Prediction of Software Project Effort Estimation: A Case Study

MOHD. SADIQ[1], MOHAMMAD ASIM[2] , JAVED AHMED[1], VINISH KUMAR[3], SHADAB KHAN [3]

Abstract—**It is well known fact that predicting software effort for the software development projects with any acceptable degree remains challenging. In this paper we have used the organic software projects because in each case the projects size lies between 2-50 KLOC. In this paper we have applied the linear regression model i.e. Effort = -1.5 + 0.1804 FP to predict the software project effort and function point, and on the basis of the fuzzy logic we have also predicted the software project effort. After obtaining the software effort, project manager can arrange the project progress, control the cost and ensures the quality more accurately.**

Indexed Terms—**Software Effort, Function Point, MRE, Fuzzy Function Point, Membership Function, Mean Relative Error.**

I. Introduction

Software development effort estimation is a branch of forecasting that has received increased interest in academia, application domains and media. Efficient development of software requires accurate estimates. Unfortunately, software development effort estimates are notorious for being too optimistic. Inaccurate software estimates causes trouble in business processes related to software development such as project feasibility analyses, budgeting and planning.

It is unrealistic to expect very accurate effort estimates of software development effort because of the inherent uncertainty in software development projects, and the complex and dynamic interaction of factors that impact software development effort use. Still, it is likely that estimates can be improved because software development effort estimates are systematically overoptimistic and very inconsistent. Even small improvements will be valuable because of the large scale of software development.

Software researchers have addressed the problems of effort estimation for software development projects since at least the 1960s. Research is found in areas such as [30]:

1) Creation and evaluation of estimation methods. Describes work on the creation and evaluation of estimation methods, such as methods based on expert judgment, structured group processes, regression-based models, simulations and neural networks.
2) Calibration of estimation models. Tailoring a model to a particular context (calibration) has been found to be difficult in practice. Problematic issues are related to, among others, when, how and how much local calibration of the models that are beneficial.
3) Software system size measures. The main input to estimation models is the size of the software to be developed. It has been proposed many size measures, for example based on the amount of functionality that is described in the requirement specification.
4) Uncertainty assessments. Software developers are typically over-confident in the accuracy of their effort estimates. Realistic uncertainty assessments are important in order to enable proper software project budgets and plans.
5) Measurement and analysis of estimation error. Proper accuracy measurement is essential when evaluating estimation methods, and identifying causes of estimation error.
6) Organizational issues related to estimation. Organizational issues such as processes to control the cost and scope of the project may have a large impact on estimation accuracy.
7) Measuring and analyzing estimation error are the basis of estimation learning related activities, such as deciding whether or not an organization has an estimation problem, identifying risk factors related to project performance in software development, and, evaluating and improving estimation and uncertainty assessment methods and tools.

The most commonly used measure of estimation error is the Magnitude of Relative Error (MRE). The mean MRE (MMRE) is often used to average estimation error for multiple observations. It is not unproblematic to use MMRE as a measure of estimation accuracy, and several other measures, such as PRED and MER, is sometimes used. However, all estimation error measures have shortcomings. Hence, the measure that should be used in any given case depends on the context.

The rest of the paper is organized as follows: In section 2 we have explained all the background and related work that are based on the prediction of the effort from the linear regression model and also from other techniques. Brief description about the software project effort estimation and function point analysis are given in section 3, and section 4

Manuscript received on March 20, 2011

First Authors are with Computer Engineering Laboratory, Section of Computer Engineering, University Polytechnic, Faculty of Engineering and Technology, Jamia Millia Islamia (A Central University), New Delhi-110025, India. E-mail: sadiq.jmi@gmail.com

Second Author is the M.Tech. Scholar, Department of Computer Science and Engineering, AL-Falah School of Engineering and Technology, Dhauj, Faridabad, affiliated to Maharshi Dayanand University, Rohtak, Haryana, India.

Third Author are working with the Department of Computer Science and Information Technology, Sunder Deep College of Engineering and Technology, Dasna, Ghaziabad, U.P., India, affiliated to U.P. Technical University, Lucknow, U.P, India

respectively. Section 5 contains the case study and experimental work to predict the software cost without using fuzzy logic and section 6 contains the fuzzy function point analysis and its experimental work is carried out in section 7 and finally we conclude the paper in section 8.

II. Background and Related Work

In [1] the authors have developed a tool that are based on software engineering matrices, this tool is used to evaluate the function point of software. In [2] the authors conduct a set of machine learning experiments with software cost estimation data from two separate organizations. In this paper the first data set consists of 104 vectors and second data set consists of 434 vectors extracted from a Electronics Commerce Software and Fleet Management Software respectively. In [3], Ingunn Myrtveit et al. have worked on reliability and validity of Software Prediction Models. In this paper the authors have used 3 different measuring procedure i.e. reliability and validity; cross-validation; and accuracy indicators. These papers have used Finnish data set in order to generate the linear regression model and also log-linear regression model. In the simulation study of this paper that is based on Finnish data set and the author opted the log-linear model. In [4], Iman Attarzedh et al. proposed a New Software Cost Estimation Model based on Artificial Neural network. In [5] the effort estimation is evaluated on the basis of genetic algorithm. There are so many approaches to estimate the effort like Machine learning approach, genetic algorithm, ANN, and algorithmic approaches [6, 7, 14]. In [11, 12] the authors have developed a tool to estimate the software risk and cost. Function point is very popular software metric to calculate the effort, productivity, cost, etc. In [15, 16], the authors have worked on software function, source lines of code, and prediction of development effort. In [19], Ho-Leung has explained the function point evaluation using case studies. Some applications of the function point are available in [11,12,21,22]. These applications are based on conventional function point rather than fuzzy function point. In [24] Osias de Souza Lima Junior et al. have worked on fuzzy model for function analysis for the enhancement project assessment and in this paper authors have worked on trapezoidal fuzzy numbers . Several papers have been published in this area. So in the continuation of the earlier work we have used the fuzzy approach using triangular fuzzy number to predict the software effort.

III. Software Project Effort Estimation

In this paper we have included function points as an algorithmic method since they are dimensionless and therefore need to be calibrated in order to estimate the error. In the software Engineering literature there are so many models that are used to estimate the effort. Some of the important estimation techniques are [05]

1. SEL –Model
2. Walston- Felix Model
3. COCOMO basic Model
4. COCOMO Intermediate Model
5. Intermediate Organic Model

In this paper we have used the organic mode of COCOMO basic model. The general form of the Effort can be written as

$$E= a\ LOC^{b,} \qquad (1)$$

where E is the effort, LOC is the size typically measured in thousand lines of code or function points, a is the productivity parameter and b is an economic or diseconomies of scale parameter. Apart from this approach we have another technique that is based on algorithmic approach which is used to calibrate a model by estimating values for the parameters. The most straightforward method is to assume a linear model. Using regression analysis the model can be represented as:

$$E = a1 + a2\ S \qquad (2)$$

where a1 represents fixed development cost and a2 represents productivity [19].

Effort estimation predicts how many hours of work and how many workers are needed to develop a project? The effort invested in a software project is probably one of the most important and most analyzed variables in recent years in the process of project management. The determination of the value of this variable when initiating software projects allows us to plan adequately any forthcoming activities. As far as estimation and prediction is concerned there is still a number of unsolved problems and errors. To obtain good results it is essential to take into consideration any previous projects.

Estimating the effort with a high grade of reliability is a problem which has not yet been solved and even the project manager has to deal with it since the beginning. Several methods have been used to analyze data, but the reference technique has always been the classic regression method. Therefore, it becomes necessary to use some other techniques that search in the space of non linear relationship. Some works in the field have built up models (through equations) according to the size, which is the factor that affects the cost (effort) of the project. The equation that relates size and effort can be adjusted due to different environmental factors such as productivity, tools, complexity of the product and other ones. The equations are usually adjusted by the analyst to fit the real data.

IV. Function Point Analysis

Function points were defined in 1979 in A New Way of Looking at Tools by Allan Albrecht at IBM. The functional user requirements of the software are identified and each one is categorized into one of five types: outputs, inquiries, inputs, internal files, and external interfaces. Once the function is identified and categorized into a type, it is then assessed for complexity and assigned a number of function points. Each of these functional user requirements maps to an end-user business function, such as a data entry for an Input or a user query for an Inquiry. This distinction is important because it tends to make the functions measured in function points map easily into user-oriented requirements, but it also tends to hide internal functions (e.g. algorithms), which also require resources to implement. Over the years there have been different approaches proposed to deal with this perceived weakness; however there is no ISO recognized FSM Method that includes algorithmic complexity in the sizing result.

In this paper we have not defined the complexities of the projects i.e. low, average and high. These three parameters are generally used to calculate the unadjusted function point (UFP). To find out the values of the UFP five different functional units are required. In the FP literature the functional units contains external inputs, external outputs, external inquiries, internal logical files and external interface files. Finally the FP is calculated using the following relationship.

$$FP = UFP * VAF \qquad (3)$$

Whereas the VAF i.e. value adjustment factor is calculated using 14 general system characteristics. To get the detailed description about the 14 general system characteristics, readers are advised please refer to [13]. The detailed description about the function point analysis and its computation is available in [03,15,20].

To find out the values of the FP, we have used the relationship between LOC and FP given in [13].

V. Case Study And Experimental Work

Generation of artificial data with known properties to generate the software engineering data set modeling technique was first proposed by Pickard et al. [06]. It provides the researchers with a great deal and more control over the characteristics of a data set. With the help of the result of [2,30] in this paper we have considered 10 different projects of organic mode and the size of the projects lies between 2-50 KLOC. The information about the Lines of code of each project is available in table-1.

TABLE-01

Project No.	Lines of code
1.	14000
2.	12000
3.	09000
4.	06000
5.	08000
6.	05000
7.	10000
8.	15000
9.	11000
10.	13000

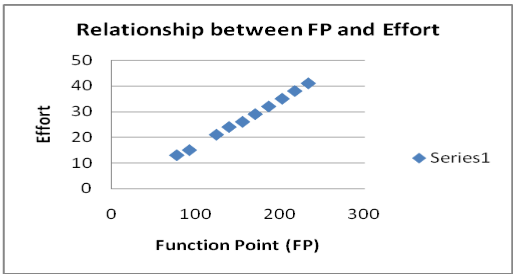
Using the results of [06, 13] we can roughly estimate the value of the function point. In this paper we have considered that all the projects are written in C++, so in this case the value of LOC/FP=64. We have applied the basic COCOMO to estimate the effort for each project. Effort would be calculated with the help of the following equation:

$$E = a\,(KLOC)^b \qquad (4)$$

In case of the COCOMO organic mode the values of a and b would be 2.4 and 1.05 respectively. We have tabulated the results of FP and efforts after applying the results of [06] and equation-1 in table-2

TABLE-02

Project No.	FP	Effort
1.	218	38
2.	187	32
3.	140	24
4.	93	15
5.	125	21
6.	78	13
7.	156	26
8.	234	41
9.	171	29
10.	203	35



Graph-01

The relationship between the variable and dependent variable tends to be straight line and the relationship between the FP and the Effort is given in graph-01, which has a highly positive correlation. Therefore a linear regression model Y= a +bX can be assumed to represent the relationship between software effort and function point. From the data set given in table-2, we have found following linear relationship between effort and FP:

$$Effort = -1.5 + 0.1804\,FP. \qquad (5)$$

For evaluating the different software effort estimation model, the most widely accepted evaluation criteria are the mean magnitude of relative error (MMRE) and the probability of a project having a relative error of less than or equal to 0.25.The magnitude of relative error (MRE) is defined as follows:

$$MRE_i = \text{absolute}\,(\text{Actual Effort}_i - \text{Predicted Effort}_i) / \text{ActualEffort}_i \qquad (6)$$

The MRE value would be calculated for each observation i whose effort is predicted. The summation of MRE over multiple observations (N) can be achieved through the Mean MRE (MMRE) as follows:

$$MMRE = 1/N \sum_{i}^{N} MREi \qquad (7)$$

Finally we have tabulated the results of actual effort, predicted effort and MRE in table-03

TABLE-03

Project No.	Actual Effort	Predicted Effort	MRE_i MMRE= 0.1356
1.	38	37.83	0.0192
2.	32	32.23	$7.187 X10^{-3}$
3.	24	23.756	0.0101
4.	15	15.2772	0.0184
5.	21	21.05	$2.380 X 10^{-3}$
6.	13	12.57	0.03307
7.	26	26.6	0.02307
8.	41	40.71	$7.073 X 10^{-3}$
9.	29	29.34	0.0117
10.	35	35.1212	$3.462 X 10^{-3}$

VI. Fuzzy Function Point Analysis (FFPA)

In this section we have applied the fuzzy logic in order to compute the function point. Triangular fuzzy numbers can be represented as TFN (α, m, β). The membership function for the triangular fuzzy numbers can be represented as:

$$\mu(x) = \begin{Bmatrix} 0; \; x \leq \alpha \\ x-\alpha/m-\alpha \, ; \; \alpha \leq x \leq m \\ \beta - x/^{\beta - m} \, ; m \leq x \leq \beta \\ 0; x \geq \beta \end{Bmatrix} \quad (8)$$

In the case of triangular fuzzy number (as shown below) the value of α = 50, β=60 and m=55. Following figure shows the representation of the Triangular fuzzy numbers, Trapezoidal Fuzzy numbers and Bell Shaped Fuzzy numbers.

In this paper we have estimated the value of the FP for the new linguistic variable, ***Very High,*** *using* Newton's Gregory Forward Interpolation Formula. It can be defined as follows [26]:

If y_0, y_1, y_2,---------y_n are the values of y=f (x) corresponding to equidistant values of x=x_0, x_1, x_2,-------x_n, where $x_i - x_{i-1}$ = h, for i= 1,2,3,4,----,n, then $y= y_0 + (u/1!) \Delta y_0 + \{ u (u-1)/ 2!\} \Delta^2 y_0 +$ ---------------------, where $u= (x-x_0)/h$.

After applying the Newton's Gregory Forward Interpolation Formula on the contents of table -01, we have computed the new values for the linguistics variable very high, and we have tabulated the results in table-04

TABLE-04

Level	Function Points				
	ILF	**EIF**	**EI**	**EO**	**EQ**
Low	7	5	3	4	3
Average	10	7	4	5	4
High	15	10	5	7	6
Very High	22	14	9	10	9

FFPA consists of the following 4 stages **[24]**

1) Fuzzification of the linguistic terms of FPA complexity matrices by generating fuzzy numbers.
2) Extended FPA complexity matrices by generating new linguistics terms.
3) Determine FP values for the new linguistics terms
4) Defuzzification of the linguistics terms of FFPA function point values.

In this paper we have considered the triangular fuzzy numbers i.e. TFN (α, m, β) and implemented the above steps in C language and summarized the results in the following tables.

Calculation for First Project

TABLE-05 Calculation of FFP and FP for EIF

DET	RET	µ	Count	UFFP	UFP
80	3	0.5	2	12*2=24	20
42	5	0.1	1	9.7*1=9.7	7
24	3	0.31	4	5.62*4=22.48	28
30	4	0.68	4	6.4*4=25.6	28
Total				**81.78**	**83**

TABLE-06 Calculation of FFP and FP for EO

DET	FTR	µ	Count	UFFP	UFP
18	2	0.143	2	6*2=13.428	10
11	2	0.89	2	4.25*2=9.7	10
04	1	0.4	2	1.6*2=3.2	8
Total				**26.328**	**28**

TABLE-07Calculation of FFP and FP for EQ

DET	FTR	µ	Count	UFFP	UFP
18	1	0.11	2	3.01*2=6.02	6
5	1	0.26	3	0.789*3=2.37	9
8	2	0.21	2	3.21*2=6.42	10
11	5	0.33	2	8.01*2=16.02	12
Total				**30.83**	**37**

TABLE-08 Calculation of FFP and FP for EI

DET	FTR	µ	Count	UFFP	UFP
2	1	0.25	2	0.75*2=1.5	6
3	2	0.5	2	3.5*2=7.0	6
1	3	0.5	3	2*3=6	12
8	2	0.67	3	3.67*3=11.01	12
Total				**25.51**	**36**

Comparison of Unadjusted FP using Conventional and Fuzzy Technique is available in Table-10.

TABLE-09

	UFFP	**UFP**
ILF	0	0
EIF	81.78	83
EO	26.328	28
EQ	30.83	37
EI	25.51	36
Total	164.448	184

In order to compute the value of the function point and fuzzy function point, we have assumed that the value of the 14 general system characteristics is 50.The actual values of the function point and the fuzzy function point is available in table-11.

TABLE-10
For Project-01

	Unadjusted FP/FFP	VAF	Total UFP or UFFP*VAF
FP	184	1.65	303.6
FFP	164.448	1.65	271.3392

In the similar fashion we have computed the values of the FFP and FP of 5 different projects and we have summarized the results of FFP and FP of these 5 different projects in table-11.

TABLE-11

Project No	FFP	FP
1.	271	303
2.	288	320
3.	328	360
4.	376	408
5.	266	298

VII. Case Study and Experimental Work

The information about the Lines of code of each project using function point and fuzzy function point are available in table-12 and 13.

TABLE-12

Project No.	FP	LOC	KLOC
11.	303	19392	19.392
12.	320	20480	20.48
13.	360	23040	23.04
14.	408	26112	26.112
15.	298	19072	19.072

TABLE-13

Project No.	FFP	LOC	KLOC
1.	271	17344	17.344
2.	288	18432	18.432
3.	328	20992	20.992
4.	376	24064	24.064
5.	266	17024	17.024

Using the results of [20] we can roughly estimate the value of the Function Point. In this paper we have considered C++ projects, so in this case the value of LOC/FP=64. We have applied the basic COCOMO to estimate the effort for each project. Effort would be calculated with the help of the following equation:

$$E= a\ (KLOC)^{b} \qquad (9)$$

In case of the COCOMO organic mode the values of a and b would be 2.4 and 1.05 respectively. We have tabulated the results of (FP, Effort) and (FFP, Effort) after applying the results of [13] and equation-09 on the contents of table-12 and 13 respectively.

TABLE-14

Project No.	FP	Effort
11.	303	53.98
12.	320	57.162
13.	360	64
14.	408	73
15.	298	53.043

TABLE-15

Project No.	FFP	Effort
1.	271	48
2.	288	51
3.	328	58.67
4.	376	67.71
5.	266	47

A linear regression model Y= a +bX can be assumed to represent the relationship between software effort and function point. From the data set given in table-14 and table-15, we have got the following linear relationship between (FP, effort) and (FFP, effort) as shown in equation 10 and equation 11 respectively:

Effort = -2 .493+0.185 FP (10)

Effort = -1.15 +0.181 FFP (11)

For evaluating the different software effort estimation model, the most widely accepted evaluation criteria are the mean magnitude of relative error (MMRE) and the probability of a project having a relative error of less than or equal to 0.25.The magnitude of relative error (MRE) is defined as follows:

$$MRE_i = \text{absolute}\ (\text{Actual Effort}_i - \text{Predicted Effort}_i) / \text{Actual Effort}_i \qquad (12)$$

The MRE value would be calculated for each observation i whose effort is predicted. The summation of MRE over multiple observations (N) can be achieved through the Mean MRE (MMRE) as follows:

$$MMRE = 1/N \sum_{i}^{N} MRE_i \qquad (13)$$

VIII. Results and Conclusion:

In this paper we have predicted the software project effort using linear regression model. On the basis of the data set values we have computed the following regression model.

Effort = -1.5 + 0.1804 FP

On the basis of the resultant regression model we can compute the following:

1) 0.1804 means that it will cost 0.1804 man-day to finish one function point [14]. When estimating the software effort, firstly we must know the count of function point, and then calculate the project effort according to equation (5). Enterprise can establish their own linear model by using their records. As we know that it is difficult to figure out the count of function point and it will greatly simplify the process of software estimation.

In our case study the value of the MMRE is found to be 0.1356. In this paper we have predicted the software project effort using linear regression model. On the basis of the data set values we have got the following FP regression models using FP and FFP.

Effort = -2 .493+0.185 FP

Effort = -1.15 +0.181 FFP

On the basis of the resultant regression model we can compute the following:

2) 0.185 means that it will cost 0.185 man-day to finish one function point. Enterprise can establish their own linear model by using their records. As we know that it is difficult to figure out the count of function point and it will greatly simplify the process of software estimation. In case of the FFP it will cost 0.181 man day to finish one function point.

3) In the future we will predict the values of the function point from the effort and in this case we will get the following linear regression model:

FP = a + b Effort.

REFERENCES

[1] Daya Gupta, SatyaPal Jee Kaushal, Mohd. Sadiq, "Software Estimation tool Based on Software Engineering Metrics Model", IEEE International Conference on Management of Innovation & Technology, Bangkok, Thailand. Pp.623-628

[2] Gary D. Boetticher, "Using Machine Learning to Predict Project Effort: Empirical Case Studies in Data –starved Domains",

[3] Ingunn Myrtveit, Eric Stensrud, Martin Shepperd, "Reliablity and Validity in Comparative Studies of Softwarte Prediction Models", IEEE Transaction on Software Engineering, Vol. 31, No. 5, pp. 380-391, May. 2005.

[4] Iman Attarzadeh, Siew Hock Ow, "Proposing a New Software Cost Estimation Model Based on Artificial Neural Network", Proceedings of the 2nd International Conference on Computer Engineering and Technology (ICCET 2010), 978-1-4244-6349 2010 IEEE, Vol. 3, pp. 487-491

[5] Kavita Choudhary, "GA Based Optimization of Software Development Effort Estimation" International Journal Of Computer Science And Technology Sept'10

[6] L. Pickard, B. Kitchenham, and S. linkman, " An Investigation Analysis techniques for Software Datasets, "Proceeding of Sixth IEEE International Software Metrics, Symposium, 1999.

[7] Martin Shepperd and Chris Schofield, "Estimating Software Project Effort Using Analogies", IEEE Transaction on Software Engineering, Vol. 23, No. 12, pp. 736-743, Nov. 1997

[8] Martin Shepperd and CartwrightEmpirical, "Predicting with Sparse Data", IEEE Transactions on Software Engineering , Vol. 27, pp. 987-998, Nov 2001

[9] Mohd. Sadiq, and Shabbir Ahmed, " Computation of Function Point of Software on the basis of Average Complexity", Proceedings of 2nd International Conference on Advanced Computing and Computing Technologies, ICACCT2007, Panipat, Haryana. Pp.591-594.

[10] Mohd.Sadiq, Shabbir Ahmed, "Relationship between Lines of Code and Function Point and its Application in the Computation of Effort and Duration of a Software using Software Equation", International Conference on Emerging Technologies and Applications in Engineering, Technology and Sciences , ICATETS2008, Rajkot, Gujarat, India

[11] Mohd. Sadiq, Abdul Rahman, Shabbir Ahmad, Mohammad Asim, Javed Ahmad", esrcTool: A Tool to Estimate the Software Risk and Cost", IEEE Second International Conference on Computer Research and Development,pp. 886-890,DOI 10.1109/ICCRD.2010.29

[12] Mohd. Sadiq, Sunil, Sherin Zafar, Mohammad Asim, R. Suman, "GUI of esrcTool: A Tool to Estimate the Software Risk and Cost", The 2nd IEEE International Conference on Computer and Automation Engineering (ICCAE-2010), Singapore, pp 673-677.

[13] Pressman, Roger S., "Software Engineering", A practitioner's Approach V Edition, pp94.

[14] Yinhuan Zheng, Beizhan Wang, Yilong Zheng, Liang Shi, "Estimation of Software Project effort based on functional point" Proceeding on 2009 4th International Conference on Computer Science & Education, 978-1-4244-3521 2009 IEEE, pp. 941-943.

[15] Albrecht A.J. and Gaffney J.E. 1983, Software Function, Source Lines of Code and development Effort Prediction: Software Science Validation, IEEE Trans. Software Engineering, Pp.639-648.

[16] Albrecht A.J. 1979, Measuring Application Development Productivity, Proc. IBM Applications Development Symp., Monterey, Calif. ,Pp.14-17.

[17] Boehm B.W. 1981: Software Engineering Economics, Prentice Hall.

[18] C.J. Lokan, 1999. An Emprical Study of the Correlations between Function Point Elements, Proc of 6-th International Symposium on Software Metrics.

[19] Ho-Leung, TSOI, 2005, "To Evaluate the Function Point Analysis: A Case Study" , International Journal of the Computer , the Internet and Management Vol.13#1, Pp 31-40.

[20] International Function Point User Group (IFPUG), Function Point Counting Practices Manual, Release 4.0, IFPUG, Westerville, Ohio, April 1990.

[21] Kemerer C.F.1987, An Emprical Validation of Software Cost Estimation Models, Comm.ACM, Vol.30, no.5.

[22] Low G.C. and Jeffery D.R.1990, Function Point in the Estimation and Evaluation of the Software Process, IEEE Trans Software Engineering, Vol.16, no.1.

[23] Mohd. Sadiq, Aleem Ali, Megha Gupta, Shavi Gupta, Shadab Khan, " Fuzzy Function point Analysis using Triangular Fuzzy Numbers", 4th IEEE International Conference on Software Testing , Verification, Validation, Berlin. (Communicated)

[24] Osias de Souza Lima Junior, Pedro Porfirio Muniz Parias, Arnaldo Dias Belchior, " A Fuzzy Model for Function Point Analysis to Development and Enhancement Project Assessment", CLEI Electronic journal 5(2), 1999.

[25] Symons C.R., 1988 "Function Point Analysis: Difficulties and Improvements, IEEE Trans. Software Engineering, Vol 14, no1.

[26] T. Veeranjan and T. Ramachandran, "Numerical Methods with programs in C", tata McGraw Hill.

[27] Walston C.E. and Felix, C.P. 1977: A method of program measurement and estimation, IBM System Journal, Vol.16,

[28] Yinhuan Zheng, Beizhan Wang, Yilong Zheng, Liang Shi, "Estimation of Software Project effort based on functional point" Proceeding on 2009 4th International Conference on Computer Science & Education, 978-1-4244-3521 2009 IEEE, pp. 941-943.

[29] Zuse H. 1991., Software Metrics-Methods to Investigate and Evaluate Software Complexity Measures, Proc. Second Annual Oregon Workshop on Software Metrics, Portland.

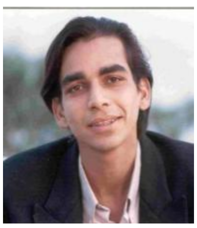
[30] Mohd. Sadiq, Aleem Ali, Syed Uvaid Ullah, Shadab Khan, Qamar Alam, "Prediction of Software Project Effort using Linear Regression Model", *2011 3rd IEEE International Conference on Computer and Automation Engineering (ICCAE 2011), pp 433-437.*

Mohd. Sadiq was born at Bakewar; district Etawah, U.P. India, on 20 March, 1980. Currently he is pursuing Ph.D. in Computer Engineering from National Institute of Technology, Kurukshetra, Haryana. He did Master of Technology in Computer Science and Engineering with specialization in Software Engineering from Aligarh Muslim University (AMU), Aligarh, U.P., India, in 2005 and Bachelor of Engineering in Computer Science and Engineering from Al-Falah School of Engineering and Technology, Dhauj, Faridabad, Haryana, affiliated to Maharishi Dayanand University, Rohtak, Haryana, in 2001with first position in second year. He has more than 5 years of teaching experience and currently he is working as an Assistant Professor of Computer Engineering in Section of Computer Engineering, University Polytechnic, Faculty of Engineering and Technology, Jamia Millia Islamia (A Central University), New Delhi, India. He has published more than 25 research papers in international and national journals like IACSIT International Journal of Engineering and Technology,
International Journal of Recent Trends in Engineering, Finland and International Journal of Futuristic Computer Application, Duke University, USA; and in international and national conferences like IEEE international conferences at Singapore, Thailand, Kerala (India), and Bangalore (India). His research interest includes Software Engineering, Data Structure and Algorithms.

Mr. Sadiq is a member of International Association of Engineers (IAENG) England, International Association of Computer Science and Information Technology (IACSIT), Singapore and member of Computer Science Teachers Association (CSTA), USA.

Mr. Sadiq is a member of the Editorial Review Board of Journal for Computing Teachers (JCT), Buffalo State College, New York, U.S.A., and also the Member of the Editorial Review Board of Journal of Internet and Information System. Victoria Island Lagos, Nigeria. He is the Reviewer of the International Journal of Computer Science and Technology and International Journal of Electronics and Communication Technology. He is also the **Regional Coordinator** of International Journal of Emerging technologies and Applications in Engineering technology and Sciences and International journal of Computer Applications in Engineering, Technology and Sciences.

Mr. Sadiq has also chaired the session of IEEE International Conference on Advances in Recent Technologies in Communication and Computing, 2009, Kerala, India. He is also the member of Technical Program Committee of Springer and ACM International Conferences on Advances in Computing and Communication (ACC-2011) , Kochi, Kerala, India.



Mohammad Asim was born at New Delhi India, on 30 March, 1975. Currently he is pursuing M. Tech in Computer Science & Engineering from Al-Falah School of Engineering and Technology, Dhauj, Faridabad, Haryana, affiliated to Maharishi Dayanand University, Rohtak, Haryana. He did Post Graduate Diploma in Computer Application from Jamia Millia Islamia,New Delhi India, in 1999 and B.Sc

Engineering in Civil Engineering from Jamia Millia Islamia,New Delhi India,, in 1996. He has more than 13 years of experience and currently he is working as an Assistant Professor in Al-Falah School of Engineering and Technology, Dhauj, Faridabad, Haryana, affiliated to Maharshi Dayanand University, Rohtak, Haryana,, India.

He has published more than 5 research papers in International and national conferences like IEEE international conferences at Singapore, Malaysia , and Bangalore (India). His research interest includes Software Engineering, Data Structure and Algorithms.

Mr. Asim is a member of International Association of Engineers (IAENG) England, International Association of Computer Science and Information Technology (IACSIT), Singapore and member of Computer Science Teachers Association (CSTA), USA. He has attended many conferences, workshops & seminars. He won various Chess championships during his studies in 1992, 1996, 1999, 2009 and 2011.

Javed Ahmad was born at Sultanpur on December 23, 1985. He did B.Tech. in Computer Science and Engineering from Shobhit Institute of Engg. & Technology, Meerut, affiliated to U.P. Technical University, Lucknow, U. P. India in 2008.Currently he is pursuing M. Tech. (Part Time) in Computer Science from Jamia Hamdard (Hamdard University), New Delhi, India.

He is working as Guest Lecturer in Computer Engineering in Section of Computer Engineering, University Polytechnic, Faculty of Engineering and Technology, Jamia Millia Islamia (A Central University), New Delhi-110025

Vinish Kumar was born at Ghaziabad district of U.P. India, on 14th October, 1980. Currently he is pursuing Ph.D. in Computer Science from Singhania University. He did Master of Technology in Information Technology from Allahabad Agricultural Institute, A Deemed University, Allahabad , U.P., India, in 2007 and Bachelor of Engineering in Computer Science and Engineering from S.D. College of Engineering and Technology Engineering ,MuzaffarNagar, U.P. India in 2001.He has more than 10 years of teaching experience and currently he is working as an Associate Professor of Computer Engineering in department of Computer Engineering, at SunderDeep College of Engineering and Technology (Affiliated from U.P. Technical .University ,Lucknow) Ghaziabad ,U.P. India. He has communicated 3 research papers in international journals and 4 research papers in national journals and conferences.

Mr.Vinish is a Life member of The Indian Society for Technical Education (ISTE), and Computer Society Of India (CSI), International Association of Computer Science and Information Technology (IACSIT), Singapore. Mr. Vinish has also chaired the session of National Conference on Emerging trends in Embedded Technology 2009(SGIT-ETET-2009), Ghaziabad; India. Ha has also attended various Technical Workshops and actively participated in Faculty Development Programs.

Shadab khan was born at Ghaziabad district of U.P. India on 12th December 1980.Currently he is pursuing M.Tech in Computer Science from JSS Academy of Technical Education, Noida affiliated to U.P Technical University. He did Bachelor of Engineering from Al-Falah School of Engineering and Technology in 2004. He has more than 4 years of teaching experience and currently he is working as a Senior Lecturer of Computer Engineering in department of computer Engineering at Sunder Deep College of Engineering and Technology (Affiliated to U.P Technical University, Lucknow) Ghaziabad, U.P. India. He has communicated 2 research papers in international conferences. He also attended various Technical Workshops and actively participated in Faculty Development Programs.