International Journal of Advanced Science and Technology
Vol. 29, No. 7s, (2020), pp. 5164 – 5172

An Empirical Investigation on Practicing Secure Software Development in Software Development Life Cycle in Small & Medium Level Software Firms in Bengaluru

Sandhya Soman,
Dr. Piyush
Kumar Pareek
Dept of Computer Science
Research Scholar, Shri JJT University,
Rajasthan, India
Asst Professor ,Kristu Jayanti College,
Bengaluru, India
sandhyasomancg@gmail.com

Dept. of CSE
East West College of Engineering
Bengaluru, India
piyushpareek88@gmail.com

Veeranna Kotagi
Dept of Computer Science
Research Scholar, Shri JJT University,
Rajasthan, India
Asst Professor ,East West College of
Engineering
Bengaluru,
Indiaveerkotagi21@gmail.com

Abstract

Software security is a basic necessity for software frameworks. Notwithstanding, late examination shows that numerous software development systems don't expressly incorporate strategies for consolidating data security into the software development life cycles (SDLC). This exploration researches, utilizing poll survey, the philosophies being utilized in software development in Bengaluru small and medium software firms and portrays a model for incorporating security into the SDLC. The point is to recognize the suitable methods for presenting security gauges a lot prior in the SDLC. The examination recognized different significant components as security measures, arrangements, forms being drilled, and devices utilized inside SDLC ventures. In such manner, proposals and confirmation were accumulated to inspire the real exercises that are suitable to be led at each period of SDLC.

Keywords—*secure software development, secure engineering, software security.*

I. INTRODUCTION

Programming engineers are confronting expanded strain to bring down advancement time, discharge new programming renditions progressively regular to clients and to adjust to a quicker showcase. This new condition powers engineers and organizations to move from an arrangement based cascade advancement procedure to an adaptable lithe procedure. By limiting the pre advancement arranging and rather expanding the correspondence among clients and engineers, the light-footed procedure attempts to make another, increasingly adaptable method for working. This better approach for working permits engineers to concentrate their endeavors on the highlights that clients need. With expanded connect ability and the quicker element discharge, the security of the product item is pushed. To create secure programming, numerous organizations use security building forms that are plan substantial and unbendable. These two methodologies are every others alternate extremes and they legitimately negate one another.

II. LITERATURE SURVEY

The concept of early vulnerability detection is similar to early fault detection. The intention is to detect any anomaly resulting in a vulnerability in the product that would require effort to be corrected after the product has been released to customers. Efforts in detecting a specific type of vulnerability should also be focused in the phase and method that is most cost effective for that type of vulnerability. Studies in early fault detection have shown that detecting the fault earlier in development reduces the development cost (Boehm 2012). Because implementation vulnerabilities are also faults the same benefit should be there when using static tools. However, one characteristic of vulnerabilities is that they are harder to detect than regular faults. It might therefore be necessary to specialize the detection method to specific types of vulnerabilities. In addition, it is likely that for some vulnerability it might be more cost effective to detect them later in development. As an example, a complete manual source code audit with highly specialized security developers should, in theory, detect all implementation vulnerabilities. However, if the source code were larger than a few thousand lines of code the audit would require many experts and be very time-consuming (Porter et al. 2015). As such, it would be economically sounder to use penetration testing on release ready code instead of expanding the implementation phase to incorporate the enormous audit. For an early vulnerabilities detection method or tool it is therefore important to know what types of vulnerabilities are most likely detected and to what cost, then a strategy to detect the vulnerabilities effectively can be created.

Early in the development processes there are security activities such as, Security Requirements that cover both overt functional security and emergent characteristics that are best captured by Abuse Cases and attack patterns. The purpose is to identifying and documenting security and functionality for a given software project. The abuse cases enunciate the behavior of the system under threat; each specifying what can be protected, from whom and the longevity of the same. Role matrices are defined to specify the user roles and their access levels.

ISSN: 2005-4238 IJAST

entry design.

With the end of the design phase the security engineering activities switch the focus towards the product source code. At the code level, the focus is on bugs surfacing during implementation which are detected using static analysis tools, which identify common threats and vulnerabilities. A process should have mandatory Static Code Analyses with predefined rules and priorities. Coding Rules are helpful in clarifying the rationale for the deprecation of unsafe functions and help identify and recommend alternatives for these unsafe functions. While performing source Code Reviews on source code before the code change is committed to the source code repository increase the chance of detecting the vulnerability early (Rigby et al. 2018). An agile concept where developers code in pairs. Solving and reviewing problems directly as the code is written (Williams et al. 2017).

III. METHODOLOGY

In order to understand the current scenario in SMEs in Bengaluru focusing primarily on the security aspects in SMEs in Bengaluru, a well- structured questionnaire was used to get the respondents insight into the same and later was validated through a reliability test. Decision Tree method has been used to understand the relationships between the undertaken variables. Structural equation modeling has been used to identify the important security aspects

IV. RESULTS AND DISCUSSIONS

A. DECISION TREE 1:: GROWING CHAID METHOD

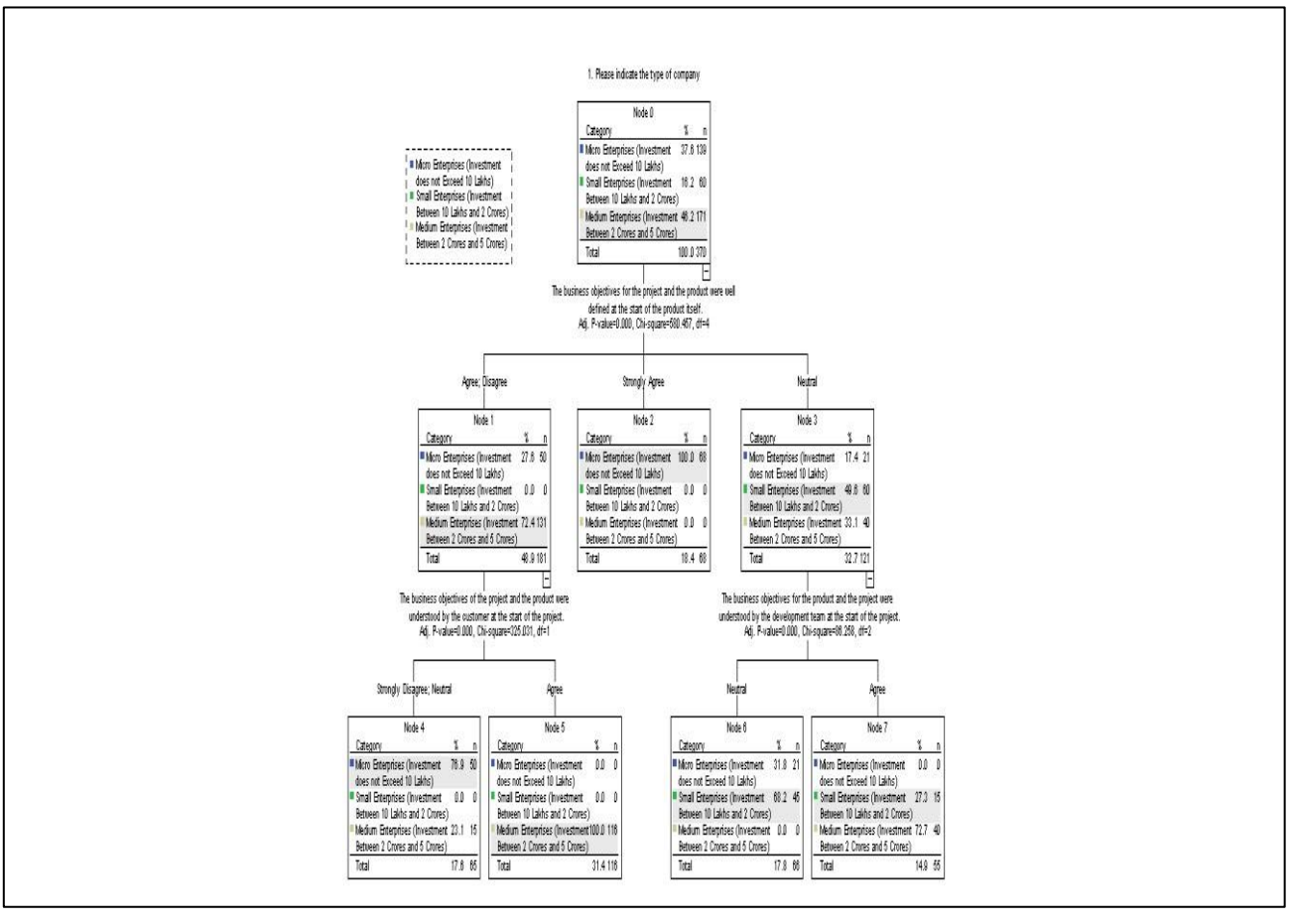
I) DECISION TREE 1: GROWING METHOD: CHAID

DEPENDENT VARIBALE: TYPE OF COMPANY

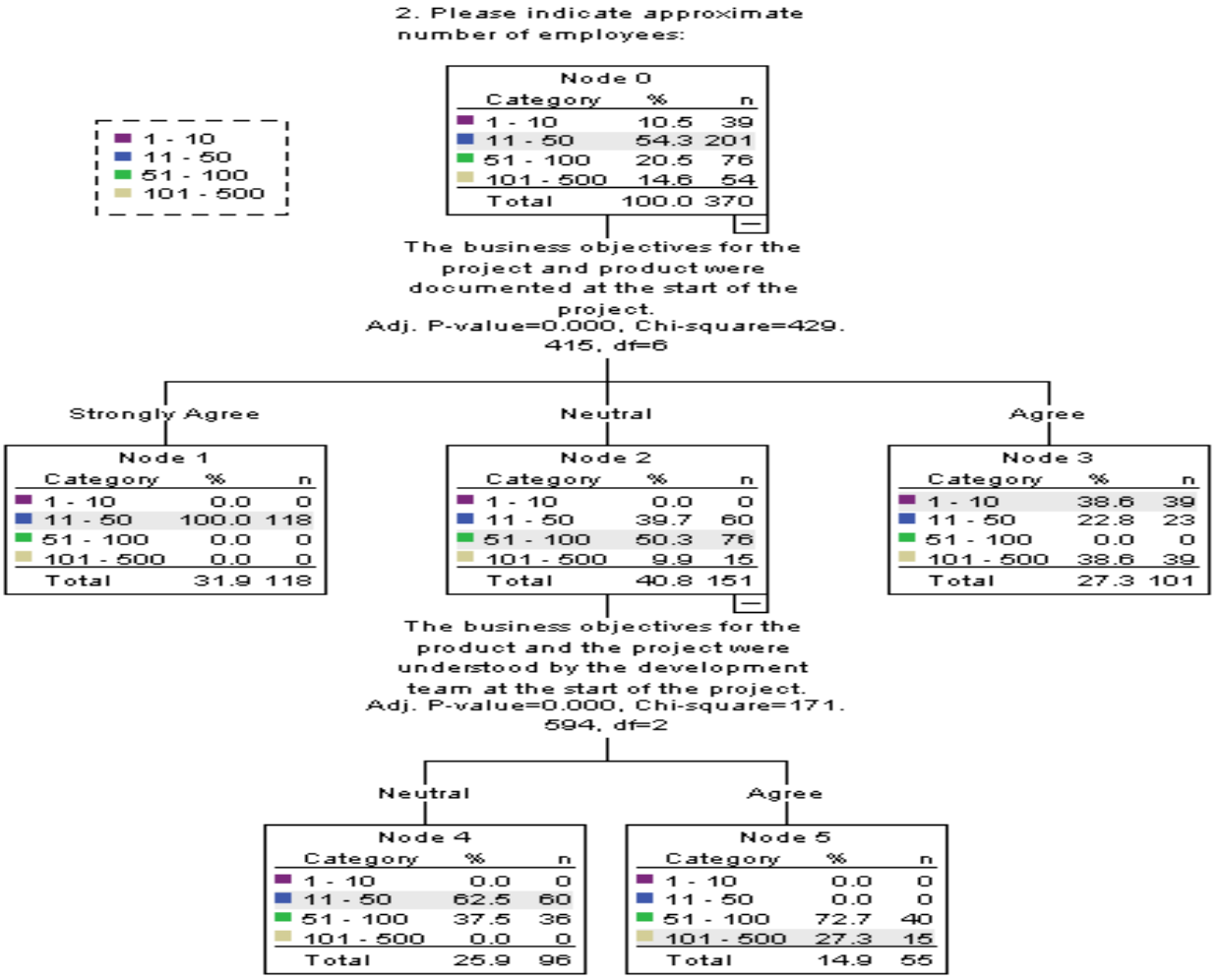
INDEPENDENT VARIABLES:

- The business objectives for the project and the product were well defined at the start of the product itself.
- The business objectives for the project and product were documented at the start of the project.
- The business objectives for the product and the project were understood by the development team at the start of the project.
- The business objectives of the project and the product were understood by the customer at the start of the project.

INFLUENCING VARIABLES: Please indicate average percentage of your turnover to export market



Model Summary		
Specifications	Growing Method	CHAID
	Dependent Variable	1. Please indicate the type of company
	Independent Variables	The business objectives for the project and the product were well defined at the start of the product itself., The business objectives for the project and product were documented at the start of the project., The business objectives for the product and the project were understood by the development team at the start of the project., The business objectives of the project and the product were understood by the customer at the start of the project.
	Validation	None
	Maximum Tree Depth	3
	Minimum Cases in Parent Node	100
	Minimum Cases in Child Node	50
Results	Independent Variables Included	The business objectives for the project and the product were well defined at the start of the product itself., The business objectives of the project and the product were understood by the customer at the start of the project., The business objectives for the product and the project were understood by the development team at the start of the project.
	Number of Nodes	8
	Number of Terminal Nodes	5
	Depth	2



II) DECISION TREE 2: GROWING METHOD: CHAID

DEPENDENT VARIBALE: Average Number of employees

INDEPENDENT VARIABLES:

- The business objectives for the project and the product were well defined at the start of the product itself.
- The business objectives for the project and product were documented at the start of the project.
- The business objectives for the product and the project were understood by the development team at the start of the project.
- The business objectives of the project and the product were understood by the customer at the start of the project.

WE SHOWED THE EFFECTIVENESS AND THE VIABILITY OF THE PROPOSED APPROACH. EIGHT AREAS NEEDED IMMEDIATE AND CORRECTIVE ACTIONS TO MAKE A BALANCE FOR AN EFFECTIVE SOFTWARE DEVELOPMENT PROJECT PLANNING.

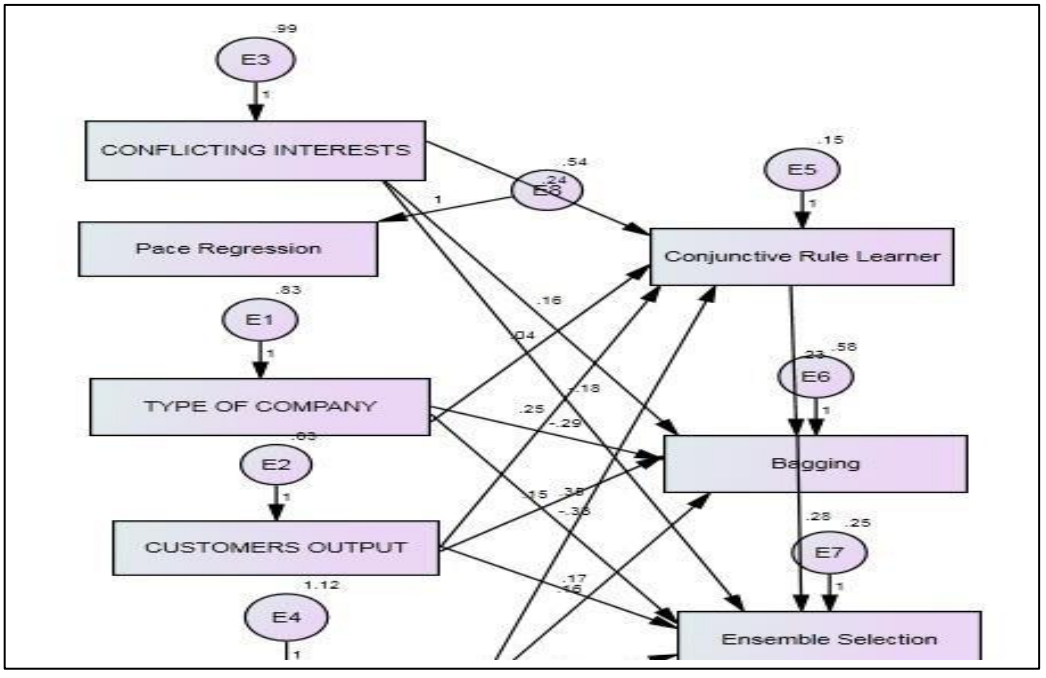
TABLE 1: DETAIL ANALYSIS OF THE CRITICAL LEVEL CLASSIFICATION OF FAILURE MODE IN REQUIREMENT PHASE

Model Summary		
Specifications	Growing Method	CHAID
	Dependent Variable	2. Please indicate approximate number of employees:
	Independent Variables	The business objectives for the project and the product were well defined at the start of the product itself., The business objectives for the project and product were documented at the start of the project., The business objectives for the product and the project were understood by the development team at the start of the project., The business objectives of the project and the product were understood by the customer at the start of the project.
	Validation	None
	Maximum Tree Depth	3
	Minimum Cases in Parent Node	100
	Minimum Cases in Child Node	50
Results	Independent Variables Included	The business objectives for the project and product were documented at the start of the project., The business objectives for the product and the project were understood by the development team at the start of the project.
	Number of Nodes	6
	Number of Terminal Nodes	4
	Depth	2

ISSN: 2005-4238 IJAST

Potential Failure Mode for planning phase	S	O	D	RPN	Critical Level
(1) The chosen software development model is not appropriate to the scope, magnitude and the complexity of the defined task	7	6	3	126	Critical
(2) The chosen model does not fit the complexity of requirements for the project at	7	5	5	175	Critical
(3) The documentation and results obtained by the chosen software model does not reach the concerned management, development or	2	2	3	12	Normal
(4) Specifications for a certain component of the software project are not clearly defined or not correctly documented	7	7	6	294	Critical
(5) The information and documentation about the selected standards, methods, tools and programming language is not enough to have the required tasks completed	6	6	5	180	Critical
(6) The required assignments are not achievable because of the limitations of the	4	3	3	36	Normal
(7) There is no clearly defined hierarchy model for the different type of people using the system	3	4	1	12	Normal
(8) During the development process the schedule made at the beginning fails to determine the	3	4	1	12	Normal
(9) The project fails because of not enough experienced engineers and insufficient abilities of	3	4	1	12	Normal
(10) There is no clearly defined hierarchy model for the different type of people using the system	3	4	1	12	Normal
(11) Cost estimation of the projects is not correctively	9	7	5	315	Critical

STRUCTURE EQUATION MODELLING



Requirements are validated to ensure deliverables quality, attack surface reduction are used to simplify interfaces and to limit

Observed, endogenous variables Q69, Q70,Q71,Q1,Q29,Q15,Q33,Q53
Unobserved, exogenous variables E5,E6,E7,E1,E2,E3,E4,E8

Variable counts (Group number 1)

Number of variables in your model: 16
Number of observed variables: 8
Number of unobserved variables: 8
Number of exogenous variables: 8
Number of endogenous variables: 8

	Weights	Covar	Var	Means	Inter cepts	Total
Fixed	8	0	0	0	0	8
Labeled	0	0	0	0	0	0
Unlabeled	14	0	8	0	0	22
Total	22	0	8	0	0	30

Number of distinct sample moments: 36
Number of distinct parameters to be estimated: 22
Degrees of freedom (36 - 22): 14

			Estimate	S.E.	C.R.	P	Label
Q69	<---	Q1	.036	.022	1.600	.110	
Q69	<---	Q29	.247	.022	11.072	***	
Q69	<---	Q15	.236	.020	11.531	***	
Q69	<---	Q33	.355	.019	18.454	***	
Q70	<---	Q69	.225	.101	2.225	.026	
Q70	<---	Q1	-.287	.044	-6.578	***	
Q70	<---	Q29	.152	.050	3.021	.003	
Q70	<---	Q15	.163	.046	3.501	***	
Q70	<---	Q33	.155	.052	2.993	.003	
Q71	<---	Q29	.175	.029	5.995	***	
Q71	<---	Q15	-.177	.027	-6.577	***	

Model	RMR	GFI	AGFI	PGFI
Default model	.249	.659	.123	.256
Saturated model	.000	1.000		
Independence model	.282	.468	.316	.364

Model	PRATIO	PNFI	PCFI
Default model	.500	.229	.228
Saturated model	.000	.000	.000
Independence model	1.000	.000	.000

Model NCP LO 90 HI 90

Model	NFI Delta1	RFI rho1	IFI Delta2	TLI rho2	CFI

Default model	.457	-.085	.461	-.087	.457
Saturated model	1.000		1.000		1.000
Independence model	.000	.000	.000	.000	.000

Variable Summary (Group number 1)
Your model contains the following variables (Group number 1)

Model	NCP	LO 90	HI 90
Default model	912.622	816.517	1016.118
Saturated model	.000	.000	.000
Independence model	1679.688	1547.942	1818.800

Model	FMIN	F0	LO 90	HI 90
Default model	2.511	2.473	2.213	2.754
Saturated model	.000	.000	.000	.000
Independence model	4.628	4.552	4.195	4.929

Model	RMSEA	LO 90	HI 90	PCLOSE
Default model	.420	.398	.444	.000
Independence model	.403	.387	.420	.000

Model	AIC	BCC	BIC	CAIC
Default model	970.622	971.722	1056.719	1078.719
Saturated model	72.000	73.800	212.886	248.886
Independence model	1723.688	1724.088	1754.996	1762.996

Model	ECVI	LO 90	HI 90	MECVI
Default model	2.630	2.370	2.911	2.633
Saturated model	.195	.195	.195	.200
Independence model	4.671	4.314	5.048	4.672

Model	HOELTER .05	HOELTER .01
Default model	10	12
Independence model	9	11

Minimization:	.018
Miscellaneous:	.383
Bootstrap:	.000
Total:	.401

CONCLUSION

Different software engineering approaches are followed for the design and development of software that includes the spiral model, waterfall model, agile methods and iterative approaches. These are efficient software engineering approaches, but security is neglected part and requires special consideration. Therefore, all these

approaches needs security blend to make secure software engineering. In this research paper we have made an attempt to understand various security practices which were adopted in SMEs , It has been observed that there needs to be planning stage security enhancements to avoid loop holes at a later stages.

Reference

[1] Pareek, P. K., Nandikolmath, T. V., & Gowda, P. (2012). FMEA Implementation in a Foundry in Bangalore to Improve Quality and Reliability.
International Journal of Mechanical Engineering and Robotics Research, 1(2), 81-87.

[2] Nandikolmath, T., Pareek, P. K., & SA, V. K. (2012). "Implementation of a Lean model for carrying out value stream mapping in manufacturing
industry", International Journal of Mechanical Engineering and Robotics Research, 1, no. 2, 88-95.

[3] Nandikolmath, T., Pareek, P. K., & SA, V. K. (2012). "Implementation of a Lean model for carrying out value stream mapping in manufacturing
industry", International Journal of Mechanical Engineering and Robotics Research, 1, no. 2, 88-95.

[4] Piyush Kumar Pareek , Dr.Praveen Gowda , et al 'Ergonomics in a Foundry in Bangalore to improve productivity',International Journal of Engineering
and Social Science , ISSN: 2249- 9482 ,Volume 2,Issue 5 (May 2012) , pp 1-6.

[5] Sandhya Soman and Dr. Piyush Kumar Pareek 2019 J. Phys.: Conf. Ser. 1427 012010

[6] PK Pareek ,Optimization of the critical parameters affecting the fuel lid performance ,– 2019, 2019-28-2413

[7] Dr. Piyush Kumar Pareek, ROC Structure Analysis of Lean Software Development in SME's Using Mathematical CHAID Model (May 17, 2019) [8] KD Menon, A Raj Jain, Dr.Piyush Kumar Pareek – 2019 , Quantitative Analysis of Student Data Mining ,

[9] A Pai, VS Veesam, BS Babu, Piyush Kumar Pareek , SIX SIGMA APPROACHES USED IN IMPLEMENTING IN SUPPLY CHAIN MANAGEMENT: A REVIEW , Research and Applications of Web Development and Design 1 (2), 12-16

[10] Piyush Kumar Pareek , Questionnaire Survey using CHI-Square Test in Six Sigma in SME's in Bengaluru" , Advancement in Image Processing and
Pattern Recognition 1 (1), 20-24

[11] Piyush Kumar Pareek , Analysing Tools in Six Sigma in SME's in Bengaluru , Journal of Advancement in Software Engineering and Testing 1 (1), 19-
29

[12] Piyush Kumar Pareek, SIX SIGMA APPROACHES USED IN IMPLEMENTING IN SUPPLY CHAIN MANAGEMENT: A REVIEW , Journal of
Advancement in Software Engineering and Testing 1 (1), 14-18

[13] A Pai, VS Veesam, Piyush Kumar Pareek, BS Babu , Challenges in SME's ANOVA ANALYSIS PART-2 in Bengaluru , Research and Reviews: Advancement in Robotics 1 (1), 9-15

[14] A Pai, VS Veesam, BS Babu, Piyush Kumar Pareek, ANOVA Analysis Part One of Challenges in SME'S in Bengaluru , Research and Reviews: Advancement in Robotics 1 (1), 1-8

[15] Piyush Kumar Pareek, ANOVA Analysis Part One of Challenges in SME'S in Bengaluru , Research and Reviews:, K Swathi, P Shetteppanavar , An efficient machine translation model for Dravidian language , 2017 2nd IEEE International Conference on Recent Trends in Electronics

[16] PK Pareek , An adoptive Model for lean software development in small and medium level firms in Bengaluru,2016

[17] KV Rao, R Balakrishna, HA Pai, Piyush Kumar Pareek , Data Mining for Healthy Tomorrow with the Implementation of Software Project
Management Technique , Artificial Intelligence and Evolutionary Computations in Engineering Systems

[18] Piyush Kumar Pareek Dr. AN Nandakumar,'Lean software development Survey on Benefits and challenges in Agile and Lean usage in small and medium level firms in Bangalore' ,International Journal of Advanced Research in Computer Science and Software

[19] Piyush Kumar Pareek , Identifying Wastes in software , , International Journal of Engineering Studies and Technical Approach

[20] Failure Mode Effective Analysis of Requirements Phase in small software Firms', Paper ID: ICSTM

[21] PK Pareek, DAN Nandakumar , YMCA/2015/292, International Conference on Science

[22] Mr.Suhas G K, etal. "An Exploration on Recommendation Based Interactivity through Multiple Platforms in Big Data." IOSR Journal of Computer
Engineering (IOSR-JCE), 22.1 (2020), pp. 31-36.