ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

Empirical Analysis of Software Projects - Research View

Vuppu Padmakar[1], Dr. B V Ramana Murthy[2], Dr. Vaibhav Bansal[3]

Research Scholar, Department of CSE, OPJS University, Rajasthan, India

Professor, Department of CSE, Mahaveer Institute of Science & Technology, Hyderabad, India

Associate Professor, Department of CSE, OPJS University, Rajasthan, India

ABSTRACT: Empirical Analysis of software projects: Software projects is not a perfect clone of software engineering, but it borrows many of software engineering fundamental concepts and principles. we have addressed the numerical/empirical analysis of software projects, which enables the project manager and software developers to understand software engineering better. We have limited our study to software management in this area. A questionnaire was prepared for the study, based on the four important concepts of software engineering, namely, software attributes, software methods, software framework, software practices. The attributes relevant for each of the above concepts are identified after developers in the industry. The AHP technique is used for designing and evaluating the responses to the questionnaires[6,7]. The AHP techniques given the weights of the attributes considered in the questionnaire, which helps the project manager as a indicator for the further process. Once attributes weights are known group-wise, the task that follows is testing their applicability in practices and relevance of the attributes identified. For the purpose, the questionnaire has been applied for nine real-time projects which are under use and developed by a project manager. The projects were ranked based on the 36 attributes put under 4 graphics, by generating pair-wise comparison matrices using AHP techniques. The Ideal Point Analysis is used which integrates software attributes, weights of respondents and attributes wise weights of projects using AHP given by project manager. The Ideal Point Analysis helps in rating the projects and validation of the research.

KEYWORDS: Software Projects, AHP, FP, Project Manager

I. INTRODUCTION

There are many different types of software System and there is no universal set of software techniques that is applicable to all of these. The software engineering methods and tools used depend on the type of application being developed, the requirements of the customer and the background of the development team.

Application Types

1. **Stand-alone applications:** These are application systems that run on a local computer, such as a PC. They include all necessary functionality and do not need to be connected to a network.

2. **Interactive transaction-based applications:** Applications that execute on a remote computer and are accessed by users from their own PCs or terminals. These include web applications such as e-commerce applications.

3. **Embedded control systems:** These are software control systems that control and manage hardware devices. Numerically, there are probably more embedded systems than any other type of system.

4. **Batch processing systems:** These are business systems that are designed to process data in large batches. They process large numbers of individual inputs to create corresponding outputs.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

5. **Entertainment systems:** These are systems that are primarily for personal use and which are intended to entertain the user.

6. **Systems for modelling and simulation:** These are systems that are developed by scientists and engineers to model physical processes or situations, which include many, separate, interacting objects.

7. **Data collection systems:** These are systems that collect data from their environment using a set of sensors and send that data to other systems for processing.

8. **Systems of systems:** These are systems that are composed of a number of other software systems. Some fundamental principles apply to all types of software system, irrespective of the development techniques used: Systems should be developed using a managed and understood development process. Of course, different processes are used for different types of software. Dependability and performance are important for all types of system. understanding and managing the software specification and requirements (what the software should do) are important. Where appropriate, you should reuse software that has already been developed rather than write new software.

II. RELATED WORK

Key Points to develop a software products Software engineering is an engineering discipline that is concerned with all aspects of software production. Essential software product attributes are maintainability, dependability and security, efficiency and acceptability. The high-level activities of specification, development, validation and evolution are part of all software processes. The fundamental notions of software engineering are universally applicable to all types of system development. There are many different types of system and each requires appropriate software engineering tools and techniques for their development. The fundamental ideas of software engineering are applicable to all types of software system.

Empirical Analysis of Software Projects Motivation and Background:-Software projects -based systems and applications deliver a complex array of content and functionality, to a broad population of end-users. Project management is the process that is used to create high-quality software applications. Project management is a perfect clone of a software engineering, but it borrows many of software engineering fundamental concepts and principles. In addition, the project management process emphasizes similar technical and management activities are conducted, but the overriding philosophy dictates a disciplined approach to the development of a computer based system. Software Engineers and non-technical content developers create the software applications[8,9]. As software projects becomes increasingly integrated in business strategies, for small and large companies, the need to build reliable, usable, and adaptable systems grows in importance. That is why a disciplined approach to software application development is necessary[10]. Like any engineering discipline, software engineering applies a generic approach that is tempered with specialized strategies, tactics and methods.

The software engineering process begins with the formulation of a problem to be solved by the software application. The software engineering project is planned and the requirements and design of the application are modelled. The system is constructed, using specialized technologies and tools associated with the software. It is then delivered to end-users and evaluated, using both technical and business criteria. Because software applications evolve continuously, mechanism for configuration control, quality assurance, and on-going support must be established.

Objective: To enable the Project Manager/Software developers to understand Project Management better.

Scope: The research work is helpful for Project Management.

Problem Statement: To provide an indication of the quality of the project management from a technical point of view.

Software projects are not a perfect clone of software engineering, but it borrows many of software engineering fundamental concepts and principles. we have addressed the numerical/empirical analysis of software projects, which

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

enables the project manager and software developers to understand software engineering better. We have limited our study to software management in this area. A questionnaire was prepared for the study, based on the four important concepts of software engineering, namely, software attributes, software methods, software framework, software practices. The attributes relevant for each of the above concepts are identified after developers in the industry. The AHP technique is used for designing and evaluating the responses to the questionnaires[6,7]. The AHP techniques given the weights of the attributes considered in the questionnaire, which helps the project manager as a indicator for the further process.

Once attributes weights are known group-wise, the task that follows is testing their applicability in practices and relevance of the attributes identified. For the purpose, the questionnaire has been applied for nine real-time projects which are under use and developed by a project manager. The projects were ranked based on the 36 attributes put under 4 graphics, by generating pair-wise comparison matrices using AHP techniques. The Ideal Point Analysis is used which integrates software attributes, weights of respondents and attributes wise weights of projects using AHP given by project manager. The Ideal Point Analysis helps in rating the projects and validation of the research.

The Software Project Management module suggested that management involves the following activities:

1. Planning – deciding what is to be done;
2. Organizing – making arrangements;
3. Staffing – selecting the right people for the job, etc.;
4. Directing – giving instructions;
5. Monitoring – checking on progress;
6. Controlling – taking action to remedy hold-ups;
7. Innovating – coming up with new solutions;
8. Representing – liaising with users, etc.

The management task is to ask mangers what their most frequent challenges are. A survey of software project mangers produced the following list.

1. Copy with deadlines - (85%);
2. Coping with resource constraints - (83%);
3. Communicating effectively among task group (80%);
4. Gaining commitment from team members (74%)
5. Establishing measurable milestones - (70%)
6. Coping with changes - (60%)
7. Working out project plan agreement with their team (57%)
8. Gaining commitment from management - (45%)
9. Dealing with conflict - (42%)
10. Managing vendors and sub-contractors - (38%)

The percentages relate to the numbers of managers identifying each challenge. A manager could identify more than one.

III. SIMULATION RESULTS

As the different software projects like ECS and ES types of projects holds the high cost in development and testing due to complexity of the production follows by DCS, SMSSIM then ECS ITBA and SCA. As the different software projects based on the weightages as suggested by the project manager the following weightages are calculated.

The empirical analysis of types of software projects applications to the development of the product gives project manager an view to gives estimation, the analysis will gives insight of the complexity being obtained by the different types of software projects applications in the form of weightages, more than weightages more to the complexity of development and testing in terms of estimation one more.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

Weights of the 10 sample of different types software's projects data

	SLA	ITBA	ECS	BPS	ES	SBSSIS	DCS	S&S	
SLA	1.00	0.17	0.17	0.14	0.33	0.14	0.17	0.33	
ITBA	6.00	1.00	0.25	2.00	5.00	0.50	3.00	3.00	
ECS	6.00	4.00	1.00	4.00	3.00	0.50	0.33	5.00	
BPS	7.00	0.50	0.25	1.00	4.00	5.00	4.00	4.00	
ES	3.00	0.20	0.33	0.25	1.00	0.33	0.25	0.25	
SMSSIS	7.00	2.00	2.00	0.20	3.00	1.00	2.00	4.00	
DCS	6.00	0.33	3.00	0.25	4.00	0.50	1.00	2.00	
S&S	3.00	0.33	0.20	0.25	4.00	0.25	0.50	1.00	
	39.00	8.53	7.20	8.09	24.33	8.23	11.25	19.58	
	SLA	ITBA	ECS	BPS	ES	SBSSIS	DCS	S&S	
SLA	0.03	0.02	0.02	0.02	0.01	0.02	0.01	0.02	0.15
ITBA	0.15	0.12	0.03	0.25	0.21	0.06	0.27	0.15	1.24
ECS	0.15	0.47	0.14	0.49	0.12	0.06	0.03	0.26	1.72
BPS	0.18	0.06	0.03	0.12	0.16	0.61	0.36	0.20	1.73
ES	0.08	0.02	0.05	0.03	0.04	0.04	0.02	0.01	0.29
SMSSIS	0.18	0.23	0.28	0.02	0.12	0.12	0.18	0.20	1.34
DCS	0.15	0.04	0.42	0.03	0.16	0.06	0.09	0.10	1.06
S&S	0.08	0.04	0.03	0.03	0.16	0.03	0.04	0.05	0.46
	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	

Weights of sample 10 for 'C' different projects

Like above the weights of samples of 1 to 10 for different software projects priority of weights range 1 to 7 for calculations and consolidated sample collected and their graph for different software projects weightages as follows.

10 Samples graph for different projects

	Sample1	**Sample2**	**Sample3**	**Sample4**	**Sample5**	**Sample6**	**Sample7**	**Sample8**	Sample9	**Sample10**
SLA	0.15	0.20	0.16	0.15	0.13	0.15	0.15	0.15	0.14	0.15
ITBA	1.24	1.19	1.88	1.33	1.25	0.90	1.28	0.72	0.96	0.73
ECS	1.72	2.48	1.39	1.51	1.16	1.22	1.43	1.17	1.14	0.98
BPS	1.73	1.50	1.31	1.77	1.00	1.29	1.09	1.00	1.40	1.67
ES	0.29	0.34	0.59	0.84	0.81	0.98	1.19	1.61	1.30	1.35
SMSSIS	1.34	1.14	1.14	1.11	1.71	1.47	0.50	0.66	1.04	1.13
DCS	1.06	0.64	0.97	0.70	1.07	1.20	1.65	1.37	0.68	0.74
S&S	0.46	0.51	1.51	0.59	0.86	0.79	0.71	1.31	1.33	1.25
	7.99	8.00	8.95	8.00	7.99	8.00	8.00	7.99	7.99	8.00

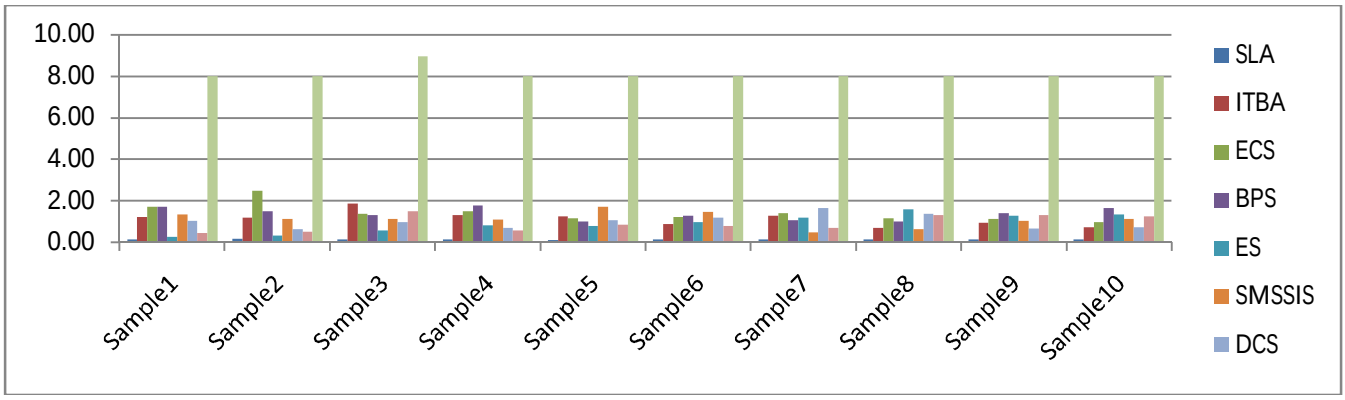
weights of 'C' different projects of 10 samples

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

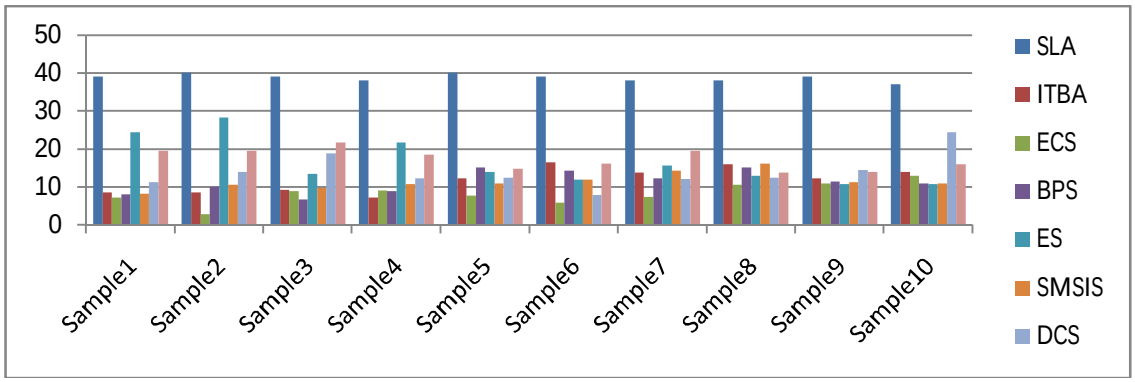


Weights of 'C' different projects of sample 10

Different projects 10 sample graphs

	SLA	ITBA	ECS	BPS	ES	SMSIS	DCS	S&S	
Sample1	39	8.53	7.2	8.09	24.33	8.23	11.25	19.58	126.21
Sample2	40	8.51	2.82	10.12	28.33	10.58	13.95	19.45	133.76
Sample3	39	9.28	8.98	6.81	13.5	9.78	18.89	21.64	127.88
Sample4	38	7.29	9.08	8.98	21.64	10.78	12.33	18.45	126.55
Sample5	40	12.37	7.75	15.07	14	10.87	12.41	14.83	127.3
Sample6	39	16.5	5.9	14.33	12	11.92	7.84	16.17	123.66
Sample7	38	13.74	7.42	12.23	15.59	14.33	12.06	19.58	132.95
Sample8	38	15.92	10.65	15.14	13.01	16.08	12.49	13.87	135.16
Sample9	39	12.23	11	11.45	10.83	11.33	14.48	13.93	124.25
Sample10	37	13.95	13	10.9	10.73	10.89	24.4	15.95	136.82
	387	118.32	83.8	113.12	163.96	114.79	140.1	173.45	

weights of sample 10 for 'C' different projects



weights of sample 10 for 'C' different projects

IV. Conclusion and Future Work

Software projects are not a perfect clone of software engineering, but it borrows many of software engineering fundamental concepts and principles. We have addressed the numerical/empirical analysis of software projects, which enables the project manager and software developers to understand software engineering better. We have limited our

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 2, February 2016

study to software management in this area. A questionnaire was prepared for the study, based on the four important concepts of software engineering, namely, software attributes, software methods, software framework, software practices. The attributes relevant for each of the above concepts are identified after developers in the industry. The AHP technique is used for designing and evaluating the responses to the questionnaires. The AHP techniques given the weights of the attributes considered in the questionnaire, which helps the project manager as a indicator for the further process.

REFERENCES

[1] Fenton, N., "Software Measurement: A Necessary Scientific Basis", IEEE Trans. Software Engineering, Vol. SE-20, no. 3, March 1994, pp. 199-206.
[2] Humphrey, W., A Discipline for Software Engineering, Addison-Wesley, 1995.
[3] Somerviue, I., Software Engineering, 6th ed. , Addison-Wesley, 2001.
[4] Gilb, T., Principles of Software Engineering Management, Addison-Wesley, 1988.
[5] Putnam, L., and W. Myers, Measures of Excellence, Yourdors Press, 1992.
[6] Felican, L., and G. Zalateu, "Validating Halstead's Theory for Pascal Programs", IEEE Trans. Software Engineering, Vol. SE-15, no. 2, December 1989, pp. 1630-1632.
[7] Albrecht, A. J., and J. E. Gaffney, "Software Function, Source Lines of Code and Development Effort Prediction: A Software Science Validation", IEEE Trans. Software engineering, November 1983, pp. 639-648.
[8] A. J. Albrecht, "Measuring application development productivity", Proc. of the Joint SHARE / GUIDE / IBM Application Development Symposium, Oct. 1979, pp. 83-92.
[9] A. J. Albrecht, J. E. Gaffney, "Software Function, source lines of code, and development effort prediction: A software science validation ",IEEE Trans. Software Eng. 9, no. 6 (1983) 639-648.
[10] J. B. Dreger, Function Point Analysis, Prentice-Hall. 1989.

BIOGRAPHY

Mr. V Padmakar is pursuing PhD in CSE, and has done his M Tech (CSE) from JNTUH, presently working as Professor in Computer Science and Engineering has 17 years of experience in Teaching and Industry. His primary area of interests is Software Engineering, Network Security and Data mining

Dr. B. V.Ramana Murthy has done his PhD from Osmania University, presently he working as Professor in Computer Science and Engineering, has 18 years of experience in Teaching and R&D. His primary area of interest is Software Engineering & Web Engineering.