Indian Journal of Science and Technology, *Vol 9(33), DOI: 10.17485/ijst/2016/v9i33/87656, September 2016*
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Estimating the size of E-Learning System using Learning Object Points Method

Shiny Angel[1*] and Paul Rodrigues[2]

[1]SRM University, Kattankulathur, Chennai - 603203, Tamil Nadu, India; shinyangeldavid@gmail.com
[2]DMI College of Engineering, Chennai – 600123, Tamil Nadu, India; paul_rodrigues@yahoo.com

Abstract

Background/Objectives: Software size estimation is the key factor to determine the planning activities of software development process. Size is the base factor to determine effort, duration, schedule, cost and others that affect the development process. E-Learning system is also a software system support for Computer and internet based teaching and learning process. Development of E-Learning system is under crises because of improper estimates that lead incompleteness, loss and delay, it affects customer satisfaction. To overcome the problem, a research made on the sizing techniques used in industry and the inabilities are identified. **Methods/Statistical Analysis:** Based on the analysis of all industry oriented size estimation techniques, this research introduces a new sizing technique called Learning Object Point method (LOP). Learning Object Points method is quantifying the size and complexity of an E-Learning system in terms of learning objects and functionalities. Sizing is independent of computer languages, development methodology and technology behind the development. LOP can be estimated early stage of software development process. It is prepared based on the user perspective so users of the E-Learning system have a better understanding of what LOP are measuring. **Findings:** The performance analysis of Learning Object Point's method was conducted over Function point Analysis by using different projects developed in the industry. Size and duration calculated using FPA produced wrong results. So the project management activities like planning, scheduling and costing produced imprecise outcomes but LOP produced more close to actual results so it supports project management activities effectively.

Applications/Improvements:

- LOP can be used to size E-Learning applications accurately. Sizing is important component in determining productivity.
- It is easily understood by the non-technical user. This helps communicate sizing information to a user or customer.
- Conversion to LOC is similar to FP to LOC conversion.
- It also supports to estimate any kind software application other than E-Learning system also.
- Estimate development effort and Cost benefit analysis using LOP.
- To Derive Business Decisions.

Keywords: E-Learning System, Learning Object Points Method, Size Estimation, Software Project Management, Software Sizing

1. Introduction

Software industries need an effective management of the software process to construct software's with in time, cost and quality. For that essential planning and estimation activities are important[3–12,14–30] for E-Learning development also have no exception. E-Learning system is a software package Supporting for teaching and learning process by the use of computer and internet facilities that consists of various components like learning content documents which support for learning, which includes text, image, animation, simulation, video and audio based information's, Fund transfer facilities,

*Author for correspondence

assessment system, Database management system, Forum for discussion, etc[1,2]. So numerous files and objects involved in the development of E-Learning system[13]. The major files involved are files having executable instructions; files having executable scripts, Time based multimedia files including video, audio, animation and simulation, Executable graphic files, tables, textual documents, databases, etc. All the popular sizing techniques are not supported for perfect sizing[25–30]. So LOP method is introduced for considering all the aspects of E-Learning system to produce accurate size.

Learning Object Points method is quantifying the size and complexity of an E-Learning system in terms of learning objects and functionalities. Sizing is independent of computer languages, development methodology and technology behind the development[3–5]. LOP can be estimated early in the analysis and design. It is prepared based on the user perspective so users of the E-Learning system have a better understanding of what LOP are measuring.

2. Learning Object Point Method

It is a proposed method for sizing E-Learning system. LOP is a unit of measurement to express the amount of learning objects and operational functionalities an E-Learning system provides to a user. This method was introduced to overcome the drawbacks of existing size estimationtechniques[26–30]. In E-Learning system huge volume of input output transactions happened in the form of registration, Fund transfer, Submission of learning and assignments contents, etc. Numerous logical files involved for eligibility checking, grading calculation, grouping, ordering, assessing, etc. Interface files support for connecting external components like database to our application. E-Learning system is a web application so number of Web pages associated. Screens and reports are associated with E-Learning system. Screens act as a user interface and reports are the great output expected by the stakeholders of the system. It has huge volume of multimedia files, graphic files, databases and internet based knowledge transfer happened. All these aspects of E-Learning system are considered in LOP method. It is the Modified version of ELSE proposed by T. S. Shiny Angel et al. in 2012.

2.1 Architecture of LOP Method

It has three major components to cover all the aspects of software. The first component is Unadjusted Learning Point (ULP), which includes Number of Inputs (NI), Number of Outputs (NO), Number of Files (NF), Number of Interfaces (NI), Number of Web Pages (NWP), Number of Screens and Reports (NSR), Duration of Multimedia Files (DMF), Number of Graphic Files (NGF) and Number of Document Pages (NDP). The second component is Technical Complexity Factor (TCF), It is calculated based on the 14 system Characteristics which includes Data communications, Distributed data processing, Performance, Heavily used configuration, Transaction rate, On-Line data entry, End-user efficiency, On-Line update, Complex processing, Reusability, Installation ease, Operational ease, Multiple sites, Facilitate change and the third component is Learning Complexity Factor (LCF) which includes Familiar with E-Learning System Development (FELSD), Analyst Capability (AC), Motivation (M), Requirement Stability (RS), Number of Courses (NC) and Expected Students Strength (ESS). The following Figure 1 shows the architecture of LOP Method.

The LOP of an E-Learning system is calculated by using Equation 1:

$$LOP = ULP * TCF * LCF \quad (1)$$

Where

LOP - Learning Object Points
ULP - Unadjusted Learning Points
TCF - Technical Complexity Factor
LCF - Learning Complexity Factor

Steps to Calculate LOP

- *Identify the counting Scope and application boundary. For small projects take the scope as a whole. For large projects, split them in to multiple parts, calculate LOP for each part and sum up together to receive a final count.*

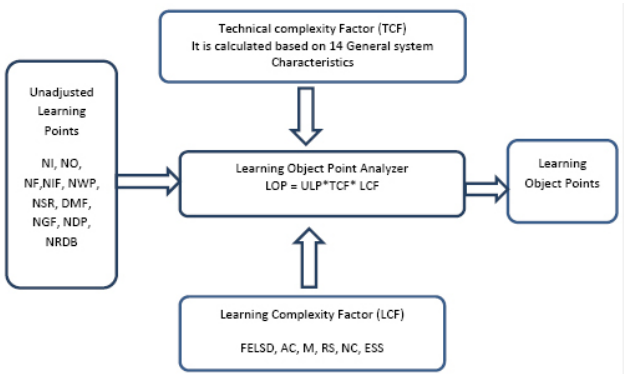

Figure 1. Architecture of LOP Method.

- *Determine Unadjusted Learning Points.*
- *Determine technical Complexity Factor*
- *Determine Learning Complexity Factor*
- *Calculate LOP of a given system*

2.2 Determination of Unadjusted Learning Points (ULP)

ULP can be calculated with the help of ten major components of E-Learning system. The components of ULP are as follows:

NI – *Number of Inputs:* Number of Inputs accepted by E-Learning system. It is an elementary process in which data crosses the boundary from outside to inside. This data may come from a data input screen or another application. The data may be used to maintain one or more logical files. The data can be either control information or business information.

NO – *Number of Outputs:* Number of Outputs produced by an E-Learning system. It is an elementary process in which derived data passes across the boundary from inside to outside.

NF – *Number of Files:* Number of Files used to process data in E-Learning system.

NIF – *Number of Interface Files:* Number of Files referenced by the application, but maintained within the boundary of another application.

NWP – *Number of Web Pages:* E-Learning system is a web application. So it may have huge number of web pages.

NSR – *Number of Screens and Reports:* E-Learning system used multiple screens and reports for receiving inputs and providing outputs.

DMF – *Duration of Multimedia Files:* E-Learning system has huge volume of multimedia files used to deliver learning content. They may be in the form of video, audio, animation or simulation.

NGF - *Number of Graphic Files:* E-Learning system used graphical files for demonstrating the learning content. They may be in the form of Images, images with special effect, diagrams, text with special effects, structured tables, etc.

NDP – *Number of Document Pages:* It represents the number of pages used to express learning content in E-Learning system.

NRDB – *Number of Records in Databases:* It represents total number of records accepted by E-Learning system.

Steps to determine ULP

1. Determination of Unadjusted Learning Objects (ULO). It is for calculating the count of learning objects.
2. Determination of Unadjusted Other object Points (UOP). It is for calculating the count of operational functionalities in a system.
3. Calculate ULP. ULP is calculated by using Equation 2.

$$ULP = ULO + UOP \quad (2)$$

Where ULO - Unadjusted Learning Objects.
UOP - Unadjusted Other Object Points.

2.3 Determination of Unadjusted Learning Objects (ULO)

Components *DMF, NGF* and *NDP* are used to determine the size of learning content. This course content may be delivered in the form of full video, audio, textual document, document with simulation or animation otherwise combination of all. For small projects calculate ULO as a whole. For large systems, if all the course contents having similar type of objects then calculate the ULO of one course and multiply with number of courses(n). For all the courses in a system having numerous form of learning objects then group them based on similarity, calculate the ULO of each group and sum up together provide ULO of a system. The DMF component assessed based on the duration of multimedia files deliver the learning objects. Count the number of Graphic Files involved in the course content. Count the document pages or number of slides delivers the course content. Find the complexity of each component. Table 1 assists to assign complexity of learning content. The complexity level to be assessed as Low, Average, High and Very high.

Table 1. Learning complexity assessment

Components	Low	Average	High	Very High
Duration of multimedia files	1 to 15 hour	16 to 30 hour	31 to 45 hour	>45 hour
Number of graphic files	1 to 15 clips	16 to 30 clips	31 to 45clips	>45clips
Number of graphic files	1 to 375 pages or slides then Transaction value should be 50	376 to 750 then Transaction value should be 100	751 to 1125 then Transaction value should be 150	>1125 then Transaction value should be 200

After learning complexity assessment, assign proper weights. Table 2 shows the weights for learning complexity.

The counts for NGF and NDP components and total time required for multimedia file can be entered into following Table 3. It is used to calculate Unadjusted Learning Objects. Each count is multiplied by the weight value shown on Table 2 determine the rate of each component except NDP because developing NCP in E-Learning system is not complex as like coding so a fixed value given. The rated values on each row are summed across the table, giving a total value for each type of component. These totals are then summed down to arrive at the Unadjusted Learning Objects.

2.4 Determination of Unadjusted Other Object Points (UOP).

Count the number of all other components considered for determining ULP. Rate them based on their complexity. Complexity should be specified based on the count of each component. Count is high then complexity is high, otherwise low or average. Table 4 assists to find complexity of a component.

Find the count and complexity of each component using Table 4 and fix the weight value of each component using Table 5. The weight values are specified based on the good features of existing sizing techniques.

The counts for all components can be entered into Table 6 other then Number of records in databases because instead of count use transaction value should be assigned for specifying UOP of a component and each count is multiplied by the weight value shown on table determine the rated value. The rated values on each row are summed across the table, giving a total value for each type of component. These totals are then summed down to arrive at the UOP of E-Learning System.

Let us have unadjusted learning object size and Unadjusted Other object Points (UOP), these values enough to calculate Unadjusted Learning Points (ULP). The Equation 2 is used for calculating ULP.

Table 2. Weights for learning complexity

Components	Low	Average	High	Very High
Duration of multimedia files	7	10	15	24
Number of graphic files	4	6	8	13
Number of document pages	50 (Fixed Value)	100 (Fixed Value)	150 (Fixed Value)	200 (Fixed Value)

Table 3. Unadjusted Learning Objects calculation

Components	Low	Average	High	Very High	Total
Duration of multimedia files	__*7= __	__*10=__	__*15=__	__*24=__	____
Number of graphic files	__*4=__	__*6=__	__*8=__	__*13=__	____
Number of document pages	50	100	150	200	____
Unadjusted Learning Objects					____

Table 4. Complexity assessment for other objects

Components	Low	Average	High	Very High
Number of Inputs	1-20	21-40	41-60	>60
Number of Outputs	1– 20	21-40	41-60	>60
Number of files	1– 10	11-25	26 - 50	>50
Number of Interface files	1-5	6-10	11-15	>15
Number of web pages	1 -5	6-10	11-20	>20
Number of screens and reports	1 -15	16 - 25	26 - 35	>35
Number of records in Databases	1– 10000 then Transaction value should be 5	10001 - 20000 then Transaction value should be 10	20001-30000then Transaction value should be 15	>30000 then Transaction value should be 25

Table 5. Weight values for other objects

Components	Low	Average	High	Very High
Number of Inputs	3	4	6	10
Number of Outputs	4	5	7	12
Number of files	4	10	15	22
Number of Interface files	5	7	10	16
Number of web pages	5	10	15	20
Number of screens and reports	3	7	11	16
Number of records in Databases	3	4	6	10

2.5 Determination of Technical Complexity Factor

The Technical Complexity Factor (TCF) is based on 14 General System Characteristics (GSC's) that rate the general functionality of the application being counted. Each characteristic has associated descriptions that help to determine the degrees of influence of the characteristics. The degrees of influence range on a scale of zero to five, from no influence to strong influence. The General System Characteristics are represented in Table 7.

Table 6. UOP calculation

Components	Low	Average	High	Very High	Total
Number of Inputs	__*3=__	__*4=__	__*6=__	__*10=__	____
Number of Outputs	__*4=__	__*5=__	__*7=__	__*12=__	____
Number of files	__*4=__	__*10=__	__*15=__	__*22=__	____
Number of Interface files	__*5=__	__*7=__	__*10=__	__*16=__	____
Number of web pages	__*5=__	__*10=__	__*15=__	__*20=__	____
Number of screens and reports	__*3=__	__*7=__	__*11=__	__*16=__	____
Number of records in Databases	__*3=__	__*4=__	__*6=__	__*10=__	____
Unadjusted Other object Points (UOP).					____

Table 7. General system characteristics

General System Characteristics	
1	Data communications
2	Distributed data processing
3	Performance
4	Heavily used configuration
5	Transaction rate
6	On-Line data entry
7	End-user efficiency
8	On-Line update
9	Complex processing
10	Reusability
11	Installation ease
12	Operational ease
13	Multiple sites
14	Facilitate change

Once all the 14 GSC's have been answered, they should be tabulated, using TCF equation and it also similar to FPA's Value Adjustment Factor calculation. Rate of factors varies from 0 to 5. The given factor not used in system then rate is 0. Otherwise the rate of a given factor is represented based on its influence. 0 - No influence, 1 – Incidental, 2 - Moderate, 3 - Average, 4 – Significant and 5 – Essential. Sum up all the influence level of each component and called as T sum. The TCF is calculated using Equation 3.

$$TCF = 0.65 + (0.01 * Tsum) \quad (3)$$

2.6 Determination of Learning Complexity Factor (LCF)

LCF addresses the skills, performance and abilities of development environment. This factor influence to productivity and stability. There are six factors associated with LCF calculation is represented in the Table 8.

Rate each factor's influence from 0 to 5. Zero denotes no experience in E-Learning system development; poor in analyst capability, motivation and requirement stability and courses and strength are not mentioned. Five denotes high experience in E-Learning system development; very high in analyst capability, motivation, requirement stability, courses and strength are expected level. For each factor, multiply the degree of influence by the weight and sum all products to obtain learning complexity sum, Lsum. Compute LCF using Equation 4.

$$LCF = 1.4 - 0.03 * Lsum \quad (4)$$

2.7 Determination of LOP

To determine LOP, three important factors are necessary, they are Unadjusted Learning Points (ULP), Technical Complexity Factor (TCF), Learning Complexity Factor

Table 8. Factors to assess learning complexity and their weights

Factor	Description	weight
F1	Familiar with E-Learning system Development (FESD)	1.5
F2	Analyst Capability (AC)	0.5
F3	Motivation (M)	1
F4	Requirement Stability (RS)	2
F5	Number of Courses (NC)	2
F6	Expected Students Strength (ESS)	2

(LCF). After determine these three factors, compute Learning Object Point (LOP) using the following Equation 1.

3. Result Analysis and Discussion

To analyze the performance of LOP Method we decided to compare its results with traditional FPA on real FPA data received from software industry. It consists of nine E-Learning projects and their function point calculations. In addition with collected each project how long it took to develop, estimated duration, how many people worked on the project and actual duration to finish. The Function Point calculated based on unadjusted Function points and Value Adjustment Factors. The Following Tables 9, 10, 11 describe them.

Table 9. Unadjusted function points for nine projects

Function Type	Fn. Complexity		Simple Projects						Medium Projects						Complex Projects					
			Project1		Project2		Project3		Project1		Project2		Project3		Project1		Project2		Project3	
			Weight	UFPs	Weight	UFPs	Weight	UFPs	Weight	UFPs	Weight	UFPs	Weight	UFPs	Weight	UFPs	Weight	UFPs	Weight	UFPs
EI	Low	3	14	42	11	33	12	36	15	45	16	48	17	51	16	48	17	51	18	54
	Average	4	3	12	5	20	6	24	6	24	4	16	4	16	6	24	7	28	8	32
	High	6	0	0	0	0	1	6	0	0	1	6	1	6	4	24	4	24	5	30
EO	Low	4	10	40	11	44	7	28	14	56	17	68	18	72	16	64	19	76	22	88
	Average	5	4	20	4	20	7	35	4	20	2	10	3	15	6	30	3	15	2	10
	High	7	0	0	0	0	1	7	1	7	0	0	1	7	2	14	2	14	1	7
EQ	Low	3	2	6	2	6	2	6	3	9	4	12	3	9	2	6	4	12	4	12
	Average	4	3	12	2	8	1	4	2	8	0	0	1	4	1	4	3	12	1	4
	High	6	0	0	0	0	0	0	0	0	0	0	0	0	1	6	0	0	1	6
ILF	Low	7	2	14	3	21	3	21	3	21	0	0	0	0	0	0	0	0	0	0
	Average	10	0	0	0	0	0	0	0	0	4	40	4	40	5	50	5	50	5	50
	High	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
EIF	Low	5	2	10	3	15	3	15	3	15	0	0	0	0	0	0	0	0	0	0
	Average	7	0	0	0	0	0	0	0	0	4	28	4	28	5	35	5	35	5	35
	High	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Total of UFPs				156		167		182		205		228		248		305		317		328

Table 10. Value adjustment factors calculation

Sno	Apply Environmental Factors	Rating (0, 1, 2, 3, 4, 5)								
		Simple Projects			Medium Projects			Complex Projects		
1	Data Communication	1	2	1	1	1	2	1	2	3
2	Distributed Data Processing	4	4	4	4	4	4	4	4	4
3	Is performance critical?	1	2	2	3	3	3	4	4	4
4	Heavily Used configuration	2	2	2	2	2	3	3	3	3
5	Transaction Rate	1	1	1	2	2	2	3	3	3
6	Onine Data Entry	2	3	2	2	3	3	2	3	3
7	End-User Efficiency	3	4	4	4	4	4	4	4	5
8	Online Update	1	2	1	2	1	2	2	1	2
9	Complex Processing	2	3	3	2	3	3	3	4	4
10	Resuability	3	3	3	3	3	4	3	3	4
11	Installation Ease	2	2	2	2	3	3	2	3	3
12	Operational Ease	2	2	2	2	2	2	2	2	2
13	Multiple Site	0	0	0	0	0	0	0	0	0
14	Facility Change	1	2	1	2	2	2	2	2	3
	Degree of Influence	8	9	8	9	10	11	9	10	12
	VAF	0.73	0.74	0.73	0.74	0.75	0.76	0.74	0.75	0.77

Table 11

Projects	UFP	VAF	FP
Project1	156	0.73	113.88
Project2	167	0.74	123.58
Project3	182	0.73	132.86
Project4	205	0.74	151.7
Project5	228	0.75	171
Project6	248	0.76	188.48
Project7	305	0.74	225.7
Project8	317	0.75	237.75
Project9	328	0.77	252.56

The Table 12 shows size and effort calculated for nine different E-Learning based projects using FPA. The actual duration increased 2 to multiples times more than estimated because some projects used videos, simulations or animations it took more time for editing and formatting. It shows FPA is not suitable one for estimating the size of E-Learning Systems.

The following Equations 5 and 6 are applied for calculating Estimated Effort and Estimated time.

$$Estimated\ Effort = FP * PF \quad (5)$$

Where FP = Function Points calculated for an Application and PF = Productivity Factor. The productivity factor may change from organization to organization. Our organization uses productivity factor as 16 because they took in and average 16 hours per Function points.

$$Estimated\ Time = Estimated\ Effort/(176 * Human\ Resources) \quad (6)$$

Here 176 denote working hours per month that means Indian software industry people work on 22 days per month and per day 8 hours, totally 22*8 = 176 hours.

3.1 Measuring Learning Object Points

Measuring LOP, we used FPA data given from the industry, their requirement specification document and the initial business model. Using the information we calculated Unadjusted Learning Objects (ULO) and Unadjusted Other object Points (UOP). Using them Unadjusted Learning Point ULP calculated. Tables 13 and 14 describe them.

Using Tables 13 and 14 ULP calculated, it shows in Table 16. For LOP calculation other two components needed are Technical Complexity Factor (TCF) and Learning Complexity Factor (LCF). Technical

Table 12. Effort and development time variance

Projects	UFP	VAF	FP	Human Resources	Estimated Effort(In Man Hours)	Estimated Time (in Months)	Actual Time (in Months)	Time Variance(in Months)
Project1	156	0.73	113.88	5	1822.08	2.07	5	2.93
Project2	167	0.74	123.58	5	1977.28	2.25	21	18.75
Project3	182	0.73	132.86	5	2125.76	2.42	21	18.58
Project4	205	0.74	151.7	5	2427.2	2.76	7	4.24
Project5	228	0.75	171	5	2736	3.11	14	10.89
Project6	248	0.76	188.48	5	3015.68	3.43	10	6.57
Project7	305	0.74	225.7	5	3611.2	4.1	10	5.9
Project8	317	0.75	237.75	5	3804	4.32	23	18.68
Project9	328	0.77	252.56	5	4040.96	4.59	8	3.41

Table 13. Unadjusted Learning Objects (ULO)

Components	fn complexity		Project1	ULO of a Component	Project2	ULO of a Component	Project3	ULO of a Component	Project4	ULO of a Component	Project5	ULO of a Component	Project6	ULO of a Component	Project7	ULO of a Component	Project8	ULO of a Component	Project9	ULO of a Component
Duration of Multimedia Files	Low	**9**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Average	**10**	25	250	0	0	0	0	25	250	0	0	0	0	0	0	0	0	0	0
	High	**15**	0	0	0	0	0	0	0	0	45	675	0	0	0	0	0	0	0	0
	Very High	**25**	0	0	60	1500	60	1500	0	0	0	0	0	0	0	0	60	1500	0	0
No.Of Graphic Files	Low	**4**	0	0	0	0	0	0	15	60	0	0	0	0	0	0	0	0	0	0
	Average	**6**	0	0	0	0	0	0	0	0	24	144	0	0	0	0	0	0	25	150
	High	**8**	0	0	0	0	0	0	0	0	0	0	48	384	40	320	0	0	0	0
	Very High	**14**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Number of Document Pages	Low	**50**	0	0	0	0	0	0	250	50	0	0	0	0	0	0	0	0	0	0
	Average	**100**	0	0	0	0	0	0	0	0	600	100	0	0	0	0	0	0	500	100
	High	**150**	0	0	0	0	0	0	0	0	0	0	800	150	900	150	0	0	0	0
	Very High	**200**	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ULO				**250**		**1500**		**1500**		**360**		**919**		**534**		**470**		**1500**		**250**

Complexity Factor calculation is as same as VAF calculation of FPA. It considered 14 general system characteristics. So we used Table 9 for TCF calculation. Table 15 describes LCF calculation.

Using Tables 13, 14, 15 and 10 LOP of 9 projects calculated, that is given in Table 16.

After LOP effort and development Time Variance to be calculated. The Table 17 shows size effort and development time variance calculated for nine different E-Learning based projects using LOP. It produced minor differences. It implies LOP is more suitable for sizing E-Learning projects.

Table 14. Unadjusted other Object Points (UOP)

Components	fn complexity		Project1	UOP of a Component	Project2	UOP of a Component	Project3	UOP of a Component	Project4	UOP of a Component	Project5	UOP of a Component	Project6	UOP of a Component	Project7	UOP of a Component	Project8	UOP of a Component	Project9	UOP of a Component
No. of inputs	Low	3	17	51	16	48	19	57	0	0	0	0	0	0	0	0	0	0	0	0
	Average	4	0	0	0	0	0	0	21	84	21	84	22	88	26	104	28	112	31	124
	High	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Very High	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
No of outputs	Low	4	14	56	15	60	15	60	19	76	19	76	0	0	0	0	0	0	0	0
	Average	5	0	0	0	0	0	0	0	0	0	0	22	110	24	120	24	120	25	125
	High	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Very High	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Number of Files	Low	7	2	14	3	21	3	21	3	21	4	28	4	28	5	35	5	35	5	35
	Average	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	High	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Very High	25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Number of interface Files	Low	5	2	10	3	15	3	15	3	15	4	20	4	20	5	25	5	25	5	25
	Average	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	High	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Very High	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
No. of Web Pages	Low	5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Average	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	High	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Very High	25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Number of Screens and Reports	Low	3	7	21	7	21	7	21	10	30	10	30	10	30	10	30	12	36	12	36
	Average	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	High	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Very High	18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Number of Records in Databases	Low	3	5	15	5	15	5	15	5	15	5	15	5	15	5	15	5	15	5	15
	Average	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	High	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Very High	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
UOP of Projects				**167**		**180**		**189**		**241**		**253**		**291**		**329**		**343**		**360**

4. Comparison with LOP and FPA

Size and effort calculated for nine different E-Learning based projects using FPA produced inaccurate results. So the project management activities like planning, scheduling and costing produced imprecise outcomes but LOP produced 95% approximate results so it supports project management activities effectively. The following Figure 2 shows the actual development time, estimated time using LOP and FPA. FPA produced more differences and LOP is more close to actual development time.

5. LOP versus other Popular Estimation Techniques

In the software project management view LOP is too good for estimating the size of E-Learning system. The following table shows the comparison between LOP and

Table 15. Determination of Learning Complexity Factor (LCF)

Factors	Weight	Project1	LCF of a Component	Project2	LCF of a Component	Project3	LCF of a Component	Project4	LCF of a Component	Project5	LCF of a Component	Project6	LCF of a Component	Project7	LCF of a Component	Project8	LCF of a Component	Project9	LCF of a Component
Familier with E-Learning System Development	**1.5**	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3
Analyst Capability	**0.5**	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1
Motivation	**1**	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
Requirement Stability	**2**	2	4	2	4	2	4	2	4	2	4	2	4	2	4	2	4	2	4
Number of Courses	**2**	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
Expected Students Strength	**2**	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
Learning Complexity Sum(Lsum)			17		17		17		17		17		17		17		17		17
Learning Complexity Factor(LCF)			0.9		0.9		0.9		0.9		0.9		0.9		0.9		0.9		0.9

Table 16. LOP of given projects

Projects	ULO	UOP	ULP	TCF	LCF	LOP
Project1	250	167	417	0.73	0.9	273.969
Project2	1500	180	1680	0.74	0.9	1118.88
Project3	1500	189	1689	0.73	0.9	1109.673
Project4	360	241	601	0.74	0.9	400.266
Project5	919	253	1172	0.75	0.9	791.1
Project6	534	291	825	0.76	0.9	564.3
Project7	470	329	799	0.74	0.9	532.134
Project8	1500	343	1843	0.75	0.9	1244.025
Project9	250	360	610	0.77	0.9	422.73

Table 17. Effort and development time variance

Projects	LOP	Human Resources	Estimated Effort(In Man Hours)	Estimated Time (in Months)	Actual Time (in Months)	Time Variance(in Months)
Project1	273.969	5	4383.504	4.98	5	0.02
Project2	1118.88	5	17902.08	20.34	21	0.66
Project3	1109.673	5	17754.768	20.18	21	0.82
Project4	400.266	5	6404.256	7.28	7	-0.28
Project5	791.1	5	12657.6	14.38	14	-0.38
Project6	564.3	5	9028.8	10.26	10	-0.26
Project7	532.134	5	8514.144	9.68	10	0.32
Project8	1244.025	5	19904.4	22.62	23	0.38
Project9	422.73	5	6763.68	7.69	8	0.31

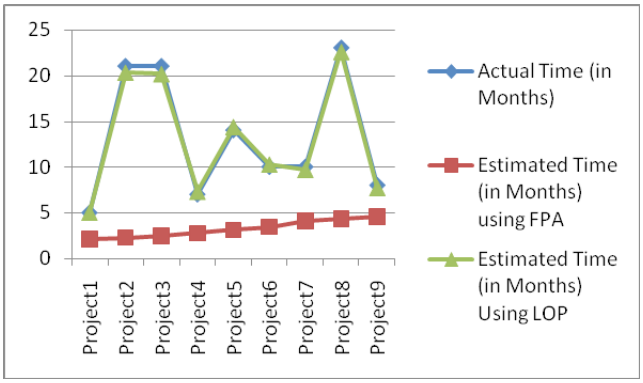

Figure 2. Development time comparison.

other popular sizing techniques based on the features of E-Learning system. Table 18 describes the comparison. LOP considered all the needful features of E-Learning system.

In E-Learning system huge volume of input output transactions happened in the form of registration, fund transfer, submission of learning and assignments contents, etc. Numerous logical files involved for eligibility checking, grading calculation, grouping, ordering, assessing, etc. Interface files support for connecting

Table 18. Comparative analysis of sizing techniques in the sense of E-Learning system sizing

Techniques/ Features of E-Learning System	LOC	FPA	Feature Point	Use case point	Object point	Internet points(Web count, Web objects)	Learning Object Point
Input and output consideration	Nil	Inputs and outputs are the two separate components for calculating Unadjusted function points	Inputs and outputs are the two separate components for calculating Raw feature points	Inputs are passed from actor and outputs are passed to actors. It count the number of actors but it never care input and output transactions	Nil	Nil	Inputs and outputs are the two separate components for calculating Unadjusted Learning points
Logical files involvement	Count the number of executable codes of each file for calculating LOC	Count all files for calculating Unadjusted function points	Count all files for calculating Raw feature points	Mostly the logical files are the use cases. Use case modeling shows the major use cases. So it may not consider all the logical files involved in application	In the form of rule set or 3GL module, all the logical files are considered	Little bit considered.	Count all files for calculating Unadjusted Learning points
Interface files	Nil	Count all interface files for calculating Unadjusted function points	Count all interface files for calculating Raw feature points	Nil	Nil	Considered in the form of hyperlinks	Count all interface files for calculating Unadjusted Learning points
Web pages	Nil – it may count the lines of script but it is not useful because single executable instruction is not equal to single line of textual information in the script.	Nil	nil	nil	nil	Good for sizing web pages	Good for sizing web pages
Screens and reports (GUI Support)	Nil	Calculate inputs and outputs but it never identify the worth of screens and reports	Nil	Nil	Major components to calculate OP	Nil	Major components to calculate Unadjusted Learning points
Multimedia files – video, audio, simulation, animation	Nil	Nil	Nil	Nil	Nil	Little bit considered. Because the sizing is suitable for small website not for document rich E-Learning system	All kind of multimedia files are considered and their complexity are assessed for sizing

(*Continued*)

Graphic files	nil	Nil	Nil	Nil	Nil	Little bit considered Because the sizing is suitable for small website not for document rich E-Learning system	All kind of Graphic files are considered and their complexity are assessed for sizing
Textual document	It may count the lines of document. But it is not good	nil	nil	nil	nil	Count number of words and number of pages. So estimating the size of small web site it may good	Number of textual pages are encounter for sizing
Accuracy in sizing of E-Learning system	Inaccurate	Inaccurate	Inaccurate	Inaccurate	Inaccurate	Inaccurate	Accurate
Quality in sizing of E-Learning system	Nil	Nil	Nil	Nil	Nil	Nil	Provides Expected quality
Reusability	Nil	Considered	Nil	Nil	Highly considered	Nil	Considered
Database Support	Nil	Nil	considered	Nil	Nil	Nil	Highly considered
Data communication	Nil	Considered as one of the complexity adjustment factor	Nil	Nil	Nil	Nil	Highly considered

external components like database to our application. E-Learning system may a web application so number of web pages associated. Screens and reports are associated with E-Learning system. Screens act as a user interface and reports are the great output expected by the stakeholders of the system. It has huge volume of multimedia files, graphic files, databases and internet based knowledge transfer happened. The comparison table prepared based on the features of E-Learning system.

6. Conclusion

One of the main difficulties in project management is to estimate the project size to be able to deduce the important factors like cost and effort. There are many approaches to estimate the size of the project but are not fully supporting for estimating the size of E-Learning system. For E-Learning system LOP is considering all the physical, technical and learning content aspects so sizing using LOP is more suitable for project management activities.

7. References

1. Allen IE, Seaman J. Sizing the opportunity: The quality and extent of online education in the United States. Wellesley, MA: The Sloan Consortium; 2003.
2. Allen IE, Seaman J. Staying the course: Online education in the United States. Needham MA: Sloan Consortium; 2008.

3. Timp A. Function Point Counting practices manual release 4.3.1. USA: International Function Point Users Group (IFPUG); 2010.
4. Hughes B, Cotterell M, Mall R. Software project management. 5th Ed. USA: Tata McGraw Hill; 2011.
5. Jones C. Estimating software costs: Bringing realism to estimating. USA: McGraw Hill Companies; 2011.
6. Hari CHVMK, Reddy PVGD. A fine tuning parameter for COCOMO81 software effort estimation using Particle Swam Optimization. Journal of Software Engineering. 2011; 5(1):38–48.
7. Ferchichi A, Bourey JP, Bigand M, Barron M. Design system engineering of software products implementation of a software estimation model. IMACS-Multi Conference on Computational Engineering in Systems Applications. Beijing, China. 2006; 2:1181–8.
8. Heemstra FJ. Software cost estimation, information and software technology. 1992; 34(10):627–39.
9. Furey S. Why we should use function points. IEEE on Software. 1997; 14(2):28–30.
10. Land SK. Software effort by combining estimation models and quality assurance. Results of the IEEE Survey of Software Engineering Standards Users; 1997. p. 242–69.
11. Tian L, Noore A. Multistage software estimation. Proceedings of the 35th Southeastern Symposium on System Theory; 2003. p. 232–6. .
12. Alves LM, Oliveira S, Ribeiro P, Machado RJ. An empirical study on the estimation of size and complexity of software applications with function point analysis. 14th International Conference on Computational Science and its Applications; Guimaraes. 2014. p. 27–34.
13. Mehta HK, Chandwani M, Kanungo P. Towards development of a distributed E-Learning eco system. 2010 International Conference on Technology for Education (T4E); Mumbai. 2010. p. 68–71.
14. Kaur M, Sehra SK. Particle Swarm Optimization based on effort estimation using function point analysis. International Conference on Issues and Challenges in intelligent Computing Techniques (ICICT); Ghaziabad. 2014. p. 140–5.
15. Phankokkruad M. Implementation of cloud computing for E-Learning system. 2012 International Conference on Computer and Information Science (ICCIS). Kuala Lumpeu. 2012; 1:7–11.
16. Nasir M. A survey of software estimation techniques and project planning practices. IEEE 7th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, (SNPD'06); Las Vegas, NV. 2006. p. 305–10.
17. Fenton NF, Pfleeger SL. Software metrics: A rigorous and practical Approach. USA: CRC Press; 1997.
18. Stutzke RD. Estimating software - Intensive systems: Projects, products and processes. USA: Addison Wesley; 2005.
19. Futrell RT, Shafer DF, Shafer LF. Quality software project management. USA: Prentice Hall; 2001.
20. Barros RC, Ruiz DD, Tenorio NN. Issues on estimating software metrics in a large software operation. 32nd Annual IEEE Software Engineering Workshops, SEW'08; Kassandra. 2008. p. 152–60.
21. Gharehchopogh FS, Ebrahimi L, Maleki I, Gourabi SJ. A novel PSO based approach with hybrid of Fuzzy C-means and learning automata in software cost estimation. Indian Journal of Science and Technology. 2014; 7(6):795–803.
22. Mesia Dhas JT, Bharathi CR. Risks associated to size estimation of E-Commerce system using function point based estimation technique. Indian Journal of Science and Technology. 2016; 9(7):1–6.
23. Kyoo-sung N, Doo-sik L. Bigdata platform design and implementation model. Indian Journal of Science and Technology. 2015; 8(18):1–8.
24. Nagarajan R, Velanganni Joseph A, Sujatha S. Behavioral aspects of software project management- in- house software development. Indian Journal of Science and Technology. 2015 Feb; 8(S3). DOI: 10.17485/ijst/2015/v8iS3/58771.
25. Nirmalraj D, Santhosh N. Model to predict schedule variance in software application development projects. Indian Journal of Science and Technology. 2016 Feb; 9(7). DOI: 10.17485/ijst/2016/v9i7/85751.
26. Angel TSS, Rodrigues P, Mesiah Dhas JT. Limitations of function point analysis in E-Learning system estimation. International Journal of Computational Engineering Research IJCER; 2012. p. 156–61
27. Angel TSS, Rodrigues P. ELSE: E-Learning system estimator. International Review on Computers and Software. 2012; 7(3):3033–6.
28. Angel TSS, Rodrigues P, Mesiah Dhas JT, Prabhu T. Size and flexibility metrics for E-Learning system. International Journal of Software Engineering and Technology (IJSET); 2012; 4(7).
29. Angel TSS, Rodrigues P. Sizing approach for E-Learning system. International Journal of Emerging Trends in Computer Science; 2013. p. 1–10.
30. Angel TSS, Rodrigues P. Comparative analysis of sizing techniques in the sense of E-Learning system. IJAER. 2015; 10(5):139–50.