International Journal of Computer Trends and Technology
ISSN: 2231-2803 / https://doi.org/10.14445/22312803/IJCTT-V68I2P105
Volume 68 Issue 2, 25-33, February 2020

Original Article

Supporting Structure for Prioritizing and Assigning Six Sigma Software Projects Using Fuzzy Logic Topsis and Fuzzy Logic Expert System

Chandrakanth G Pujari

MCA Department, Dr AIT Bangalore

Received Date: 06 January 2020
Revised Date: 14 February 2020
Accepted Date: 16 February 2020

Abstract – *The software project selection process system can be encountered as a very important action in the success of six sigma software projects. Such a manner that is prioritizing and assigning software projects to execute software development teams is recognized, the very important step in software process plentiful research studies has been undergoing into six sigma software project selection, no one has been concentrate on selecting and assigning software projects as an articulate process occurring simultaneously. In this concern, this paper presents a supporting structure for decision making choosing and allocating the six sigma software projects to execute software project development teams, the main important action of the six sigma software project selection process is selected. Coming after, recognizing six sigma potential software projects in the small and medium scale industries. The fuzzy logic TOPSIS methodology is used to rank them. Further, the strong effect and effort indices for every software project are computed. Afterwords fuzzy logic expert system is used to assign the software projects to six sigma champions. Finally, a case study in the software organizations is presented, and the supporting structure is discussed to illustrate its software developed application.*

Keywords – *Six Sigma Project Selection, Fuzzy Logic Expert System, TOPSIS, Software Project Allocation, Multiple Criteria Decision Making.*

I. INTRODUCTION

In order to execute software projects using six sigma methodologies named DMAIC (Define, Measure, Analyze, Improve, and Control), this methodology is considered a very important key to achieving software project goals [1][18]. Now a day's, technology demonstrates the present software industry, covering the way for the construction of dynamic environments [2][19]. Software companies require continuous improvements in order to survive fierce completion [3][20]. The six sigma methodology concentrates on the most famous lean six sigma tools for waste elimination, expenditure reduction and improvement of quality throughout the organizations [4][21]. And to continuously define and solve several problems. This paper discussed the prioritizing of six sigma software projects, but there is no other model developed for assigning software projects to implement teams and six sigma champions. This is the best model for prioritizing and choosing six sigma software projects and also assigns them to corresponding champions simultaneously [22][5]. Therefore, this article is innovative in a serious attempt to achieve something to develop a comprehensive supporting structure, presenting a step by step software process to identify potential six sigma software projects throughout an s/w industry, prioritize potential software projects using fuzzy logic TOPSIS, and assigning them to champions, according to their proficiencies, through a fuzzy logic methodology, at that time [23][7].

II. LITERATURE REVIEW

A. Six sigma software project selection criteria

In the year of 1987, Motorola company was introduced the six sigma methodology, and it was utilized by electric components in early 1995. Its intention was to produce long term defect levels, DPMO [Defects Per Million Opportunities] was below 3.4σ [14][15][24]. Using six sigma improve the creative activities and produces high-level outputs, enhancing workers skills, improving operational processes, and encouraging changes [25][26][27]. Six Sigma methodologies is a technical tool for creating value. This is for achieving business success [28]. It wrongly selected six sigma software projects beginning; they will be incapable and lead to a loss in time and costs [29][30].

According to research carried out in this field, a set of criteria that can be utilized in choosing six sigma software projects are known and presented as follows:

- Software customer satisfaction
- Investment returns
- Software project scope
- Availability of resources, including Manpower
- Time requirement
- Satisfaction of software developers from software projects
- Increasing profits and financial efficiency.

B. Six sigma potential software projects identification

In potential software project selection to collect information from various sources, in this regard, the very important resources are:

- Software customer satisfaction estimation results
- Software customers voice
- Software suppliers
- Software analysis competitor
- Software cost of quality and reliability reports.
- Software waste identification
- Software previous projects

The six sigma software projects can be identified in every software industry by collecting software information from the brainstorming meeting.

C. Fuzzy logic TOPSIS model

First, Hwang and Yoon proposed a model as the TOPSIS model. This is one of the multiple attribute decision making models. This is used for selecting the best alternative and selecting software projects. The TOPSIS model is considered a major multiple attribute decision-making method in comparison with other related methods like the analytical hierarchical approach.

- TOPSIS can consist of an unrestricted level of software criteria and software performance properties.
- TOPSIS can have changes in one property that can be neutralized by other properties in an indirect or direct manner.
- TOPSIS provides us with a systematic procedure, streamlined with a relatively simple evaluation process.
- TOPSIS avoids Pairwise comparisons needed by techniques such as hierarchical analytical process.
- TOPSIS not only provides us with a preferential ranking of alternatives but also computes a numerical value for each alternative and find the rank of alternative.

D. Assigning the six sigma software projects

Six sigma software projects are assigned to the relevant champions, including six sigma black belts and six sigma green belts. Each software project is allocating Six sigma to relevant champions, according to its very important step in the six sigma software projects. Consequently, in this step, it is interesting to assign software projects to qualified champions by taking into account the amount of software developers' effort and impact of software projects that are allowed in order to postpone some software projects. The help of fuzzy expert systems has extensively been maximized.

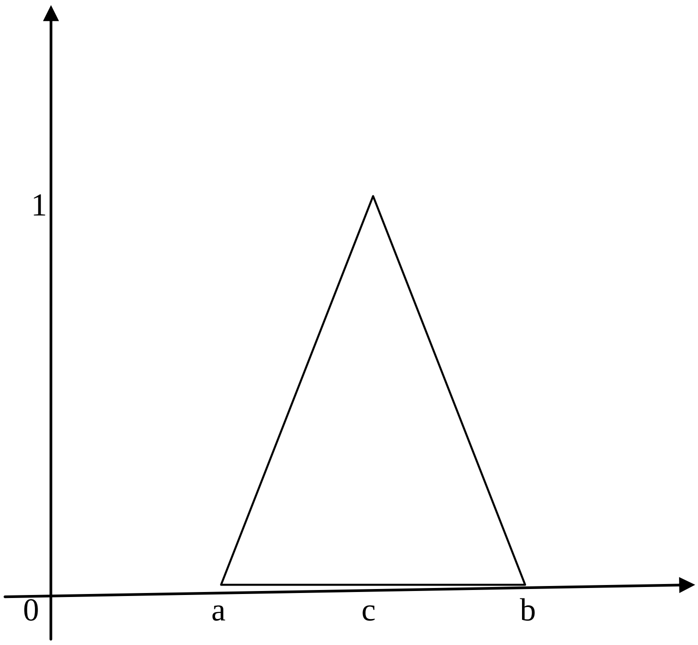


Fig.1 Triangular fuzzy logic number and the membership function

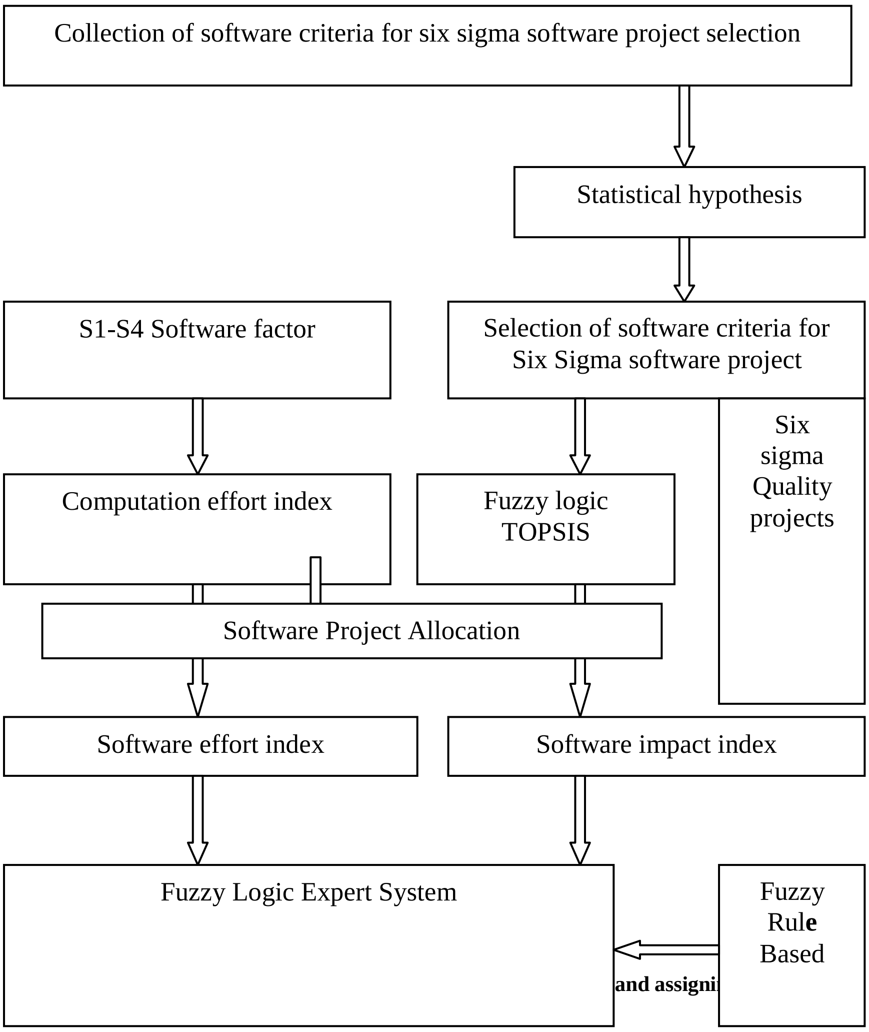
$$\mu_{a^-}(y) = \begin{cases} 0 & y < a \\ \frac{y-a}{b-a} & a \le y \le b \\ \frac{c-y}{c-b} & b \le y \le c \\ 0 & y > a \end{cases}$$

Table 1. Fuzzy logic parameters and their triangular fuzzy logic numbers

Fuzzy logic parameters for the ratings		**Fuzzy logic parameters for the relative importance weights of five criteria**	
(0.000, 0.000, 1.000)	VP	(0.000, 0.000, 1.000)	VL
(0.000, 0.000, 1.000)	P	(0.000, 0.100, 0.300)	L
(1.000, 3.000, 5.000)	MP	(0.100, 0.300, 0.500)	ML
(3.000, 5.000, 7.000)	F	(0.300, 0.500, 0.700)	M
(5.000, 7.000, 9.000)	MG	(0.500, 0.700, 0.900)	MH
(7.000, 9.000, 10.000)	G	(0.700, 0.900, 1.000)	H
(9.000, 10.000, 10.000)	VG	(0.900, 1.000, 1.000)	VH

III. PROPOSED SUPPORTING STRUCTURE FOR ASSIGNING AND CHOOSING THE SIX SIGMA SOFTWARE PROJECTS

The main objective of this article is to provide a supporting structure for prioritizing and assigning The six sigma software projects for executing in organizational units. The phases of the presented supporting structure are illustrated in the following figure 2.



IV. CASE STUDY

Wipro software organization is one of the top software projects suppliers in India. In the first step of the presented supporting structure, very important software criteria were considered in the organizational unit. In this regard, we considered 50 software developers, software managers of software development units, Six Sigma Champions, including black belts, green belts, quality assurance, and quality control experts, then the response rate was 80%, the validity of the questionnaire, such as

resources, relationship with strategic objectives, project scope etc. The reliability of the questionnaire was computed using $\alpha = 0.754$ d.f statistical hypothesis H0: $\mu i <= 3$,
H1: $\mu i>3$ is used to choose the software criteria with above-average importance. The t-test was done at the 0.05 level of significance, and n-1 d.f is investigated. The results are shown in the following table 2.

Table 2. t-test results with 95% confidence level of significance

Software criteria	**t-test value**	**n**		**Significance 2-tailed**	**95% confidence level of significance**	
					L	**U**
Software resources availability	3	0.811	32	0.463	--0.2669	0.5678
Software investment return	3	-2.199	32	0.022	-0.6842	-0.0433
Special requirements to improve	3	0.599	32	0.550	-0.3111	0.5352
Software developers satisfaction from software projects	3	-2.515	32	0.016	-0.7225	-0.0862
Time requirement	3	4.955	32	0.000	0.4911	1.1663
Scope of software projects	3	-4.699	32	0.000	-0.9663	-0.3662
Profits	3	0.211	32	0.850	-0.2938	0.3336
Relations with strategic objectives	3	6.218	32	0.000	0.5776	1.1206
Measurability	3	-2.940	32	0.005	-0.7211	-0.1388
Software customer satisfaction	3	-2.793	32	0.010	-0.7982	-0.1316
Key processes of the organization	3	0.454	32	0.650	-0.3100	0.6111
Ability to access information	3	-2.600	32	0.013	-0.7672	-0.0911

In the above t-test table2 shows the values of statistical hypothesis in those values of significance is more than 0.05, H0 is not rejected. In the above, for the six software criteria, those are more important than other criteria in this organizational unit. These software criteria will be used for prioritizing the six sigma software project by the fuzzy logic TOPSIS technique.

A. Choosing the six sigma potential software projects

In this section, for choosing the six sigma software potential projects, the software projects initial study is discussed in the studied software industry. Software potential projects, which have been removed from assessment reports of software customer satisfaction, software developer suggestion software competitor analysis, the software waste costs and the outcome of cost of quality, considered by software quality control and software quality assurance personnel of the software industry.

B. Prioritizing the six sigma software projects using fuzzy logic TOPSIS

Six potential software projects (p1,p2,p3,p4,p5, and p6) have been chosen for prioritization. These software projects have strengths and weaknesses. For prioritizing the six sigma software projects, six criteria's are:

C1: Relation with the strategic objectives
C2: Key processes of the organization
C3: Profits
C4: Special requirements to improve
C5: Time requirement
C6: Software resource availability

Among these criteria, C1, C2, C3, C4, and C6 are the most profited criteria, and C5 is the cost criteria. The advantage criterion means that a higher value is better, while, for the cost criterion, the reverse is valid. The executive software committee of three decision-makers, D1, D2, and D3, these decision-makers are experts and experienced in S/W quality control, and six sigma software projects are used to assess the importance of the software criteria, as shown in the following table 3.

Table 3. Importance weights of software criteria

Software criteria	Decision-makers			Mean weights
	D1	D2	D3	
C1	MP	MP	M	(0.030, 0.170, 0.370)
C2	G	G	G	(0.500, 0.700, 0.900)
C3	VG	VG	G	(0.630, 0.830, 0.970)
C4	P	MP	P	(0.009, 0.030, 0.170)
C5	VG	MG	G	(0.500, 0.700, 0.870)
C6	H	H	VG	(0.570, 0.770, 0.930)

Table 4. Fuzzy logic decision-making matrix and their fuzzy logic weights

	C1	C2	C3	C4	C5	C6
LV	(0.090,0.170, 0.370)	(0.500,0.700, 0.900)	(0.630,0.830,0.970)	(0.000,0.030, 0.170)	(0.500,0.700, 0.870)	(0.570,0.770,0.910)
P1	(0.330,1.670, 3.670)	(4.330,6.330, 8.330)	(7.670,9.330,10.000)	(3.000,5.000, 7.000)	(0.380,1.670, 3.670)	(4.330,6.330,8.330)
P2	(1.670,3.670,50670)	(6.330,8.330, 9.670)	(5.670,7.670,9.330)	(0.000,0.330, 1.670)	(1.670,3.670, 5.670)	(5.670,7.670,9.330)
P3	(9.000,10.000,10.000)	(3.670,5.670, 7.670)	(5.670,7.670,9.330)	(4.330,6.330, 8.330)	(2.330,4.330, 6.330)	(3.000,5.000,7.000)
P4	(0.000,1.000,3.000)	(3.670,5.670,7.670)	(9.000,10.000, 10.000)	(5.000,7.000,9.000)	(2.330,4.330,6.330)	(0.000,0.670,2.330)

P5	(2.330,4.330,6.330)	(0.330,1.670,3.670)	(8.330,9.670,10.000)	(6.380,8.330,9.670)	(6.330,8.330,9.670)	(0.670,2.330,4.330)
P6	(6.330,8.330,9.670)	(0.000,0.000,1.000)	(8.330,9.670,10.000)	(4.330,6.330,8.330)	(1.670,3.675,5.670)	(8.330,9.670,10.000)

Table 5. weighted normalized fuzzy logic decision matrix

	C1	C2	C3	C4	$C5^-$	C6
P1	(0.000,0.030,0.130)	(0.220,0.460,0.780)	(0.490,0.780,0.970)	(0.000,0.020,0.120)	(0.050,0.140,0.080)	(0.250,0.490,0.780)
P2	(0.010,0.060,0.210)	(0.330,0.600,0.900)	(0.360,0.600,0.900)	(0.000,0.000,0.030)	(0.030,0.060,0.050)	(0.320,0.590,0.870)
P3	(0.030,0.170,0.370)	(0.190,0.410,0.710)	(0.360,0.640,0.900)	(0.000,0.020,0.140)	(0.030,0.050,0.050)	(0.170,0.380,0.650)
P4	(0.000,0.020,0.110)	(0.190,0.410,0.710)	(0.570,0.830,0.970)	(0.000,0.020,0.160)	(0.030,0.050,0.050)	(0.007,0.050,0.220)
P5	(0.010,0.070,0.230)	(0.020,0.120,0.340)	(0.530,0.810,0.970)	(0.000,0.030,0.170)	(0.020,0.030,0.030)	(0.040,0.180,0.400)
P6	(0.020,0.140,0.350)	(0.000,0.000,0.090)	(0.530,0.810,0.970)	(0.000,0.020,0.140)	(0.030,0.060,0.050)	(0.470,0.740,0.930)

Table 6. Positive and negative ideal solutions

	C1	C2	C3	C4	$C5^-$	C6
P^+	(0.370,0.370,0.370)	(0.900,0.900,0.900)	(0.970,0.970,0.970)	(0.170,0.170,0.170)	(0.140,0.140,0.140)	(0.930,0.930,0.930)
P^-	(0.000,0.000,0.000)	(0.020,0.020,0.20)	(0.360,0.360,0.360)	(0.000,0.000,0.000)	(0.020,0.020,0.020)	(0.000,0.000,0.000)

Table 7. The distance of the alternatives from the positive-negative ideals and their final weights and ranks

Alternatives	D^+	D^-	$D^+ + D^-$	Final weight	Rank
P1	1.7700	1.7470	3.5140	0.4970	2
P2	1.7200	1.8160	3.5350	0.5140	1
P3	1.9300	1.6330	3.5630	0.4580	3
P4	2.1500	1.2600	3.4140	0.3690	5
P5	2.2800	1.1590	3.4430	0.3370	6
P6	1.8900	1.5780	3.4670	0.4550	4

Table 8. Decision making normalized matrix

P1	(0.230,0.430,0.630)	(0.170,0.370,0.570)	(0.030,0.170,0.370)	(0.770,0.930,1.000)
P2	(0.770,0.930,1.000)	(0.070,0.230,0.430)	(0.170,0.370,0.570)	(0.630,0.830,0.970)
P3	(0.770,0.930,1.000)	(0.300,0.500,0.700)	(0.230,0.430,0.630)	(0.000,0.030,0.170)
P4	(0.070,0.230,0.430)	(0.770,0.930,1.000)	(0.230,0.430,0.630)	(0.170,0.370,0.570)
P5	(0.000,0.030,0.170)	(0.570,0.770,0.930)	(0.630,0.830,0.970)	(0.000,0.030,0.170)
P6	(0.230,0.430,0.630)	(0.000,0.030,0.170)	(0.170,0.310,0.570)	(0.300,0.500,0.700)

Table 9. The value of impact and effort for each alternative

Projects	e	Effort Index	Impact Index	Index ES	Project allocation result
P1	(0.300,0.480,0.640)	0.47360	0.4970	0.4960	Green Belt
P2	(0.410,0.590,0.740)	0.5850	0.5140	0.6060	Black Belt
P3	(0.330,0.480,0.630)	0.4750	0.4580	0.4480	Green Belt
P4	(0.310,0.490,0.660)	0.4889	0.3690	0.3360	Give Up or Postpone
P5	(0.300,0.420,0.560)	0.4208	0.3370	0.2960	Give Up or Postpone
P6	(0.180,0.330,0.520)	0.3375	0.4550	0.4440	Green Belt

V. SUMMARY AND CONCLUSION

The six sigma methodology is considered a software quality enhancement approach, including software waste reduction and profit-maximizing. The most important phases in executing a six sigma software project, the main purpose of this article are to develop a new model for choosing the most effective software projects in the industry and assigning them to software projects champions. The main purpose of this article is to develop a new model for choosing the most effective software projects in the industry and assigning them to software projects champions. In this article, using a line by line model, all the software criteria for six sigma software project allocation are considered, and the very important software projects are identified. Then, the impact index, which is the outcome of the TOPSIS technique, and the effort index are computed.

Using the fuzzy logic expert system, the six sigma software projects are allocated to the relevant experts or champions. It can be claimed that two indexes, such as impact and also effort, are used for software projects allocation and the six sigma software potential projects to six sigma champions. To compute the impact index, the important software criteria such as resource availability, time requirement, strategic objectives relations, key processes relation in software organizations, profits, and special requirements to improve. In the case study, some of the objectives are very important. They are shown in table 2.

In six sigma software project selection and prioritization, some of the objectives are very important such as the direct relation with strategic objectives, profits, and time requirements. The effort index and the impact index are effective on six sigma software project allocation and prioritization.

Further work can concentrate on the evaluation of alpha and beta in the fuzzy logic expert system. Supporting structure can be used as a fundamental pattern for developing new models in other organizations. And very important criteria in the allocation and prioritization of six sigma projects can be mentioned in these organizations.

VI. REFERENCES

[1] handrakanth G Pujari, and Dr Seetharam K, An Evaluation of Effectiveness of the Software Projects Developed Through Six Sigma Methodology, American Journal of Mathematical and Management Sciences. 34(1) (2015-16).

[2] Buglione L, Trudel S, Guideline for sizing Agile projects with COSMIC, In Proceedings of the IWSM / MetriKon / Mensura , Stuttgart, Germanym. (2010)

[3] Bhasin S, Performance of organizations treating lean as an ideology, Business Process Management Journal. 17(6) 986-1011.

[4] Bergmiller G.G, and McCright P.R, Parallel models for lean and green operations, paper presented at Industrial Engineering Research Conference Miami, FL. (2009).

[5] Furlan A, Vinelli A, and Dal Pont G, Complementary and lean manufacturing bundles: an empirical analysis, International Journal of Operations & Production Management, 31(8) (2011) 835-850.

[6] George M, The Lean Six Sigma Guide to Doing More with Less cost, John Wiley & Sons, Hoboken, NJ. (2010).

[7] Stone K.B, Four decades of lean: a systematic literature review, International Journal of Lean Six Sigma. 3(2) (2012) 112-132.

[8] Vinodh S, and Balaji S, Fuzzy logic based Leanness assessment and its decision support system, International Journal of Production Research. 49 (2011) 40-67.

[9] Zhang Q, bbas J, Zhu X, and Shah M, Critical success factors for successful Lean Six Sigma implementation in Pakistan, Interdisciplinary Journal of Contemporary Research in Business. 4(1) (2012) 117-124.

[10] S.M. Kazemi, Six Sigma project selection by using a multiple fuzzy criteria decision-making approach a case study in poly acryl corp. CIE42 proceedings, Cape Town, South Africa 2012 CIE & SAIIE. (2012) 16-18.

[11] Snee, Ronald: Leading Six Sigma: A Step-by-Step Guide Based on Experience with GE and Other Six Sigma Companies, Financial Times Prentice Hall. (2002).

[12] Stone K.B, Four decades of lean: a systematic literature review, International Journal of Lean Six Sigma. 3(2) (2012) 112-132.

[13] Vinodh S, and Balaji S, Fuzzy logic based Leanness assessment and its decision support system, International Journal of Production Research. 49 (2011) 40-67.

[14] Furlan A, Vinelli A, and Dal Pont G, Complementary and lean manufacturing bundles: an empirical analysis, International Journal of Operations & Production Management. 31(8) (2011) 835-850.

[15] Maroofi F, and Dehghan S, Performing lean manufacturing system in small and medium enterprises, International Journal of Academic Research in Accounting, Finance and Management Sciences. 2(3) (2012) 156-163.

[16] Nordin N, Deros B.M, and AbdWahab D, A framework for managing change in lean manufacturing implementation, International Journal of Services and Operations Management. 12(1) (2012) 101-117.

[17] Stone K.B, Four decades of lean: a systematic literature review, International Journal of Lean Six Sigma. 3(2) 112-132.

[18] Chandrakanth G Pujari, and Dr Seetharam K, Top Priority of Software Success Factor for Six Sigma Execution by a Fuzzy Hierarchical Process, International Journal of Multimedia and Ubiquitous Engineering. 9(11) (2014-15) 171-180.

[19] Chandrakanth G Pujari and Dr Seetharam K, Evaluation for Defective Density in All the Right Places, Indian journal of engineering. 11(26) (2014) 30-37.

[20] Chandrakanth G Pujari, and Dr Seetharam K, Ranking of Tools use, logical software complexity, Requirement volatility, Quality requirements, Efficiency Requirements in software development, 2009 IEEE International Advance Computing Conference. (2009).

[21] Chandrakanth G Pujari, Software Information Flexibility for Lean Six Sigma Software Development Using Multiple Regression Analysis, Indian Journal of Engineering. 14(36) (2017) 95-107

[22] Chandrakanth G pujari, Dr SeetharamK, Investigating the Effects of Factors on Software Development, International Journal of Computer Applications, 1(6) (2010) 56-65.

[23] Chandrakanth G Pujari and Dr. SeetharamK.,Article: Estimation of Growth Parameters for a Software Development. International Journal of Computer Applications 35(12) (2011) 38-42.

[24] Chandrakanth G Pujari, Kavyashree N, Dr Supriya M C, Enhancement of Indian Software Quality Management Using Multi-Criteria Objects and Six Sigma Methodology, International Journal on Future Revolution in Computer Science & Communication Engineering. 4(4) (2019) 806 – 81.

[25] Chandrakanth G Pujari, Dr SeetharamK, Detection and valuation of major error trends of software projects using Pareto principle and fuzzy model, National journal on advances in computing & Management, 3(2) (2012).

[26] Chandrakanth G Pujari, Kavyashree N, Dr Supriya M C, Development of a Methodology for Software Small and Medium Scale Industries in the Selection of Suitable Lean Six Sigma Tools, JASC: Journal of Applied Science and Computations. 6(2) (2019).

[27] Chandrakanth G Pujari and Dr Seetharam K, Software Defects Identification Using Principles of Data Gathering and Pareto Analysis, (2012).

[28] Chandrakanth G Pujari and Dr Seetharam K, Software Defects Identification Using Principles of Data Gathering and Pareto Analysis. (2012).

[29] Chandrakanth G Pujari, Modeling Software Project Defects With Fuzzy Logic Maps, International Journal on Future Revolution in Computer Science & Communication Engineering IJFRCSCE. 4(4) (2018) 103 – 107.

[30] Chandrakanth G Pujari and Dr. Seetharam K, Implementation of Multivariate Clustering Methods for Software Development. [Online]. Available: https://www.bvicam.ac.in