjects that could oppositely influence the reaction rate. The recommendation and comments gave through the pilot have a look at respondents may want to enhance the assessment response rate through and pretty [13]. During this assessment, the likelihood of the overview device changed into progressed via considering the proposals given by way of the pilot look at wonders. The comments and proposals are abbreviated in going with focuses. Far and away most of the feedback had been identified with an instrument plan moreover, the experts proposed introducing the assessment factors during a stunning structure for clearness and insight. The ace's in addition quarreling over the obliged facts gave about the respondent association. It had been invigorated within the respectable series of the audit overview. Because the pilot looks at became fashioned and directed the have a look at approach, there has been no skipped question from the respondent. The tool has been adjusted and engaged problem to the appraisal given through the pilot observe people [29, 30].

5 Cloud Application Development Evaluation Using the Proposed Model

The proposed model is represented with a numerical case model. The point by point delineation is talked about beneath. A successful cloud-based Health application can help patients to improve their health. In any case, the determination of cloud-based Health applications is a moderately troublesome errand. The issue can be settled by applying AHP and Fuzzy TOPSIS procedures. The blend of AHP and Fuzzy TOPSIS has been discovered reasonable in various fields to take care of MCDM issues [30, 31]. The determinations of models and sub-rules for cloud-based Health applications are extremely pivotal. In the current examination, a double methodology was embraced to select the standards and sub-rules. In the primary stage, different measures were distinguished by alluding to the writing in the important field. Consequently, suppositions from three specialists from the scholarly community and industry were taken on the distinguished components [32–34] (Fig. 6).

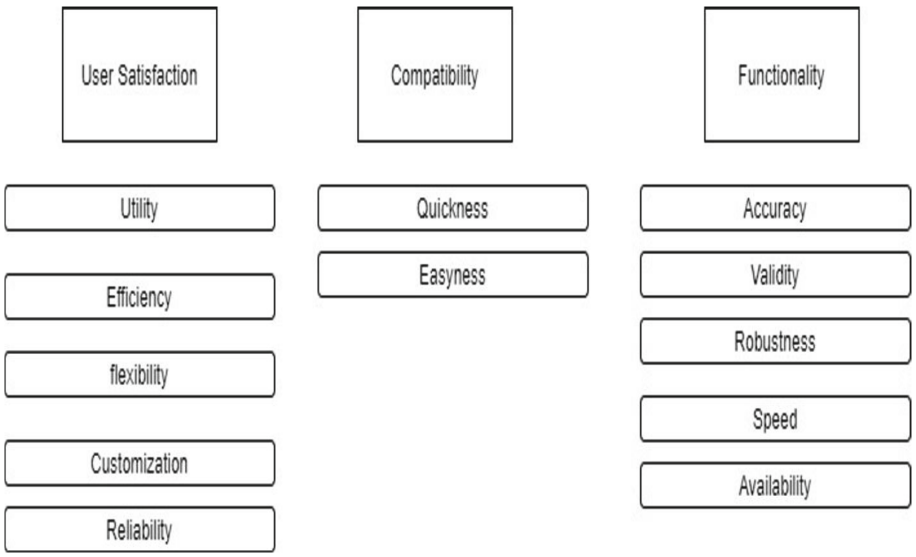

Fig. 6. Hierarchy of cloud based health application model [2]

6 Results Analysis and Conclusion

The health application has become fundamental as of late because of the digitization of society. This framework can oversee problems even more quickly if it very well may be amalgamated with the conventional human services framework. In this computerized world, a viable cloud Health application can assist individuals with dealing with their health and become progressively educated with respect to health.

Be that as it may, with the expanding number of health-related applications, it is hard to choose which applications are appropriate for the clients what is more, which are most certainly not. This examination has attempted to reveal insight into this course.

The point of this exploration is to frame a model dependent on AHP and fuzzy TOPSIS to look at and assess the exhibitions of various cloud-based health applications. For assessing exhibitions of various cloud-based health applications, nine rules and thirty-two sub-standards have been thought of. Diverse cloud-based health applications are looked at and organized dependent on the over thirty-two components. To catch the vulnerability relating to the human dynamic procedure, fuzzy ideas have been applied. Client fulfillment, usefulness, simple to learn and use, and data quality were significant factors for cloud-based health innovation determination. Solid Out has risen as the best cloud-based health application among all the options followed by Noom Weight Loss Coach. Strikingly, a huge variety of the positioning has been watched for weighted and unweighted techniques. The sub-rules assume a significant job in cloud-based health innovation choice. In any case, a leader can just utilize the fuzzy TOPSIS to rank the choices.

References

1. Khan, A.A., Basri, S., Dominic, P.D.D.: A propose framework for requirement change management in global software development. In: International Conference on Computer & Information Science, ICCIS, pp. 944–947 (2012)
2. Unterkal msteiner, M., Gorschek, T., Islam, M.M.K.A., Cheng, K.C., Permadi, R.B., Feldt, R.: Evaluation and measurement of software process improvement—a systematic literature review. IEEE Trans. Softw. Eng. **38,** 398–424 (2012)
3. Coleman, G., O'Connor, R.: Investigating software process in practice: a grounded theory perspective. J. Syst. Softw. **81**, 772–784 (2008)
4. Pettersson, F., Ivarsson, M., Gorschek, T., Öhman, P.: A practitioner's guide to lightweight software process assessment and improvement planning. J. Syst. Softw. **81**, 972–995 (2008)
5. Chen, L., Babar, M.A., Zhang, H.: Towards an evidence-based understanding of electronic data sources. In: Proceedings of the 14th International Conference on Evaluation and Assessment in Software Engineering, EASE, UK (2010)
6. Herbsleb, D.J., Moitra, D.: Global software development. IEEE Softw. **18**(2), 16–20 (2001)
7. Khan, A.A.: A framework for assisting software process improvement program in global software development: student research abstract. In: Proceedings of the 31st Annual ACM Symposium on Applied Computing. ACM (2016)
8. Kuhrmann, M., Konopka, C., Nellemann, P., Diebold, P., Münch, J.: Software process improvement: where is the evidence: initial findings from a systematic mapping study. In: Proceedings of the International Conference on Software and System Process (2015)
9. Paulk, M., Curtis, B., Chrissis, B.M., Weber, C.: Capability Maturity Model for Software, Version 1.1. CMU/SEI-93-TR-24 (1993)
10. SEI: Standard CMMI Appraisal Method for Process Improvement (SCAMPI)A, Version 1.3: Method Definition Document (CMU/SEI-2011-HB-001), Software Engineering Institute, Carnegie Mellon University, Pittsburgh, PA (2011)
11. SO: ISO 9000, Quality Management Systems-Fundamentals and Vocabulary International Organization for Standardization, Technical Report, ISO9000:2005 (2005)
12. Khan, A.A., Shameem, M., Kumar, R.R., et al.: Applied Soft Computing Journal 83 (2019) 10564823[13] ISO, ISO/IEC Information Technology- Process Assessment - Part 4: Guidance on Use for Process Improvement and Process Capability Determination, International Organization for Standardization, Technical ReportISO/IEC 15504-4:2004 (2004)
13. Gillies, A.: Software Quality: Theory and Management, International Thomson Computer Press (2011)
14. Abrahamsson, P.: Measuring the success of software process improvement: the dimensions. arXiv preprint arXiv:1309.4645 (2013)
15. Bano, M., Ikram, N.: Software process improvement: a systematic literature review. In: 15th International Multitopic Conference, INMIC, pp. 459–464 (2012)
16. Humphrey, W.S.: A Discipline for Software Engineering. Addison Wesley (1995)
17. Ayhan, M.B.: A fuzzy AHP approach for supplier selection problem: a case study in a Gear motor company. arXiv preprint arXiv:1311.2886 (2013)
18. Güngör, Z., Serhadlıoğlu, G., Kesen, S.E.: A fuzzy AHP approach to personnel selection problem. Appl. Soft Comput. **9**(2), 641–646 (2009)
19. Chamodrakas, I., Batis, D., Martakos, D.: Supplier selection in electronic marketplaces using satisficing and fuzzy AHP. Expert Syst. Appl. **37**(1), 490–498 (2010)
20. Ayhan, M.B.: Fuzzy tops is application for supplier selection problem. Int. J. Inf. Bus. Manag. **5**(2), 159–174 (2013)

21. Sulayman, M., Urquhart, C., Mendes, E., Seidel, S.: Software process improvement success factors for small and medium web companies: a qualitative study. Inf. Softw. Technol. **54**, 479–500 (2012)
22. Niazi, M., et al.: Challenges of project management in global software development: aclient-vendor analysis. Inf. Softw. Technol. **80**, 1–19 (2016)
23. Gillies, A.C.: Software Quality, 2nd edn. International Thomson Computer Press, London (1997)
24. Yamamura, G.: Software Process Satisfied Employees. IEEE Softw., 83–85 (1999)
25. Habib, Z.: The Critical success factors in implementation of software process improvement efforts: CSFs, motivators & obstacles, MSc. thesis, University of Gothenburg, Sweden (2009)
26. Paulk, M.: Capability Maturity Model for Software, Wiley Online Library (1993). S.E.I., Process Maturity Profile, Software Engineering Institute Carnegie Mellon University (2004)
27. Khan, A.A., Keung, J., Niazi, M., Hussain, S., Shameem, M.: GSEPIM: a road map for software process assessment and improvement in the domain of global software development. J. Softw. Evol. Process. **31**(1), e1988 (2019)
28. Niazi, M.: A framework for assisting the design of effective software process improvement implementation strategies, Ph.D. thesis, University of Technology Sydney (2004)
29. Babar, M.A., Niazi, M.: Implementing software process improvement initiatives: an analysis of Vietnamese practitioners' views. In: IEEE International Conference on Global Software Engineering, ICGSE 2008, pp. 67–76 (2008)
30. Khan, A.A., Basri, S., Dominic, P.D.D., Amin, F.E.: Communication risks and best practices in global software development during requirements change management: a systematic literature review protocol. Res. J. Appl. Sci. Eng. Technol. **6**, 3514–3519 (2013)
31. Dahiya, N.: A new variant of bat algorithm and clustering approach for optimization problems. In: Proceedings of First International Conference on Computational Intelligence and Communication Technologies (2017)
32. Behera, R.K., Shukla, S., Rath, S.K., Misra, S.: Software reliability assessment using machine learning technique. In: Gervasi, O., et al. (eds.) ICCSA 2018. LNCS, vol. 10964, pp. 403–411. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-95174-4_32
33. Behera, R.K., Naik, D., Rath, S.K., Dharavath, R.: Genetic algorithm-based community detection in large-scale social networks. Neural Comput. Appl. **32**(13), 9649–9665 (2019). https://doi.org/10.1007/s00521-019-04487-0
34. Shukla, S., Behera, R.K., Misra, S., Rath, S.K.: Software Reliability Assessment Using Deep Learning Technique. In: Chakraverty, S., Goel, A., Misra, S. (eds.) Towards Extensible and Adaptable Methods in Computing, pp. 57–68. Springer, Singapore (2018). https://doi.org/10.1007/978-981-13-2348-5_5
35. Behera, R.K., Rath, S.K., Misra, S., Damaševičius, R., Maskeliūnas, R.: Large scale community detection using a small world model. Appl. Sci. **7**(11), 1173 (2017)