LEAN INITIATIVES IN SOFTWARE DEVELOPMENT PROJECTS

Ph.D. Thesis

RAMKARAN YADAV

ID: 2012RME9032

DEPARTMENT OF MECHANICAL ENGINEERING

MALAVIYA NATIONAL INSTITUTE OF TECHNOLOGY JAIPUR

December 2019

Lean Initiatives in
Software Development Projects

Submitted in

fulfillment of the requirements for the degree of

Doctor of Philosophy

By

RAMKARAN YADAV

ID: 2012RME9032

Under the supervision of

Prof. M. L. Mittal and Prof. Rakesh Jain

DEPARTMENT OF MECHANICAL ENGINEERING

MALAVIYA NATIONAL INSTITUTE OF TECHNOLOGY JAIPUR

December 2019

Respectfully Dedicated to

My Parents, My Wife, My Daughter, My Son, Siblings

&

My Teachers

MALAVIYA NATIONAL INSTITUTE OFTECHNOLOGY
JAIPUR – 302017 (RAJASTHAN) INDIA

Declaration

I, **RamKaran Yadav**, declare that this thesis titled, **“Lean Initiatives in Software Development Projects”** and the work presented in it, are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this institute.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this institute or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself, jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: December 2019

RamKaran Yadav
(2012RME9032)

MALAVIYA NATIONAL INSTITUTE OFTECHNOLOGY
JAIPUR – 302017 (RAJASTHAN) INDIA

Certificate

This is to certify that the thesis entitled **"Lean Initiatives in Software Development Projects"** being submitted by **Mr. RamKaran Yadav (ID No.: 2012RME9032)** is a bonafide research work carried out under my supervision and guidance in fulfilment of the requirement for the award of the degree of **Doctor of Philosophy** in the Department of Mechanical Engineering, Malaviya National Institute of Technology, Jaipur, India. The matter embodied in this thesis is original and has not been submitted to any other University or Institute for the award of any other degree.

Prof. M. L. Mittal
(Supervisor)
Professor
Department of
Mechanical Engineering,
MNIT, Jaipur
India

Prof. Rakesh Jain
(Co-Supervisor)
Professor
Department of
Mechanical Engineering,
MNIT, Jaipur
India

Place: Jaipur.
Date: December 2019

Acknowledgement

It is a matter of great pleasure and privilege for me to present this thesis entitled **"Lean Initiatives in Software Development Projects"**.

First, I want to thank the God for all the blessings and for giving the opportunity to fulfil my dreams. Next, I take this opportunity to express my deep sense of gratitude and respect to my supervisors **Prof. M. L. Mittal** and **Prof. Rakesh Jain** for their patience, motivation, enthusiasm, and immense knowledge. Their excellent guidance, advice, feedback and encouragement have been fundamental to the completion of this project.

Besides my supervisors, I would like to thank my DREC members: Prof. A.P.S. Rathore and Dr. Gunjan Soni for their insightful comments, constructive inputs, and encouragement given during PhD progress presentations. I am also grateful to Prof. Dilip Sharma (Head, Mechanical Engineering Department) for his help and motivation during the study.

My sincere thanks also go to my friends Dr. Vinod Yadav, Dr. Manish Kumar, Mr. Dheeraj Joshi, Mr. Manoj Kumar Pal, Mr. Aniruddh Waghire, Mr. Ganesh Tanpure, Mr. Govind Singh Yadav, Dr. Manu Augustine and Mr. Dnyaneshwar Ghode for their continuous moral support and help through their knowledge in some dimensions of my work. I also extend sincere thanks to other fellow PhD scholars and all those people, whom I have not mentioned, but they had helped me directly or indirectly during the work.

Last, but by no means the least, I would like to thank my parents and my wife Smt. Manju Yadav, my daughter Jigyasa Yadav, my son Heemaj Yadav and siblings for their continuous support, encouragement and sacrifices for this endeavour that has taken such a large amount of my time. My PhD would not have been possible without their inseparable support, prayers and sacrifices. Especially, I am indebted to my wife who stood by me for the successful completion of this work. It would be an injustice to express her support, patience and sacrifices in few words.

RamKaran Yadav

2012RME9032

ABSTRACT

In recent years, software development (SD) has become a centre stage of industrial development and an important part of the economy. Shapiro (2014) revealed that the contribution of the software industry in the overall GDP of the USA was 3.2 percent ($526bn) in 2012. Bhatnagar (2006) estimated that the software and service sector accounts for over 20 percent of India's total exports and 2.6 percent of GDP. SD industry is under immense pressure to deliver software at a rapid pace along with the highest quality and lowest cost. Lean has high potential as one of the possible solutions to problems faced by the SD industry. However, a fundamental shift is required in SD projects if lean principles are to be incorporated in their businesses to create value by reducing waste.

A plethora of literature is available on the adoption of lean in manufacturing, but it is relatively scant for SD projects. The set of lean principles/practices used in manufacturing have already been well established, but it is not so for SD projects. As manufacturing and SD projects have different characteristics, it needs more research to identify as to which lean principles/practices are suitable for implementation in SD projects. There have been some case studies on lean in SD projects, but very few empirical studies are observed. As generalization is usually an issue with case studies, a more empirical investigation is required. Moreover, there is a dire need to examine the effects of management initiatives on organizational performance.

In this research work, a literature review was carried out in lean software development (LSD), and potential gaps were identified in this area. To fill these gaps, three research objectives were formulated, which are as follows: (i) To perform an exploratory study to investigate the implementation of lean in projects, (ii) To identify drivers and barriers of implementing lean in projects and (iii) To develop a model for lean implementation in Software Development projects. The purpose of the present research work is not solely to examine the suitability of particular lean practices/principles, but also to investigate the appropriateness of underlying core principles that drive the adoption of lean in SD projects and challenges in lean adoption. Further, this research is an attempt to improve organizational performance in software development

organizations by determining the extent to which lean thinking can be adopted as an appropriate lean principle/practice for practitioners and project managers.

Five exploratory case studies and surveys were utilized as a part of the research methodology. In the first phase, exploratory case studies were conducted to get preliminary insights of the adoption level of lean in software development projects. Subsequently, surveys were carried out in a large number of SD organizations for validation of the conceptual model.

It was found from exploratory studies that three types of barriers exist in the adoption of lean. These are categorized as customer-related, process-related, and people-related barriers. The major barriers found in lean adoption are uneven requirement flow and change requests, lack of communication and coordination within the team and among the cross-functional team, and lack of information sharing. Based on the exploratory study, a model was developed for implementing lean principles in SD projects to help experts of IT. Using the stated model, the research was accomplished to provide empirical evidence for the relationship of lean principles, lean thinking, and performance determinants. This is deemed to be helpful for SD organizations in improving their desired performance levels.

In the manufacturing sector, all core lean principles (value, value stream, flow, pull, and perfection) are applied, while in SD organizations, value, responsiveness, and flow are found to be more visible. Investigations within a larger set of SD companies revealed that value, flow, and responsiveness are more visible. Only a little impact of VSM (Value stream mapping), pull and perfection was visible in SD organizations. This implies that lean is only partially applicable in SD projects as opposed to the full level of applicability as found in the manufacturing sector. The present research may also help organizations to develop strategies for improving lean implementation in SD projects by focusing on these principles. Research reveals that value and flow are the most relevant lean principles for SD projects. Thus, the software industry interested in lean implementation should focus on the implementation of value and flow enhancing practices.

Value enhancing practices include mistake proofing, team-based problem-solving, VSM, standardized work, continuous improvement, and visual control. Some of the flow enhancing practices are flexible and cross-functional teams, Kanban/Kanban board, JIT information, cause and effect matrix, and set base concurrent development. These practices can be implemented according to the nature of the organization, viz. product-based, and project-based SD organizations. Value enhancing practices are useful in the improvement of products and processes. Flow enhancing practices contribute to the reduction of WIP and also help in early detection of bottlenecks in a dynamic environment. The Lean practices are also effective in bug tracking, testing, and to reduce release time and supporting software process improvements.

Lean can be implemented in SD organizations by using a set of guidelines. There is sufficient literature to safely conclude that value enhancing practices should be the first step in SD organizations. The core value enhancing practices are "waste elimination practices," such as mistake proofing, visual control, continuous improvement, and VSM. It should be followed by systematic implementation of flow enhancing practices such as work standardization, Kanban/Kanban Board, JIT information, and set-based concurrent development. There may be more practices that can be included depending upon the characteristics and performance improvement objectives of an SD organization. Future research will include efforts in clearly identifying these practices.

It is recommended that value-oriented practices, flow-oriented practices, and responsiveness measure should be significantly focused upon for performance improvement. Finally, it can be concluded that application of lean principles is positively associated with the operational performance of SD projects.

Contents

Declaration	i
Certificate	ii
Acknowledgement	iii
Abstract	iv
Contents	vii
List of Figures	xiii
List of Tables	xv
Abbreviations	xviii

	Title	**Page No.**
Chapter 1 : Introduction		**1-9**
1.1	Background	2
1.2	Research motivation	4
1.3	Research questions and objectives	6
1.4	Research process overview	6
	1.4.1 Literature review	6
	1.4.2 Selection of SD organization for exploratory case studies and issue in lean adoption	7
	1.4.3 Analysis of exploratory case studies	7
	1.4.4 Development of survey instrument	7
	1.4.5 Survey in SD Projects	7
	1.4.6 Filtrations of survey data and generation of survey results	8
	1.4.7 Development of model establishing relation between lean principles, lean thinking and operational performance	8
	1.4.8 The guidelines for lean adoption	8
1.5	Outline of the thesis	8

Chapter 2: Literature Review **10-41**

2.1	Background	11
2.2	Defining Lean	11
2.3	Evolution of Lean	12
2.4	Lean Adoption in Projects	14
2.5	Lean Adoption in Software Development Projects	16
	2.5.1 Literature on Lean in Different Projects	16
	2.5.2 Year wise publications on Lean in software development projects	17
	2.5.3 Paper published based on research Methodology adopted in LSD	17
2.6	Lean Implementation in SD Projects	19
2.7	Lean Principles in SD Projects	21
	2.7.1 Value	22
	2.7.2 Value Stream	22
	2.7.3 Flow	22
	2.7.4 Pull	23
	2.7.5 Perfection	23
2.8	Responsiveness in SD Projects	23
2.9	Lean Practices in SD Projects	23
	2.9.1 Mistake proofing	25
	2.9.2 Value Stream Mapping (VSM)	25
	2.9.3. Standardized work	26
	2.9.4. Visual control	26
	2.9.5. Visual management	26
	2.9.6. Kanban	26
	2.9.7. Kanban Board	26
	2.9.8 Design Structure Matrix (DSM)	26
	2.9.9. JIT Information	27
	2.9.10. Flexible and cross functional team	27
	2.9.11. Cause and effect matrix	27
	2.9.12. Set base concurrent development	27
	2.9.13. Continuous improvement/Kaizen	27

	2.9.14 Team-based problem solving	28
	2.9.15 Heinjunka	28
2.10	Drivers and Barriers of Lean implementation in SD Projects	28
	2.10.1 Drivers for Lean implementation in SD Projects	28
	2.10.1.1 Team Involvement/ Employee involvement	29
	2.10.1.2 Business Value with top management support	29
	2.10.1.3 Rapid Delivery as part of performance management	29
	2.10.1. 4 Mistake proofing for quality into the product	29
	2.10.1. 5 Customer focus/ Customer Satisfaction	30
	2.10.1.6 Coordination and communication	30
	2.10.2 Barriers inhibiting Lean implementation in SD projects	30
	2.10.2.1 Customer related barriers	32
	2.10.2.2 Process related barriers	32
	2.10.2.3 People related barriers	34
2.11	Impact of Lean on performance of SD Projects	38
2.12	Lean in Manufacturing (LM) v/s Software Development (LSD)	38
2.13	Research Gaps	40
2.14	Summary	41
Chapter 3: Research Design and Methodology		**42-51**
3.1	Introduction	43
3.2	Research Plan	43
	3.2.1. Exploratory Case Studies	44
	3.2.2. Hypotheses Formulation	45
	3.2.2.1 Hypotheses related to Drivers of lean implementation	45
	3.2.2.2 Hypotheses related to Barriers in Lean Implementation	45
	3.2.2.3 Hypothesis related to status of lean in SD projects	46
3.3	Survey Research	47
	3.3.1 Development of Survey Instruments	47
	3.3.2 Questionnaire Design	47
	3.3.3 Pilot Study	48
	3.3.4 Sample Design	48

	3.3.5 Administration of Survey	48
3.4	Data Analysis	49
3.5	Structural Model testing	50
Chapter 4: Exploratory Case Studies		**52-81**
4.1	Introduction	53
4.2	Plan for conducting exploratory case studies	54
4.3	Data Collection	55
4.4	Description of cases	56
	4.4.1 Software Development Stages in Product -based Organizations	57
	4.4.2 Software Development Stages in Project -based Organization	58
4.5	Product-based organizations	60
	4.5.1 Case 1: ProdBased1	60
	4.5.2 Case 2: ProdBased2	63
4.6	Project- based Organizations	66
	4.6.1 Case3: ProjBased1	66
	4.6.2 Case 4: ProjBased2	69
	4.6.3 Case 5: ProjBased3	71
4.7	Case discussion and Analysis	74
	4.7.1 Level of adoption of lean principles in software development projects	74
	4.7.2 Level of adoption of lean practices in software development projects	75
	4.7.3 Cross case comparisons	76
	4.7.4 Issues in implementing lean in software development	78
4.8	Summary	80
Chapter 5: Descriptive Analysis		**82-110**
5.1	Introduction	83
5.2	Data Coding	83
	5.2.1 Missing Value analysis	87
	5.2.2 Outliers	87

	5.2.3 Non response bias	87
5.3	Survey Observations	88
5.4	Responses rate and Respondent's characteristics	88
5.5	Responses of adoption of lean principles in SD organizations	94
	5.5.1 Response Analysis of enablers of value principles (VL)	94
	5.5. 2. Response analysis of enablers of Value Stream (VS)	94
	5.5 3. Response analysis of enablers of flow (FL)	95
	5.5.4 Response analysis of enablers of Pull (PL) principle	95
	5. 5.5 Response Analysis of enablers of Perfection (PF)	96
	5.5.6 Response Analysis of enablers of Responsiveness (RS)	96
5.6	Response Analysis of enablers of Operational performance (OP)	97
5.7	Drivers of Lean adoption	98
5.8	Barriers in adoption of lean in SD projects	101
5.9	Overview of adoption Level of Lean Principles	105
5.10	Summary	109
Chapter 6: Impact of Lean Principles on performance		**111-147**
6.1	Introduction	112
6.2	Development of Model constructs	112
6.3	Choice of model estimation Method and Model Indices	113
6.4	Model Fit Indices	114
6.5	Factor Analysis	114
	6.5.1 KMO and Bartlett's Test	115
	6.5.2 Eigen Value	115
	6.5.3 Factor Loading and Rotation	118
	6.5.4 Internal consistency analysis	120
6.6	Structural Equation Model	121
6.7	One Factor Congeneric Model	122
	6.7.1 One Factor measurement model for value principles (VL)	122
	6.7.2 One factor measurement model for value stream principles (VS)	124
	6.7.3 One factor measurement model for Flow principles (FL)	126
	6.7.4 One factor measurement model for Pull Principle (PL)	127

6.7.5 One factor measurement model for Perfection (PF) 129
6.7.6 One factor measurement model for Responsiveness (RS) 130
6.7.7 One factor measurement model for Operational Performance (OP) 132
6.8 Multifactor measurement model for confirmatory factor analysis 133
6.8.1 First order measurement mode 133
6.8.2 Reliability and validity of measurement models 134
6.8.3 Construct Validity 138
6.8.4 Co-variance among constructs 139
6.8.5 Second Order Measurement Model 141
6.9 Multifactor congeneric model to analyze the impact of lean principles on performance Measure 141
6.10 Discussion of hypotheses among the Research Constructs 145
6.11 Conclusion 147
Chapter 7: Conclusion and Discussion 148-155
7.1 Background 149
7.2 Summary of the Research 149
7.3 Concluding Remarks 151
7.4 Theoretical Contribution 151
7.5 Research Implications 152
7.6 Practical Implications 152
7.7 Set of recommendations: guidelines in implementing lean in software development Projects 153
7.8 Managerial Implications 154
7.9 Research Limitations and Recommendations for future research 155
References 156-170
Appendices 171-179
Appendix I : Case Study Questionnaire 171
Appendix II: Lean Assessment Questionnaire 173
Appendix III: List of Publications 178
Appendix IV: Author's Biographical Sketch 179

List of Figures

Figure 2.1:	Levels of Lean	12
Figure 2.2:	Evolution of lean in various sectors	14
Figure 2.3:	Percentage of papers in lean in various projects	16
Figure 2.4:	Publications in SD Projects	17
Figure 2.5:	Paper Published based on Methodology	18
Figure 2.6:	Software Development Steps and Waste Identification	20
Figure 3.1:	Research Plan	44
Figure 4.1:	Stages of Product Based SD Organization	58
Figure 4.2:	Stages of Project Based SD Organization	59
Figure 4.3:	Relationship between level of adoption and lean practices	75
Figure 5.1:	Types of software Organization	90
Figure 5.2:	Types of software Services	90
Figure 5.3:	Types of software based on domain	91
Figure 5.4	Types of IT Support Services	91
Figure 5.5:	Types of deliverables in projects	92
Figure 5.6:	Positions of employees	92
Figure 5.7:	Number of employees	93
Figure 5.8:	Annual turnover	93
Figure 5.9:	Responses on adoption level of Value	94
Figure 5.10:	Responses on adoption level of Value Stream	95
Figure 5.11:	Responses on adoption level of Flow principles	95
Figure 5.12:	Responses on adoption level of Pull	96
Figure 5.13:	Responses on Value Stream	96

Figure 5.14:	Responses on Responsiveness	97
Figure 5.15:	Responses on Operational performance	97
Figure 6.1:	Conceptual Model	113
Figure 6.2:	Scree plot	118
Figure 6.3:	CFA results for Value principles	123
Figure 6.4:	CFA results for Value Stream principles	125
Figure 6.5:	CFA results for Flow principles	126
Figure 6.6:	CFA results for Pull principle	128
Figure 6.7:	CFA results for perfection principle	129
Figure 6.8:	CFA results for Responsiveness	131
Figure 6.9:	CFA results for Operational Performance	132
Figure 6.10:	CFA Diagram for constructs	136
Figure 6.11:	SEM Diagram	139
Figure 6.12:	Multi structural Equation Model for operational performance	142
Figure 6.13:	Relationship among lean principles, lean thinking and operational performance	144

List of Tables

Table 2.1:	Evolution of lean	13
Table 2.2:	Objective of Lean Implementation in Various Projects	15
Table 2.3:	Lean Principles in SD Projects	22
Table 2.4:	Literature according to the Lean Practices used in SD Projects	24
Table 2.5:	Major Barriers in SD projects	31
Table 2.6:	Comparison between Lean Manufacturing (LM) and Lean Software Development (LSD)	39
Table 3.1:	Summary of model fit indices	50
Table 4.1:	Plan for Conducting Exploratory Case Studies	54
Table 4.2:	Lean Assessment of Case1 (ProdBased1)	62
Table 4.3:	Lean Assessment of case 2 (Prod Based2)	65
Table 4.4:	Lean assessment of Case 3 (ProjBased 1)	68
Table 4.5:	Lean Assessment of Case 4 (ProjBased 2)	70
Table 4.6:	Lean Assessment of Case 5 ((ProjBased 3)	73
Table 4.7:	Level of adoption of lean principles	74
Table 4.8:	Value Enhancing Lean Practices	77
Table 4.9:	Flow Enhancing Lean Practices	78
Table 4.10:	Comparison of case organizations for issues	80
Table 5.1:	Coding of variables and references	83
Table 5.2:	Questionnaire with specific lean principles/practice	86
Table 5.3:	Demographic profile for the respondents	88
Table 5.4:	Types of drivers with mean and standard deviation	98
Table 5.5:	t- test for significant drivers in adoption of lean	99

Table 5.6:	Ranking of drivers in lean implementation in SD projects	100
Table 5.7:	Barriers in adoption of lean with reliability analysis in SD projects	101
Table 5.8:	t- Test for significant barriers in SD Projects	103
Table 5.9:	Ranking of barriers in adoption of lean in SD projects	104
Table 5.10:	Reliability analysis of lean principles	105
Table 5.11:	t test for important lean principles	107
Table 5.12:	Ranking of lean principles adopted in SD projects	109
Table 6.1:	Acceptance Limit of Model fit Indices	114
Table 6.2:	KMO and Bartlett's Test	115
Table 6.3:	Extraction sum of squared loadings	116
Table 6.4:	Varimax Factor Rotated Component Matrix	119
Table 6.5:	Reliability of constructs	121
Table 6.6:	Regression weights for Value principle	123
Table 6.7:	Acceptance limit of model fit Indices	124
Table 6.8:	Model fit Indices for value (VL)	124
Table 6.9:	Regression weights for value stream principles	125
Table 6.10:	Model Fit Indices for value stream (VS) principle	125
Table 6.11:	Regression weights for Flow (FL) principles	127
Table 6.12:	Model fit indices for flow (FL) principle	127
Table 6.13:	Regression weights for Pull (PL) principle	128
Table 6.14:	Model fit indices for pull (PL) principle	129
Table 6.15:	Regression weights for Perfection (PF)	130
Table 6.16:	Model fit indices for Perfection (PF)	130
Table 6.17:	Regression weights for Responsiveness (RS)	131
Table 6.18:	Model fit indices for responsiveness (RS)	132

Table 6.19	Regression weights for operational Performance	133
Table 6.20:	Model fit indices for Operational performance	133
Table 6.21:	CFA results for measurements model	137
Table 6.22:	Summary of Factor matrices for higher level constructs	138
Table 6.23:	Covariance of constructs	140
Table 6.24:	Pearson Correlations between constructs	141
Table 6.25:	Model fit indices of constructs	143
Table 6.26:	Result of hypothesized relationship in structural model	145

Abbreviations

SD	Software Development
LSD	Lean Software Development
VSM	Value Stream Mapping)
NVA	Non -Value Activities
SEM	Structural Equation Modelling
TPS	Toyota Production System
LSD	Lean Software Development
LM	Lean Manufacturing
PM	Project Manager
CFA	Confirmatory Factor Analysis
VCB	Visual Control Board
DSM	Design Structure Matrix
SCM	System Complexity Measures
SPC	Statistical Process Control
VSM	Value stream mapping
CSF	Critical Success Factors
PD	Product Development
CRs	Change Requests
QE	Quality Engineering
KIVP	Knowledge Innovation Visible Planning
PMS	Project Management System
SRD	Software Requirement Documentation
TFS	Team Foundation Server
CFD	Cumulative Flow Diagrams
MIS	Management Information System
DSM	Design Structure Matrix
SBCD	Set Based Concurrent Development
CFT	Cross-functional teams
VL	Value
VS	Value Stream
FL	Flow
PL	Pull
PF	Perfection
RS	Responsiveness
OP	Operational performance

MLE	Maximum Likelihood Estimation
ULS	Un-Weighted Least Square
SLQ	Scale Free Least Square
ADF	Asymptotically Distribution Free
GLS	Generalized Least Square
GFI	Goodness of Fit Index
NFI	Normed Fit Index
CFI	Comparative Fit Index
RMSEA	Root Mean Square Error of Approximation
KMO	Kaiser-Meyer-Olkin
AVE	Average Variance Extracted
CR	Composite reliability
R^2	Squared multiple correlations

Chapter 1

Introduction

Chapter 1

Introduction

1.1 Background

In recent years, software development (SD) has come to the centre stage of industrial development and has become an important part of the global economy. Shapiro (2014) revealed that the contribution of the software industry in the overall GDP of USA was 3.2 per cent ($526bn) in 2012. Bhatnagar (2006) estimates that the software and service sector account for over 20 percent of India's total exports and 2.6 per cent of GDP. According to NASSCOM (National Association of Software and Services Companies) analysis reports it is observed that IT sector's share in 2015-16 has increased to 9.3 per cent of GDP (Singh and Kaur, 2017). Likewise, Lo and Liu (2009) argue that India occupies about 75 percent of the global market for outsourced software. SD industry is under immense pressure to deliver software quickly, with the highest quality and lowest cost. Lean may be one of the possible solutions to problems faced by the SD industry. Therefore, a fundamental shift is required in SD projects by introducing lean principles into their procedures to create value by reducing waste.

The term ***lean*** was invented by the research team working on the International Motor Vehicle Programme at Massachusetts Institute of Technology to reflect both the waste reduction of Toyota production system (TPS) and to contrast it with craft and mass forms of production (Womack *et al.,* 1990). Ohno developed lean manufacturing management principles at Toyota (Womack *et al.*, 1990). Stone (2012) states that "lean" did not remain confined to the automobile sector only, where it originated. As of today, it has gained applicability across numerous industries and sectors.

Lean has been viewed in different ways by researchers. For example "Lean is about cutting off all efforts that customer does not want to pay for; and only by really addressing the customer's exact wants in terms of quality, delivery and cost, to create processes that truly satisfy and delight the customer" (Mehta *et al.*, 2008). According to Hines *et al.* (2004) lean existed both at strategic and operational level. The strategic aspect includes customer value and value creation.

At the operational level, lean tools are used to eliminate waste. Hallam (2003) opines that we need to view lean not as an abstract philosophy but as one which includes both philosophy and practices/tools. These tools are closely associated with the lean principles (Stone, 2012; Saurin *et al.,* 2011; Psomas *et al.*, 2018). Arlbjorn *et al.* (2008) suggests three levels of lean implementation viz. Philosophy, Principles and Tools and Techniques.

In recent years, there has been a significant growth in project undertakings across different sectors and industries. With increasing numbers of new developments, new initiatives are being pursued through projects. Projects are temporary production systems. When those systems are structured to deliver the product while maximizing value and minimizing waste, they are said to be 'lean' projects. Lean is implemented in projects in order to develop waste reduction capability and to exploit value generation through increasing customer value by reducing running cost, reaching performance targets and improving facility outputs. Middleton and Joyce (2012) observed in a case study that performance of software development improved in terms of greater business value with a focus on creating the highest value for the customer by reduction in defects and reduced lead time to delivery by adopting a lean approach. SD projects nowadays are emphasizing more on customer centric development, which translates into both reducing cost and increasing efficiency and effectiveness by eliminating waste and adding value.

The skills of the IT project manager may quickly become obsolete if they are not updated (Dalcher and Brodie, 2007). Many projects fail to deliver on scope, time or budget, or they deliver a product that does not match the client's needs (Sauser *et al.*, 2009). To be able to study the impact of market and operational & structural factors on projects, it is necessary to assess the implementation and use of lean principles within SD project. To remain competitive, organizations are trying out the principles and concepts of lean thinking (Anand *et al.*, 2014).

A plethora of empirical studies as well as case studies on lean implementation have been carried out in various sectors such as manufacturing (Abreu-Ledon *et al.*, 2018); process industries (Panwar *et al.*, 2015, 2018); small and medium enterprises (Yadav *et al.*, 2019); aerospace (Crute *et al.*,2003); product development (Liker and Morgan, 2006; Nepal *et al.*, 2011); construction (Saieg *et al.*, 2018); service (Malmbrandt and Åhlström, 2013; Kundu and Manohar, 2016) and agri-business organizations (Satolo *et al.*, 2017). However, its applications have not been explored much in SD projects. There is thus a need for identification of lean principles/practices that are suitable for SD projects and the issues that hinder its implementation in SD projects. With this objective, this research is taken up through exploratory case studies.

From literature review, a number of lean principles were identified, but no single research instrument was found that captured this broad array of practices for projects. The primary objective of this research is to identify lean tools and principles and level of adoption of these practices within projects. Research as well as practitioner literature was reviewed to identify a complete set of lean principles and existing research instruments that might be helpful for this study. Most studies have looked at the relevance of lean in manufacturing system. So far, researchers working with lean principles and their applications have devoted scant attention to SD projects.

1.2 Research Motivation

The adoption of lean production in the service sector is a more recent manifestation of the use of lean principles (Stass *et al.*, 2011). However, the utility and impact of such ideas in non-manufacturing contexts remains a contentious issue. Ebert *et al.* (2012) revealed that some researchers claim that principles from other fields can't apply to a creative and design-oriented discipline such as software development. It remains unclear how software companies that are moving towards lean, interpret and implement Lean Software Development (LSD) in practice (Rodríguez *et al.*, 2013). On the other hand, researchers argue that lean principles have universal applicability (Sousa and Voss 2001; Stass *et al.*, 2011). This is one of the major motivations for the present research work.

Frequent changes, high speed, uncertainty and complexity are becoming the characteristics of many projects (Atkinson *et al.*, 2006; Winter *et al.*, 2006). In contrast with physical products, software development has unique features such as high design cost, very complex logic and intangible design. It is not obvious how to translate production principles (that are otherwise well understood) to the software industry. Therefore, this research work aims at investigating the applicability of lean in software development.

Apart from the general ambiguity surrounding the applicability of lean in SD, it is also not clear which lean principles/practices are more suitable for SD projects. There is thus a need to identify lean principles/practices that are applicable to SD projects and the issues/barriers that hinder their implementation in SD projects. With this objective, exploratory case studies were conducted as a part of the current research work in various SD organizations.

The set of lean principles/practices used in manufacturing have already been well established, but it is not so for SD projects. As manufacturing and SD projects have different characteristics; it needs more research to identify as to which lean principles/practices are suitable for implementation in SD projects.

Only a few (e.g. Stass *et al.*, 2011; Kundu and Manohar, 2016) empirical studies have been reported about lean in SD projects. Therefore, more empirical studies are needed to investigate implementation of lean principles/practices and relation between lean implementation and operational performance of SD projects. Such studies would surely provide a sound basis for further empirical research on adoption of principles in software development projects. There have been some case studies on lean in SD projects; but very few empirical studies are observed. As generalization is usually an issue with case studies, more empirical investigation is required.

Both research papers and practitioners' works were examined to identify a comprehensive set of practices that can be considered to be essential in lean projects; and to identify survey instruments that could be used to complete a broad assessment of the lean practices in SD projects.

It is always required to examine the effects of management initiatives on organizational performance. The literature is literally lacking any work that directly or indirectly measures the effects of lean initiatives on organizational performance in SD projects. The present study is the first attempt at filling this research gap and investigates various aspects and issues in lean adoption in SD projects.

1.3 Research Questions and Objectives

RQ1. How is lean interpreted in SD projects?

RQ2: Which lean principles/practices are prevalent in SD projects?

RQ3: What are the drivers and barriers of lean implementation in SD projects?

RQ4: Does implementation of lean affect the organizational performance in SD projects?

To respond to these questions**,** the following research objectives have been articulated:

1. To perform exploratory study to examine the implementation of lean in SD projects.
2. To identify drivers and barriers of implementing lean in SD projects.
3. To develop a model for lean implementation in Software Development projects

The major contribution of this research is deep investigation of lean adoption in SD projects. The outcomes of the present study will be useful for practitioners keen on taking up lean initiatives in SD projects. This research also provides empirical evidences for relationship of lean principles, lean thinking and performance determinants, which may help SD organizations in achieving desired performance levels. The outcome of the model formulated in this study has confirmed that lean principles are strongly enabled with lean thinking, which directly affects organizations' performance. The ultimate aim of this study is to create opportunities for project manager to improve operational performance of SD projects.

1.4. Research Process overview

In this section, research approach overview is presented to achieve the stated objectives of the present research work. Exploratory case studies and survey methodology has been adopted. Five case studies were carried out to garner a good idea of lean in SD organizations. The overview of steps adopted in this research approach is as follows:

1.4.1 Literature review

A literature review was carried out to determine if existing lean methods could be applied in SD projects. The review was useful in identifying specific characteristics and challenges in SD projects and other issues mentioned in the literature. It provided the knowledge of lean tools, technology and principles that have been adopted or have potential application in SD organization. The review also helped in achieving a clear distinction between lean manufacturing (LM) and lean software development (LSD).

1.4.2 Selection of SD organizations for exploratory case studies and issue in lean adoption

Case organizations were selected on the basis of level of customization in the software development. Case studies were carried out to explore awareness of lean, lean practices, lean principles in SD organizations. The main research focus was on identifying how the lean tools/principles may be used in SD organizations through exploratory study.

1.4.3 Analysis of exploratory case studies

Data was collected through interviews, observation, and by collecting relevant documents such as face to face semi-structured interviews with project manager (PM) using a questionnaire. Interviews were conducted on field visits and companies' progress reports, meetings, project proposals and unique documents were all studied.

1.4.4 Development of survey instrument

Development of survey Instrument based on literature and findings from exploratory cases was designed after collecting feedback from academicians and practitioners' from software/IT in a pilot test. This study was conducted using a sample of SD projects demonstrating how the instrument can be used to understand factors that may limit the adoption of lean practices.

1.4.5 Survey in SD Projects

Survey questionnaire explored the awareness of lean in SD projects, principles, drivers and barriers in lean adoption. A large scale survey questionnaire was distributed in product as well as project SD organizations. It also explored the perception about lean initiatives in SD organizations.

1.4.6 Filtrations of survey data and generation of survey results

The responses were filtered statistically and summarized to generate the survey results. SPSS 18.0 and IBM AMOS 26.0 were used to generate survey results.

1.4.7 Development of model establishing relation between lean principles, lean thinking and operational performance

A model was developed using structural equation modelling (SEM) to establish the relationship between lean principles, lean thinking and operational performance.

1.4.8 The guidelines for lean adoption

Guidelines were developed for adoption of lean principles in SD projects based on model developed and outcome of case studies.

1.5 Outline of the thesis

Chapter 1 covers overview of study and background of lean, SD projects, the research motivation, and the objective of the current study. This chapter also focuses on the stages involves in the research design including research process overview.

Chapter 2 focuses on the literature review in terms of levels of lean, lean practices and lean principles, drivers for lean implementation and major barriers/challenges in lean implementation in SD projects. It also summarizes key differences between lean manufacturing (LM) and lean software development (LSD).

Chapter 3 delineates the research methodology which is adopted to carry out the current research and also includes research design. It covers development of questionnaire, pre-testing of questionnaire through pilot study, sample design, data collection and data analysis techniques. Further, exploratory case studies and survey based methodology is adopted to address the research issue in adoption of lean in SD projects. Based on gap identified from literature, certain hypotheses were formulated to examine the status and extent of lean adoption in SD organizations. In survey methodology, a survey instrument is developed based on IT experts and administered to a sample of 1151 SD organizations.

Chapter 4 focuses on the first objective of current research, which covers the exploratory study to examine the implementation of lean in five SD organizations which are classified as product and project based organizations. Exploratory case studies are carried out to explore status of lean in terms of adoption of lean practices and lean principles in SD projects. It also emphasizes on the issues/ challenges in implementing lean in SD projects. Findings of case studies were used to develop the survey questionnaire.

Chapter 5 is devoted to the second objective of the current study and examines the adoption level, drivers, and the various barriers in adoption of lean in SD projects. It also includes data collection methods and descriptive analysis of data collected from survey including response analysis of all search constructs. This chapter covers test of reliability, normality, validity and analysis of demographic profile for the respondents.

Chapter 6 addresses the third objective of the research which is to develop a model for lean implementation in software development projects. It also includes Structural Equation Modeling (SEM) for analysis. The validity of construct and model fit indices were evaluated by Confirmatory Factor Analysis (CFA) for measurement. This model demonstrates the relationship between lean principles, lean thinking and operational performance of SD projects.

Chapter 7 concludes the thesis. This chapter concludes the contribution of the present research on lean adoption in SD projects and practical implications of the findings thereof. Further, lean implementation guidelines are formulated, which comprise steps of lean adoption in SD projects. The limitations of the present research work along with scope and directions for future research are also presented.

Chapter 2

Literature Review

Chapter 2

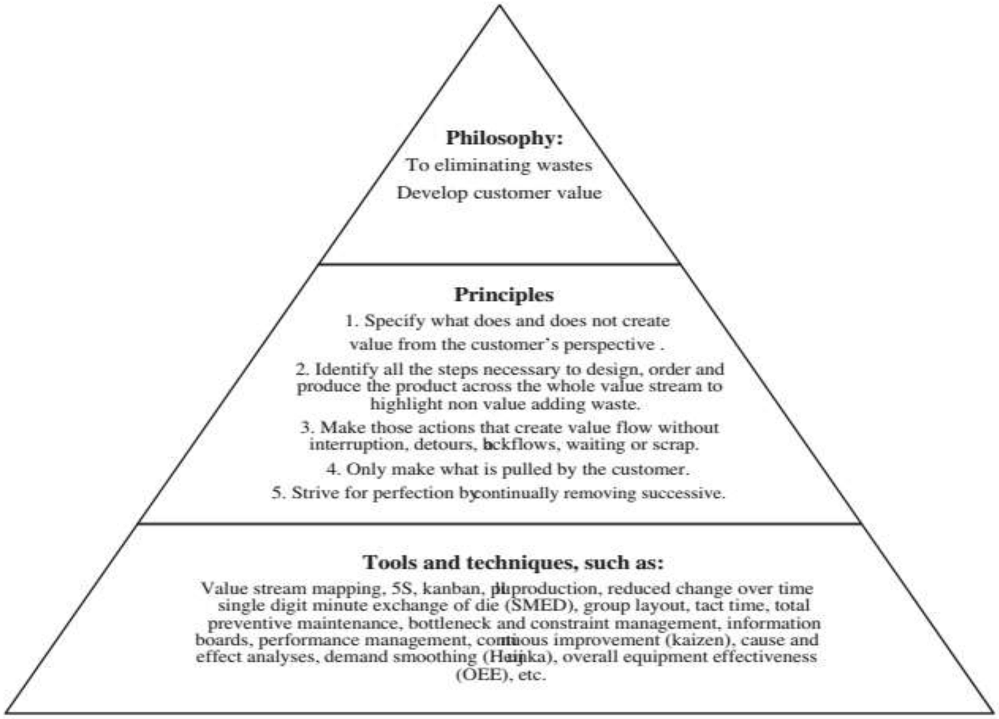
Literature Review

2.1 Background

This chapter focuses on a state-of-the art review of lean adoption in software development projects. The objective of this review is to recognize explicit characteristics and challenges of SD projects with a attention on software industries and to analysis the lean tools and techniques that have been adopted or have potential applications in the software development organizations and barriers in implementing lean in SD projects. A brief introduction about lean and its application in manufacturing, other sectors of economy including other type of projects is first provided. Then the literature on lean initiatives in SD projects is given.

2.2 Defining Lean

Lean has been viewed in different ways by researchers. For example "Lean is about cutting off all efforts that customer does not want to pay for; and only by really addressing the customers's exact wants in terms of quality, delivery and cost, to create processes that truly satisfy and delight the customer" (Mehta *et al.*, 2008). Womack and Jones (1996); Hines *et al.* (2004) have considered lean as a philosophy of eliminating waste and creating value. According to Hines *et al.* (2004) lean existed both at strategic and operational level. The strategic aspect includes customer value and value creation. At the operational level, lean tools are used to eliminate waste. Hallam (2003) opines that we need to view lean not as an abstract philosophy but one which includes both philosophy and practices/tools. These tools are closely associated with the lean principles (Stone, 2012; Saurin *et al.,* 2011; Psomas *et al.*, 2018). Arlbjorn *et al.* (2008) suggested three levels: Philosophy, Principles, Tools and Techniques of lean implementation. Figure 2.1 provides an overview of these three levels.



Three levels of lean (adopted from Arlbjørn, Freytag, and Damgaard (2008)).

Figure 2.1: Levels of Lean

The top-most level can be viewed as the philosophical level which expresses the core concern of lean to eliminate wastes (Shah and Ward 2007). The second level creates specific production related principles (Hines 2004). The five core lean principles given by Womack and Jones (1996) are value, value stream, flow, pull and continuous improvement, flow, pull and (perfection) continuous improvements. The third level can be viewed as the operational level which is the group of lean practices encouraging the activities required to accomplish the objectives as defined in various lean principles (Hines *et al.*, 2004).

2.3 Evolution of Lean

Lean philosophy was first applied in 1980s in Toyota Motor Company in Japan with an objective to reduce waste. Following the positive performance of Toyota, western manufacturers showed great interest to adopt lean production concepts (Womack *et al.*, 2007). They also applied some structural parts of lean production systems and shop-floor techniques. However, it was challenging and sometimes difficult to adopt lean directly because the mindset and organizational culture in Japan was totally different from the western one (Holweg and Pil, 2001; Mehri, 2006). Since 1990s, irrespective of geographical differences, the application of lean and its principles have spread and have been adopted in diverse industrial sectors across the globe. Its focus is getting beyond shop-floor control and improvements, and now includes dimensions such as, customer value, value stream, flow, etc (Womack and Jones, 2000). In recent past lean has also been applied to service and project environments. Table 2.1 provides a chronological evolution of lean.

Table 2.1: Evolution of lean

S. No.	Lean Stage	Year	Author
1.	Vehicle Assembly	1960	Ohno (1988), Shingo (1981, 1988),Keisuke Arai,1992
2	Supply chain	1970	Lamming (1993), Macbeth et.al. (1994), Womack et.al. (1994,
3.	Shop Floor	1980	Ohno, 1988; Schonberger, 1982; Shingō and Dillon, 1989
4.	Beyond shop-floor control	After 1990	Womack and Jones, 2000
5.	Non- manufacturing (NPD, Software Development)	After 2000	Achanga *et al.*, 2012; Gopinath and Freiheit, 2012; Houshmand and Jamshidnezhad, 2006; Lu *et al.*, 2011; Singh *et al.*, 2006) Liker and Morgan, 2006, Stone, 2012(PD)
6.	Public and Service		Arlbjorn and Freytag, 2013
7.	Service Call Centre		Piercy and Rich, 2009a, 2009b
8.	IT support service environment		Kundu and Manohar2016

Evolution of lean concept started from engine manufacturing in 1950. It has been transformed in vehicle assembly, supply chain, and shop floor in 1980. Later, after 1990 its application has been seen extended beyond shop floor. After 2000, it has spread into non- manufacturing sectors, services. Subsequently, the applications of lean were investigated in knowledge area and projects especially in software development projects. Figure 2.2 depicts the evolution of lean in various sectors.

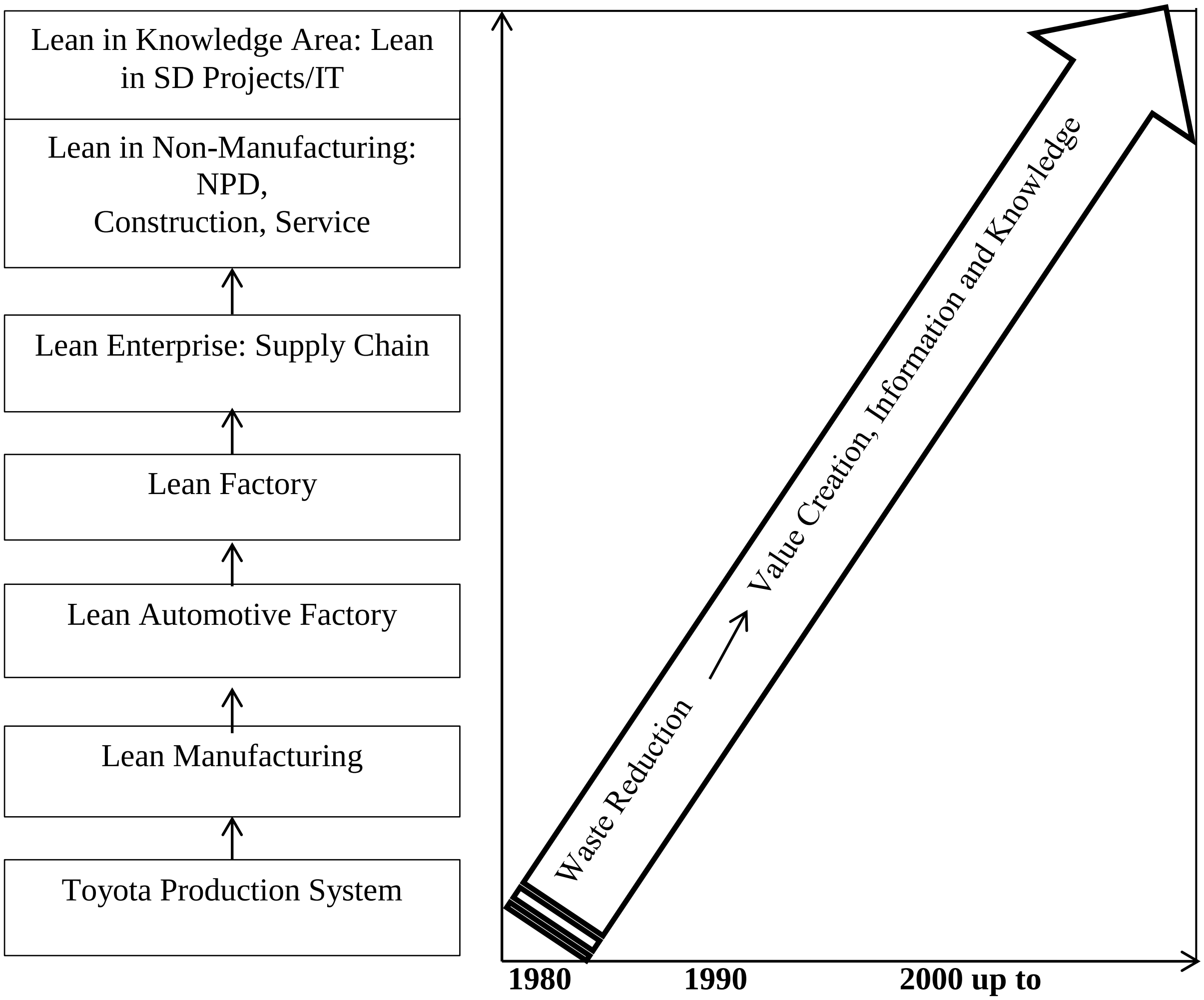


Figure 2.2: Evolution of lean in various sectors

2.4 Lean Adoption in Projects

Motivated by the success of lean implementations in Manufacturing lean is now being implemented in projects. These lean initiatives in projects have been due to variety of objectives. Table 2.2 provides the major objectives for lean implementation in different types of projects.

Table 2.2: Objectives of Lean Implementation in Various Projects

Types of projects	Author(s)	Reasons of implementing lean
Construction	Ballard (2007)	To deliver value to specific customer with less waste of time
	Ballard(2008)	To improve Lean Project Delivery System
	Bryde et. al.(2012)	To improve the construction process primarily by reducing waste and maximizing value in building refurbishment projects.
	Kovvuri *et al.*(2016)	To make project delivery more effective and give the genuinely necessary impulse to the Indian construction sector
New Product Development (NPD)	Gautam *et al.*(2008)	To maximize customer perceived value through design change in product
	Hines (2006)	A value adding system of operations across firms to provide a series of dependent value plan to end customer
	Bro¨ring *et al.* (2008)	For value creation
	Nepal *et al.* (2011)	To determine the root causes of wasteful reworks for improving the NPD Process
Software Services	Staas *et al.*(2011)	To eliminate the waste and increasing business value
Information system	Kundu and Manohar (2011)	To identify and categorize waste activities in IT support services
IT support services	Malladi *et al.*(2011)	To identify those areas which insert waste into IT service area and also some best practices in lean approach appropriate to IT service delivery.
	Kundu *et al.* (2011)	To propose a set of underlying drivers appropriate for implementation of lean principles in the IT support services enterprises.
Software Development	Middleton (2012)	To examine how the lean ideas can be applied to software project management. The exploratory case study focused on software development team.
	Ebert *et al.* (2012)	To establish a foundation and facilitate alignment on what “lean” means within Software development by proposing a framework
	Pernstål *et al.* (2013)	For software process improvement (SPI)
	Anand *et al.* (2014)	To recognize different wastes and suggest distinctive lean practices to re-engineer the business activities of an Indian software company and to improve software development processes.

2.5 Lean Adoption in Software Development Projects

This section presents a review of scholarly articles on lean adoption in software development projects.

2.5.1 Literature on lean in different projects

A total of 1200 papers focusing on lean in different types of projects have been considered in this review. Out of these 41% papers are conceptual in nature and 59% focuses on lean adoption in different types of projects viz. construction, product development/new product development, software development and services, agriculture and healthcare (figure 2.3). As can be seen only 4% papers have been devoted to lean in SD projects and Services thus a very little has been done towards lean in software development.

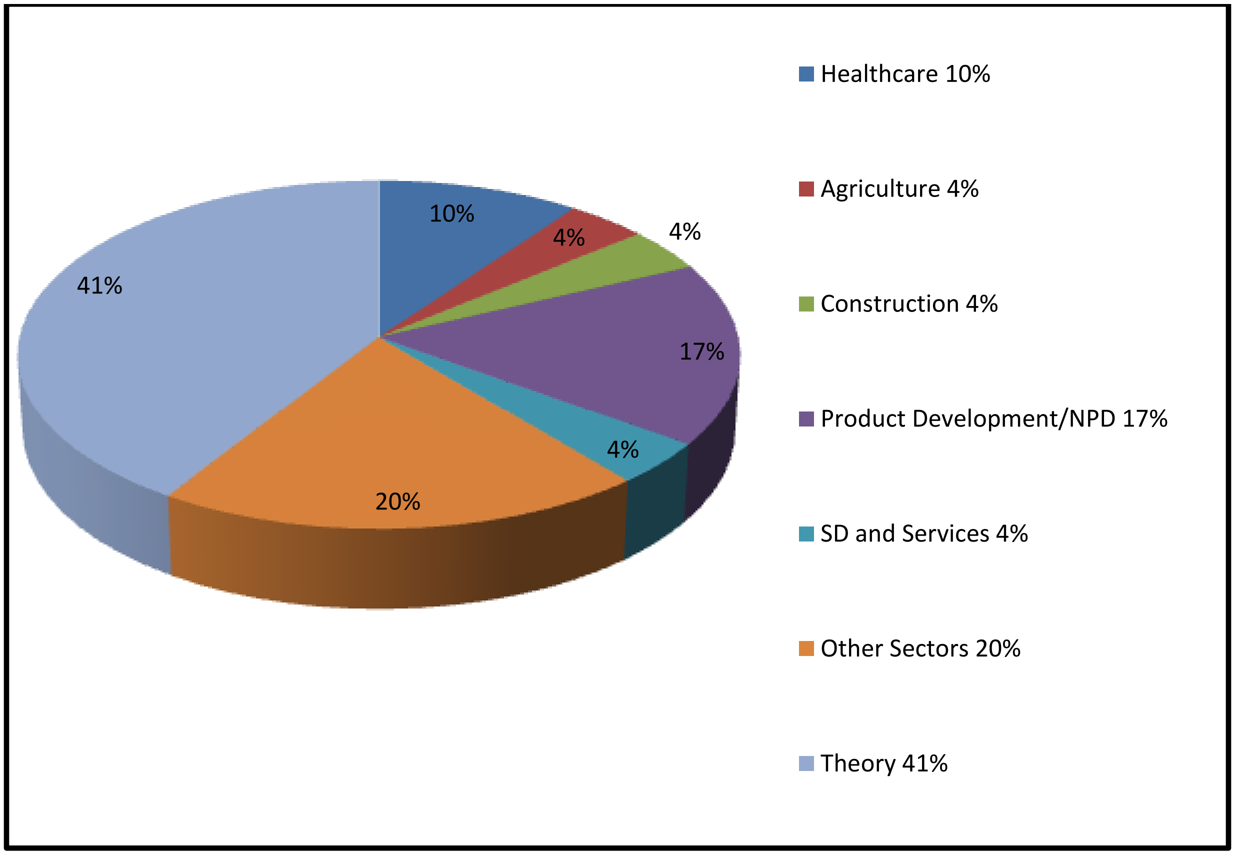


Figure 2.3: Percentage of papers in Lean in various Projects

2.5.2 Year wise publications on lean in Software development projects

For this, 48 articles have been taken from the reputed international journals: Information and Software Technology, IEEE, Journal of Change Management**,** International Journal of Project Management, Journal of Operations Management (JOM), International Journal of Operations and Production Management (IJOPM), Journal of Enterprise Transformation, International Journal of Lean Six Sigma, Journal of Systems and Software, Software Quality Journal, Information and Software Technology, Journal of Systems and Software, Journal of Information and Knowledge Management System and conferences published during 1993-2019. For searching Lean, various keywords: lean, leagile, LSD, Lean thinking, Lean Software development, Lean in projects were used. Figure 2.4 depicts year wise publications in the area of lean in software development projects.

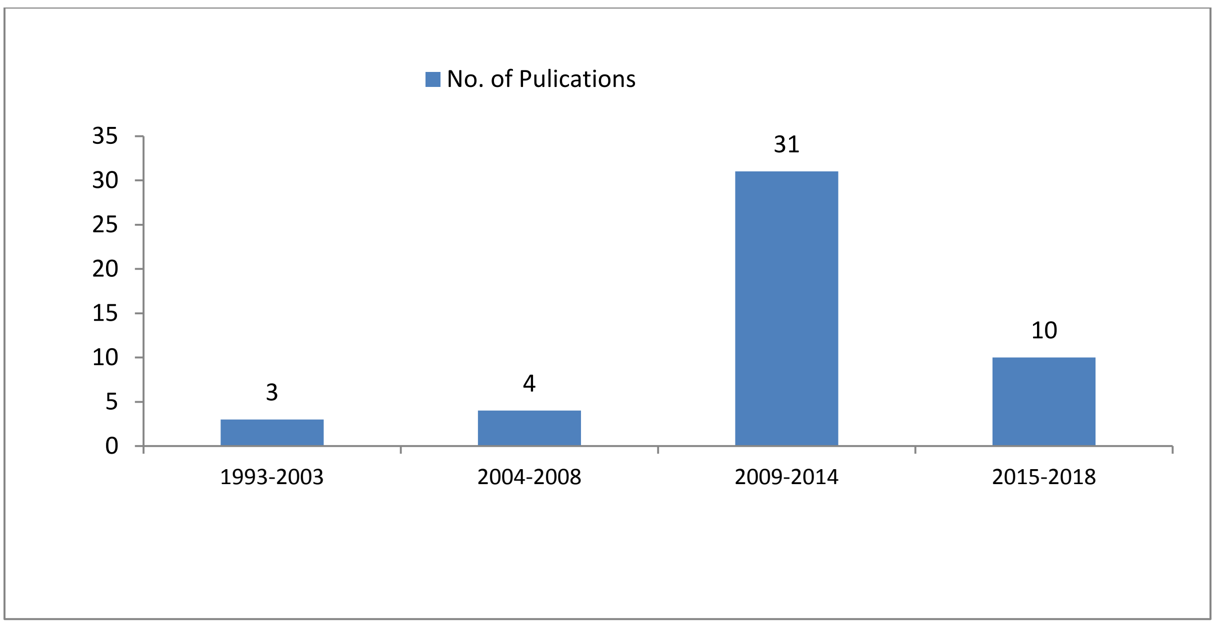


Figure 2.4: Publication in SD Projects

2.5.3 Paper published based on research methodology adopted in LSD

Kupiainen *et al.*, (2015) revealed that the earliest study began from 2002 in agile and lean SD projects, and the remainder of the research is equally distributed from 2002 to 2013. The single-case study was the most utilized research methodology (60%), then experience report (23%), multi-case studies contributed (10%) and only 7% empirical study was considered. Figure 2.5 depicts paper published based on methodology adopted in lean software development.

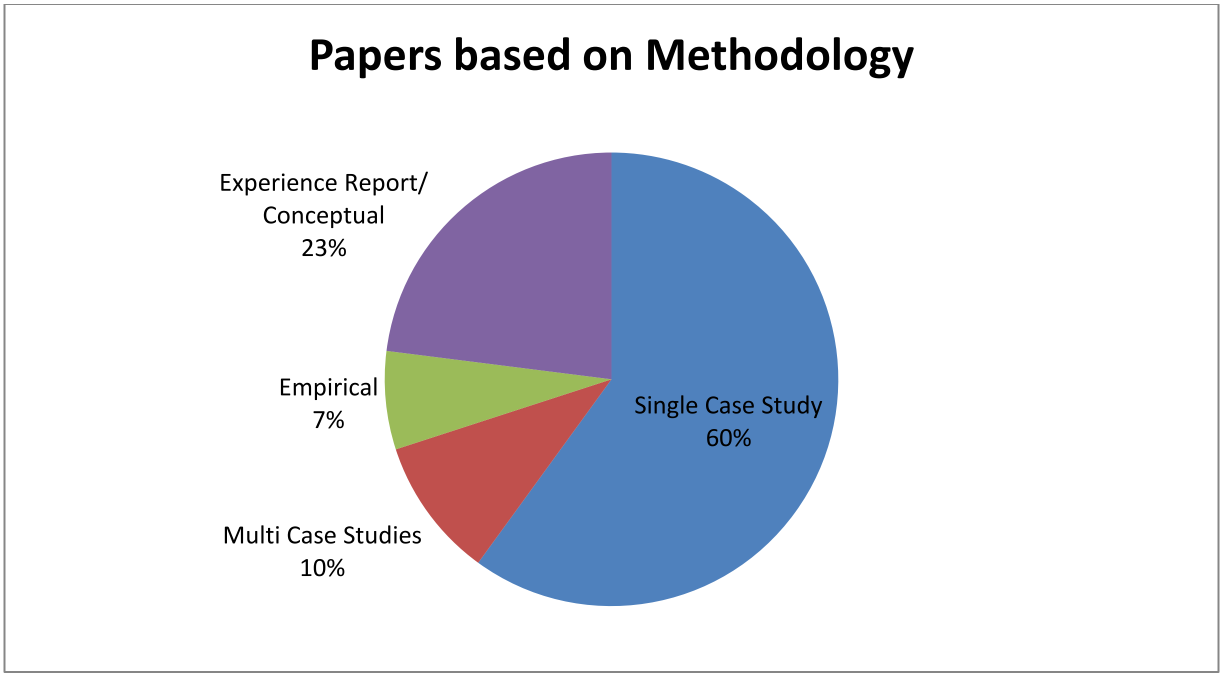


Figure 2.5: Paper published based on methodology

Traditionally, lean has been applied to manufacturing with a focus to remove waste by simultaneously minimizing supplier, customer and internal variability (Shah and Ward, 2007). In context of SD, Mehta *et al.* (2008) advocate their perception about Lean as follows: "Lean is about cutting off all efforts that customer does not want to pay for; and only by really addressing the customer's exact wants in terms of quality, delivery and cost, will we be able to create processes that truly satisfy and delight the customer". Lean has been implemented in various software companies related to R&D software, software services, virtual worlds company, software maintenance, inkjet and laser printers and Other IT companies. The lean application has also been observed in sub area of software: testing, maintenance, design and SD Life cycle. Lean software development (LSD) includes operational tools and techniques, customer defined values, requirements, new features (new codes), bug fixing (defect or error) or addition of new features and continuous flow processing.

Further, Pernstål *et al.* (2013) found that out of the total publications on lean in SD 76% papers were non-empirical while 24% were empirical. A variety of research methods have been used by researchers. Out of these the case study method (Pernstål *et al.*, 2013; Malladi *et al.*, 2011; G .K. Kundu *and* Manohar 2011; and Staas *et al.* 2011) has been used to a maximum extent with a emphasis on the impact of lean principles on performance of projects. Reich *et al.* (2014) conducted empirical study and also proposed a theoretical model. It was empirically tested the relationships between lean thinking in the model for achieving business value from IT-enabled business projects. Yadav *et al.* (2018) revealed that Lean software development (LSD) includes lean operational tools and techniques, customer defined values, new features (new codes), bug (defect or error) fixing or addition of new features and continuous flow processing.

Literature reveals that agile methodologies have been implemented in SD projects (Dingsøyr *et al.*, 2018; Kupiainen *et al.*, 2015 and Brhel *et al.*, 2015). Agile has focused more on "just in time" by incorporating changes along with quick response and not on automated early detection and elimination of defects (waste). The main focus of the current study is the identification and removal of waste in SD projects.

2.6 Lean Implementation in SD Projects

Lean is implemented in projects in order to develop waste reduction capability and exploit value generation. This is achieved by increasing customer value, reducing running cost, reaching performance target and improving facility outputs. Elimination of waste has been considered to be the main focus for adoption of lean in software development projects.

Ikonen *et al.* (2010) acknowledged some wastes in SD such as partial work done, which is equivalent to in-process inventory in lean manufacturing (LM). Likewise, extra processes and extra features were found equivalent to overproduction. Wastes in SD project are also identified by Mujtaba *et al.* (2010) as shown in Figure2.6. Poppendieck and Poppedieck, (2007) also discussed wastes such as partial work done, motion of requirements, task switching, extra features and defects.

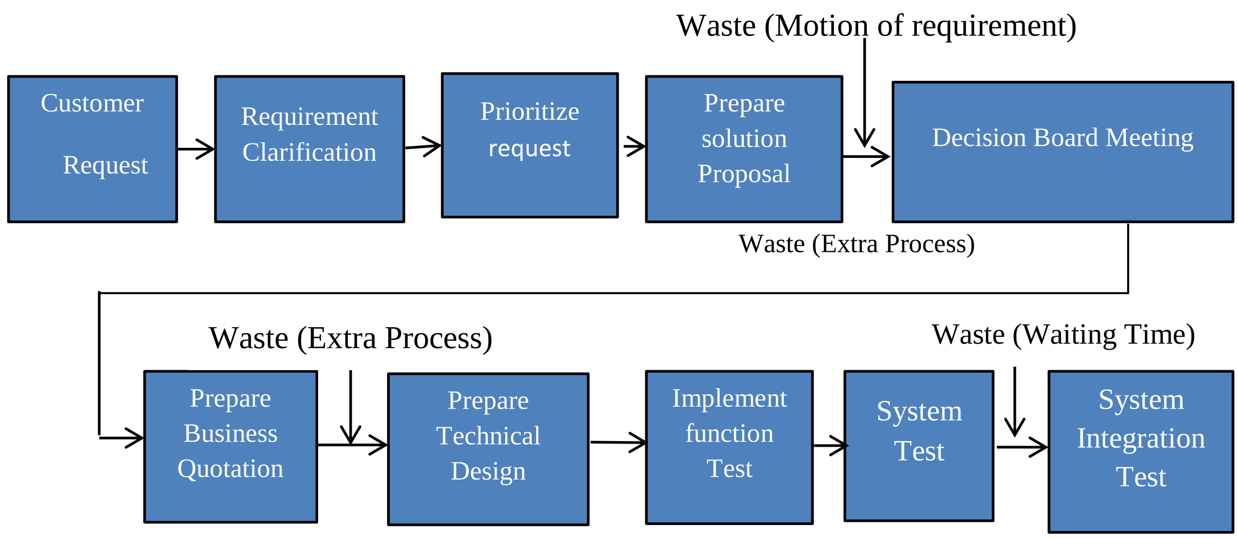


Figure 2.6: Software Development Steps and Waste Identification (Adapted from Mujtaba and Petersen, 2010)

Petersen (2012) examined lean indicators to identify waste in the software maintenance process through a case study. Pernstål *et al.* (2013) identified and classified state of art in large scale software development influenced by lean product development approach. The main contribution of this study focused on recognizing and reducing waste through discovery and assessing inventories of SD requirements, VSM and retaining flow in the SD process.

In SD, lot of time is wasted in detection of errors through testing (Mujtaba and Petersen, 2010). Widman *et al.* (2010) explained the adoption of lean in an animation firm and identified waste activities in SDP. Likewise, Kundu and Manohar (2011) identified and categorized waste activities in IT support services as defects, ineffective communication, external quality enforcement, processing inefficiencies, lack of system disciplines, waiting, over processing, resource inefficiencies, inventory, hand-offs, re-invention, and unnecessary motions. Additionally, it is to be noted that lean has been implemented to projects for different objectives. Petersen (2012) proposed lean indicators to identify waste in the software maintenance process through a case study.

2.7 Lean Principles in SD Projects

Software companies have been looking for a new innovative approach that provides operation based advantages required for increased productivity and to fulfill the changing customer demands for quite some time now. The first recorded experiments with Lean Software Development (LSD) were by Middleton in 1993 and the first industrial implementation of lean principles in Software Company (Timberline Software, Oregon) was recorded in 2002 (Middleton, 2005). The Lean principles in SD projects, however, have been articulated in different ways by different authors described below.

Middleton *et al.* (2005) used 11 lean principles continuous flow processing, customer-defined value, DSM and flow, takt time, linked processes, standardized procedures, rework elimination, load balancing, posting results, data driven decisions and inventory minimization. Poppendieck and Poppendieck (2006) articulated seven lean SD principles concerned with value, cycle time, value-creating activities, value stream, quality, people and concurrent development. Similarly, Middleton *et al.* (2010) examined the lean ideas from the Toyota Production System (TPS) to software projects and carried out case study with seven lean principles from Liker's 14 lean principles. Anderson (2010) recommended five lean principles: visualize workflow, limit WIP, manage flow, make process policies explicit and improve collaboratively. Petersen and Wohlin (2011) used seven principles (Pull, Kanban, limiting WIP, early fault detection, continuous workload, flow and overall development of lifecycle) in LSD. Middleton and Joyce (2012) carried out several case studies and focused on seven lean principles from Liker's 14 principles in SD projects. Staas *et al.* (2011) implemented lean principles in Indian software industries and found impact of techniques on problem resolving by coordination through connection and pathways and standardization.

It is noted that Womack and Jones (2003) categorized five core lean principles with reference to manufacturing**:** value (identify what really matters to the customer), value stream (ensure every activity adds customer value), flow (Eliminate discontinuities in the value stream), Pull (production is initiated by demand) and perfection (retaining integrity via continuous improvement and Poka-Yoke). It is argued here that the lean principles articulated by researchers for LSD fall into these five core lean principles and thus have been categorized accordingly in Table 2. 3. The five core Lean principles in context of SD projects are defined as follows-

Table 2.3: Lean Principles in SD Projects

Lean Principles	**Authors**
Value	Widman, (2010); Ikonen *et al.*, (2010); Womack (2003); Middleton (2005); Poppendieck and Poppendieck (2007) ; Jailia *et al.*(2011); Karvonen *et al.*(2012); Pernstål *et al.*, (2013); Rodríguez *et al.*(2013); Yadav *et al.*(2018),
Value Stream	Poppendieck and Poppendieck (2006); Karvonen *et al.*(2012)
Flow	Mehta *et al.*, (2008); Andersson, (2010); Mandić *et al.*(2010); Staats *et al.*(2011); Petersen and Wohlin (2011); Karvonen *et al.*(2012); Rodríguez *et al.* (2013); Yadav *et al.*(2018)
Pull	Petersen and Wohlin (2010); Andersson, 2010; Petersen and Wohlin, 2011; Rodríguez *et al.*(2013)
Perfection	Rodríguez *et al.*(2013)

2.7.1 Value is everything that a customer is willing to pay for. The central focus of Lean thinking is the customer value (Pernstål *et al.*, 2013; Rodríguez *et al.* 2013; Yadav *et al.*, 2018). Every activity inside an organization should add value, which really matters to the customer (Womack and Jones (2003).

2.7.2 Value stream is the optimized end-to-end collection of actions required to bring a product from customer order to customer care (Rodriguez *et al.*, 2018). The aim of the value stream is to determine those activities that do not contribute value to the product. Also current and future development process diagram are framed and the sources of waste is identified in SD such as delays, unnecessary rework.

2.7.3 Flow implies that development activities should be organized as a continuous flow, removing discontinuities. It requires that non value-added activities and inventories are eliminated (Mehta *et al.*, 2008; Andersson, 2010; Mandić *et al.* 2010; Staats *et al.*2011; Petersen and Wohlin, 2011)

2.7.4 Pull emphases on developing products just when they are truly required. Karvonen *et al.* (2012) stated "In a pull system, an upstream process develops only when downstream process is ready and pulls some more work from the upstream process".

2.7.5 Perfection refers to continuous improvement and learning cycle. Rodríguez *et al.*, 2013 revealed "Lean aims to achieve zero waste and defects based on the concept that there is no end in the strive for perfection and learning for SD projects".

In one of the recent studies, Rodríguez *et al.* (2018) identifies the lean principles that characterize the lean in SD projects and state: "Lean principles are adopted in SD organizations by creating a culture of customer value in which everyone cares about providing customer value, seeing-the whole value stream, providing continuous flow through small batches of working software and creating a learning organization to adapt to business and market changes, guided team level activities".

2.8 Responsiveness in SD projects

Five lean principles with respect to SD projects in line with manufacturing have been discussed the previous section. During our research it was observed that in software development, the highest priority is provided to fulfil the customer requirements through early and continuous delivery of software products requiring due consideration of responsiveness and agility in SD projects. According to Poppendieck and Poppendieck (2007), responsiveness is one of the main capabilities of responsiveness and lean. It indicates that speed and responsiveness are achieved by quick delivery and short release cycle in software development (Rodríguez *et al.*, 2019). Howleg (2005) defined responsiveness as the ability of the organization to respond to customer.

2.9 Lean Practices in SD Projects

Lean principles are implemented in a manufacturing or non-manufacturing organization through a variety of practices. SD projects are not exception to this. This section reviews the literature on important practices lean practices being considered in SD projects. Table 2.4 depicts the papers considering the applications of major lean practices in SD projects.

Table 2.4: Literature according to the Lean Practices used in SD Projects

Lean Practices	Sources(Authors)
Mistake-Proof-Process	Womack (2003);Poppendieck and Poppendieck (2007); Benefield (2009); Middleton(2010)
Set based concurrent development	Poppendiecks (2003), Poppendieck and Poppendieck (2007) , Jonsson(2012)
Standardized work	Poppendieck and Poppendieck (2007), Anderson (2010), Middleton(2005); Malladi *et al.* (2011)
KANBAN/ KANBAN BOARD	Anderson(2003), Ikonen *et al.*(2010); Poppendieck and Poppendieck (2007), Anderson (2010), Middleton *et al.*(2010)
Visual control board(VCB)	Stass *et al.* (2011), Anand et el.(2014)
Design Structure Matrix (DSM)	Stass *et al.* (2011), Middleton(2005)
Value Stream Mapping(VSM)	Mujtaba *et al.* (2010); Gustavsson (2010); Musat and Rudriguez (2010); Kuusela & Koivuluoma (2011); Stass *et al.* (2011), Peterson & Wohlin (2010) and Anand *et al.* (2014).
Continuous Improvement	Jonsson(2012), Womack et.al. (2003) , Poppendieck & Poppendiecks (2007), Anderson (2010); Kuusela and Koivuluoma (2011); Samanta and Mani(2015)
Kaizen	Kuusela & Koivuluoma (2011)
Visual control	Middleton & Joyce (2010)
JIT and Pull scheduling	Poppendieck & Poppendieck (2007)
Visual management	Pernstål *et al.* (2013)

This review highlights that VSM, Standardized work, Kanban/Kanban Board, Mistake proof process and continuous improvement are popular and extensively adopted in SD projects. Middleton *et al.* (2005) claimed, that lean principles and techniques (standardized procedures, linked processes, data-driven decisions, takt time, design structure matrix (DSM), breaking down requirements into chunks, minimizing inventory and posting results) have already been applied in SD company (Timberline Software) at the beginning of the twenty-first century. Middleton (2001), through a case study, concluded that some of the lean techniques which transformed manufacturing industries could also transform SD by minimizing work in progress.

Poppendieck and Poppendieck (2006) explored the utility of JIT and Pull scheduling in SD. Additionally, a little evidence has also been found regarding the application of Kaizen (Kuusela and Koivuluoma, 2011) and Visual control (Middleton and Joyce, 2010). Staats *et al.* (2011), conducted a case study in an Indian software service firm and observed that lean tools: visual control board (VCB), Design Structure Matrix (DSM), system complexity measures (SCM), VSM and iterative design were used in SD firms. Lean practices can be defined as tools and techniques used for implementing lean philosophy. Middleton and Joyce (2011) applied lean tools such as visual management, team-based problem solving, and SPC (statistical process control) to improve SD projects through a case study.

Pernstål *et al.* (2013) reported that visual management and VSM were the specific lean practices used in large scale SD projects. VSM has also been utilized by few researchers. Value stream mapping (VSM) tool has also been used by few researchers such as Mujtaba *et al.* (2010), Gustavsson and Axelsson (2010); Musat and Rodríguez (2010); Kuuselaand Koivuluoma (2011); Petersen and Wohlin (2010) and Anand *et al.* (2014). Recently, KANBAN has also gained momentum in SD projects, and its impact has been investigated by some researchers (Ikonen *et al.*, 2010; Poppendieck and Poppendieck 2007 and Corona and Pani 2012). Additionally, a little evidence has also been found for the application of visual control. It is evident from this discussion that limited lean practices are adopted in SD projects and are related to software process improvements. The most used lean practices in SD projects are described below:

2.9.1 Mistake proofing

It is used for developing mistake proof code in SD projects. Poppendieck and Poppendieck, 2007 suggested that every code base should contain a set of mistake-proofing tests that do not let error into the code including at the unit and acceptance test level and also found that agile team has the main focus on mistake-proofing of the codes.

2.9.2 Value Stream Mapping

Value stream mapping (VSM) is used for determining waste. VSM includes the development of current and future activities diagram aims to determine source which do not contribute value and all the sources of waste. Poppendieck and Poppendieck, 2007 revealed that VSM is a timeline phases for deploying code in SD projects.

2.9.3 Standardized work

Standardization is a key technique for obtaining flow in SD projects. Rodriguez *et al.* (2013) revealed that standards can be adopted in areas such as coding and partially standardized.

2.9.4 Visual control

Visual control is a lean tool to show the status of the development process so that everyone involved in the development process can understand it at a glance. Middleton and Joyce (2012) found that visual controls were used as lean practice extensively in SD projects.

2.9.5 Visual management

Visual management is used for task prioritization. Middleton and Joyce (2012) used visual management as lean practice in software development project management.

2.9.6 Kanban

Kanban is used for continuous flow of work. It contains a sufficient description of a job that is the right size and is the right thing to do next. Anderson (2010) revealed that Kanban as a means to bring Lean thinking into a SD organization. But the challenge with Kanban is not really depicting how people might go about selecting what to do next (Poppendieck and Poppendieck, 2007)

2.9.7 Kanban Board

The Kanban board a virtual KANBAN system and visual control. It envisages the status of each activity in the development process of SD projects. Kanban board is used for visually displaying the flow of work and helps to measure and manages flow. According to Middleton and Joyce (2012), Kanban boards are used to control the level of WIP and enable bottlenecks to be quickly recognized.

2.9.8 Design Structure Matrix (DSM)

The DSM is used to optimize information flow. It advantageously allows design engineers to understand the underlying complexity of the PD process by identifying dependency among design activities (Nepal *et al.***,** 2011**).** Middleton *et al.*, (2005) used DSM as lean practice for SD projects.

2.9.9 JIT Information

Yadav *et al.* (2019) found JIT information as lean practice for adoption of lean principle in SD projects. In this practice, only information is transferred on the basis of Just in Time in software development.

2.9.10 Flexible and cross functional team

While developing large scale software products with many teams and tasks going on in parallel, then with scrum framework, team members actively participate in the lean transformation. A flexible and cross functional team is a small number of people with complementary skills who are committed to a common purpose, performance goals, and approach for which they hold themselves mutually accountable.

2.9.11 Cause and effect matrix

In software development, method (Software development life cycle), code (source code, framework, code libraries), people (team members); systems (development, releases, deployments, testing) are the causes of the problem of the software development. In software development project, it can mean debugging for hours and getting no closer to a solution. This is used for identification of problems' root causes.

2.9.12 Set base concurrent development

It is a lean practice of reasoning, developing, communicating sets of solutions in parallel but independently, tradeoffs, and finally narrowing respective sets of solution based on additional information from other function such as testing and customer (Sobek *et al.*,1999)

2.9.13 Continuous improvement/Kaizen

Standardization, learning, coordination, rich communication, and path simplification are the essential building blocks for improvement of a SD and provide a specific base to continuous improvement.

2.9.14 Team-based problem solving

It is a technique to identify root causes of problems. Also, It is used in process improvement in organizational progress and prioritizing problems. Middleton *et al.* (2012) showed the significance of team-based problem solving as lean practice to improve performance of software development.

2.9.15 Heinjunka

Womack and Jones (1996) defined as level schedule by sequencing orders in a repetitive pattern and smoothing the day to day variations corresponding to demand. Likewise in software development Work load leveling is important to increase the speed of release.

From the literature review it is noted that researchers have focused more on value (reducing waste to improve customer satisfaction) and flow (of information) in the SD projects. This is investigated in the current research through case studies and empirically.

2.10 Drivers and Barriers of Lean implementation in SD Projects

Identification of drivers and barriers is one of the most important aspects of management research to help organizations in implementing an improvement managerial concept. Lean in SD projects is no exception to this hence the drivers and barriers have been studied by many authors described below.

2.10.1 Drivers for Lean implementation in SD Projects

Drivers or critical success factors (CSF) of management initiatives have been identified by researchers in different settings. For example, Dora *et al.* (2015) identifies commitment of top management, organizational culture, skill and training, resources, multifunctional team, organizational structure, remuneration and rewards, change agent, and piecemeal approach as drivers for lean in manufacturing. In one of the recent study, Kobus and Westner (2015) identifies training and education, existing skills, organizational changes/standardization, leadership involvement, employee involvement, customer focus, change culture and work ethics, communication and financial resources, implementation facilitation as CSF for lean management in IT. Following are some of the drivers complied from literature review for consideration in lean implementation in SD projects.

2.10.1.1 Team Involvement/ Employee involvement

Effective team involvement is necessary in successful adoption of lean in SD projects and to get solution of customers' problem. Middleton and Joyce (2012) revealed that team should be customer-focused and responsive to customer need and also reported that the development team should proactively move upstream to work with customers to define and analyze their problems to truly deliver value to the customers. The main objective of team involvement is to offer value as quickly as possible to the customer. Training and incentive scheme can be adopted for the team to make effective lean team.

2.10.1.2 Business Value with top management support

Middleton and Joyce (2012) observed that business value is created after delivering the product to the customer. This can be achieved to minimize cost, to eliminate wastes (rework/delays), build quality into the software product by top management support. Business strategies are decided by business board to incorporate value. Poppendieck and Poppendieck (2007) stated that business value can be achieved by delivering more and better features, instead of delivering in shortest amount of time for the lowest cost.

2.10.1.3 Rapid Delivery as part of performance management

Lean SD projects evaluate their operational performance by measuring the end-to-end cycle time of core business processes. Rapid and continuous delivery improves the project performance. Middleton and Joyce, 2010 reported that the rapid delivery with a focus on creating the highest value to the customer also reduced both technical and market risks. Fast delivery reduces lead time along with reduced variability. Through lean approach, the speed of delivery can be improved. Poppendieck and Poppendieck, 2007, considered 'deliver fast' as lean principle for SD projects.

2.10.1. 4 Mistake proofing for quality into the product

By incorporating quality into product, highest quality can be delivered to the customer. Quality of SD depends on robustness of codes. According to Poppendieck and Poppendieck (2007) lean approach to software development can create significant improvements in the quality of the software. Quality is improved in SD project by codes through model view controller or object-oriented base. It can be achieved in SD project by building quality into the code from the start rather than test. Kobus and Westner (2015) identified mistake proofing as driver of lean adoption in IT organizations.

2.10.1. 5 Customer focus/ Customer Satisfaction

Customer focus is central to lean philosophy in SD projects (Timans *et al.*, 2012). The success of any SD organization depends on customer satisfaction and close collaboration with the customer. The goal of lean development should be to find ways to delight customers by understanding their situation deeply and solving their problem completely (Popendieck and Poppendieck, 2007). Customer-defined value can be achieved through bug fixing and incorporating new features (Mehta *et al.*, 2008; Middleton and Joyce, 2010).

2.10.1.6 Coordination and communication

Faster communication among project team members and other stakeholders stimulates the performance of development teams. Yadav *et al.* (2007) stated that there is urgent need to embark on collaborative build a virtual reality collaborative environment to facilitate smooth coordination and communication for information sharing.

2.10.2 Barriers inhibiting Lean implementation in SD projects

The primary objective of this research is to identify the barriers for the successful implementation of the lean system in software development projects. The major issues/challenges in adoption of lean principles in SD project have been investigated by various researchers which have been classified into three broad categories described below.

In literature, various barriers are found in adoption of lean in SD projects-

(I) Customer Oriented- Uneven requirement flow and change requests, Lack of information sharing or communication between organization and customers

(II) People oriented-Lack of communication and coordination within the team and among cross functional team, lack of availability of required and allocation of resources, difficult to create a cross functional focus, lack of lean implementation experts with good leadership and coaching skills

(III) Process Oriented- Lack of coordination for parallel and sequential tasks, lack of prioritization of bug fixing and adding new features, lack of transparency, lack of techniques to recognize waste in SD Projects.

After reviewing literature, a few important barriers are extracted during lean adoption in SD projects are summarized in Table 2.5.

Table 2.5: Major Barriers in SD projects

Category of Barriers	**Challenges/ Barriers**	**Literature Source**
Customer - related	Uneven requirement flow and change requests.	Laura Bocock, 2011; Peterson and Wohlin,2010
	Lack of information sharing or communication between organization and customers	Mujtaba *et al.*,2010; Middleton 2001; Poppendieck & Poppendieck,2006
Process - related	Lack of techniques to recognize waste in SD Projects.	Ikonen et el.,2010
	Lack of coordination for parallel and sequential tasks.	Petersen and Wohlin,2010
	Lack of prioritization of bug fixing and adding new features	Mehta *et al.*,2008
	Company policy and structure creates obstruction	Middleton,2001
	Lack of transparency	Rodríguez *et al.*, 2013; Petersen and Wohlin (2010)
	Lack of proper requirement handover / unnecessary handovers	Petersen and Wohlin,2010
People - related	Lack of lean implementation experts with good leadership and coaching skills	Dora *et al.*,2015;Jadhav et el.,2014
	Lack of top management commitment	Dora *et al.*,2015;Jadhav et el.,2014
	Lack of communication and coordination within the team and among cross functional team	Jadhav et el.,2014
	Difficult to create a cross functional focus.	Petersen & Wohlin, 2010
	Lack of consultants and trainers in the company and outside	Jadhav et el.,2014
	Functional silos and Employees' resistance inside the organization	Rodríguez *et al.*, 2013
	Lack of availability of required and allocation of resources	Jadhav et el.,2014

2.10.2.1 Customer related barriers

Customer focus is central to lean philosophy. The success of any SD organization depends on customer satisfaction and close collaboration with the customer. In SD, two types of customers are considered: external customer or end user and internal customer. Vague requirements and frequent changes in the requirements are found to be the most important issues in lean implementation. Customers were more influenced by the impact of lean on SD product customization and the capability to improve their requirement awareness. Customer related issues in SD projects are failure in understanding customer's needs and requirement/ priorities change in Software (Bocock, 2011). Some of the challenges observed in literature related to customer such as uneven requirement flow and change requests (Peterson and Wohlin, 2010), lack of information sharing or communication between organization and customers. Customer related barriers are discussed below-

I Uneven requirement flow and change requests

Success of a software project pivots on a clear and complete understanding of the problem to be solved as well as a thorough understanding of the customer's needs and expectations. Lean can be applied where frequency and demand are consistent (Perez *et al.*, 2010). That's why it is a big challenge for lean adoption in SD projects where requirement and request are uneven (Laura Bocock, 2011; Peterson and Wohlin, 2010). A complete and accurate requirement is essential for the development of a successful adoption of lean in SD project.

II Lack of information sharing or communication between organization and customers

For effective lean implementation, substantial information sharing among various segments of SD organizations is necessary. Jadhav *et al.*, 2014 revealed that high interactions and correct information sharing about customers' requirements become essential for effective implementation of lean.

2.10.2.2 Process related barriers

Repetitive testing and debugging were found to be the major process-related issues as they increase the development time and delay release. The other major process related issues include the prioritization of bug fixing and addition of new features in SD. If the requirement changes are frequent, then technology constraints such as platform and hardware issues become prominent. Process related issues are achieving flow, transparency, creating learning culture (Rodríguez *et al.*, 2013). Financial capability, organizational culture, organizational structures have also been identified as key issues in SD projects (Middleton, 2001). Likewise, Mehta at el. (2008) identified internal service, testing and debugging as issues in implementing lean in SD projects. Process related challenges are discussed below-

I Lack of prioritization of bug fixing and adding new features

Mehta *et al.* (2008) raised a question regarding SD project that should we fix the error or should we continue the work by adding new features without fixing the problem. Based on the customer value, it was recommended to assign a common prioritization code for errors and new features. New features are incorporated into SD and bug fixing is carried out in product based software organization on priority and severity of the impact.

II Lack of techniques to recognize waste in SD Projects

Ikonen *et al.* (2010) revealed that in SD projects, there is invisibility of waste and not easy to recognize waste in SD projects. Waste restrains the progress of SD project, which endanger their success. Waste such as rework is identified as major issue which is source of delay (Mehta *et al.*, 2008). Waste creates obstruction in flow of value and information. Lean tools which contribute to waste elimination in SD projects are such as VSM, visual control and mistake proofing (Anand, 2014).

III Lack of coordination for parallel and sequential tasks

Petersen and Wohlin (2010) found that highly complex with many different tasks going on in parallel in SD projects, which makes coordination challenging. Nepal *et al.*, 2011 characterized that sequential tasks are those which depend on information produced by earlier tasks and parallel tasks that can be completed at the same time, as there is no information sharing between them. In software development, it is observed that sometimes there is lack of coordination in parallel and sequential tasks.

IV Lack of transparency

Mehta *et al.* (2008) identified challenges to display substantial information in software development. Likewise, Petersen and Wohlin, 2010 revealed that SD projects are highly complex system, which makes transparency challenging. Rodríguez *et al.* (2013) stated that there is lack of transparency in SD projects, because there is still a tendency in development teams to limit collaboration outside the team to agile need.

V Company policy and structure creates obstruction

Adoption of lean practices depends on scale of organizational change; hence it limits their adoption (Middleton, 2001). Organizational structure can create hindrances as management; training and operations are typically distinct departments that may have a little interaction. Recognition and rewards from the top management will fill in as a booster for involvement and continuous improvement (Wong *et al.*, 2009)

VI Lack of proper requirement handover / unnecessary handovers

Hand over is the process of transferring the responsibilities for the acquired software products from the project manager to the software support organization (Khan, 2013). Lack of requirement handover created due to an inappropriate communication, lack of awareness of handover status.

2.10.2.3 People related barriers

People-related issues are also vital to adoption of lean in SD projects. Like other management initiatives, developing an effective cross-functional team is a challenge in lean implementation. Poppendieck and Poppendieck (2006) argued that excellent software products start with highly competent technical experts in many areas: architecture, object oriented technologies, coding strategies, data structures and test automation. People related issue is cross-functional conflicts (Middleton, 2001). Similarly, training and skill building, communication, management leadership, management support, top management commitment are concerned with people related issue (Kundu and Manohar, 2011).

People barrier applies to lean implementation is to be investigated in present research. Some of the major lean barriers identified in literature for project such as lack of top management commitment, difficult to create a cross functional focus, lack of lean implementation experts with good leadership and coaching skills, functional silos and employees' resistance inside the organization, lack of availability of required resources and allocation of resources, lack of communication and coordination within the team and among cross functional team and lack of consultants and trainers in the company and outside the company. Due to its collaborative nature, managing people, organizational inertia and change management are even more critical to SD implementation (Middleton, 2001). People related issues are discussed beow-

I Lack of lean implementation experts with good leadership and coaching skill

Tracy, 2007 identified that it's necessary to bring in outside experts to successfully shift to lean. It is difficult to hire experts for implementing lean in SD projects because of apprehensions that it may lead a loss of expertise. Lack of awareness for implementation may be reversed into "create awareness for lean implementation in SD projects". Cudney and Elrod (2010) revealed that lean implementation could not reach its anticipated purpose if there were unsuitable training methods and knowledge handovers. Likewise, Basin (2012) investigated that training (coaching) and education is a fundamental part of the adoption of lean management as preventive cost in order to avoid subsequent costs caused by inappropriate skills in industry. Therefore an effective lean team and training to employees and management are important lean practices in SD projects. Train team leads, supervisors, and managers how to lead, how to teach, and how to help workers use a disciplined approach to improving work processes.

II Lack of communication and coordination within the team and among cross Functional team

Reinertsten (2005) revealed that information is the output in product development (PD).Since software development is akin to PD. For lean transformation, communication is a key to share information. Jadhav *et al.* (2014) revealed that lack of team autonomy and lack of organizational communication led to the termination of the lean projects. Yadav *et al.*, 2010 revealed that lack of communication creates 70 percent of rework in product development. Due to a high level of tacit knowledge in SD (Poppendieck and Poppendieck, 2006; Staats *et al.*, 2011; Anand and

Kodali, 2010a), it is difficult to communicate and hand off knowledge to other people. A communication plan should be developed to share timely accurate information about the status of the SD project.

III *Difficult to create a cross functional focus*

Excellent software products start with highly competent technical experts in many areas architecture, object-oriented technologies, coding strategies, data structures, and test automation (Poppendieck and Poppendieck, 2007). Software development involves constant problem solving; people make complex decisions many times a day, often with implications far beyond their own work. There is still much individual work in some teams and resistance to pair work was also found. The team oriented issues can be resolved by using multi-skilling employees to balance workload and to eliminate the bottlenecks.

IV Functional silos and Employees' resistance inside the organization

Information overload, demand for agile operations and technology complexity have created unnecessary barriers between the teams and organizational departments. Further, lack of clarity about responsibilities and measurement of deliverables are quite common in SD projects which may be because of tacit knowledge and the existence of silos.

In many ways, this trend has also contributed toward the silo mentality within organizations, where individuals, groups and departments hesitate from sharing information and tasks with others across the organization. Functional silo is an individual business function which is divided into sub groups and has its own strategies and works parallel with other organizations. Rodrigues *et al.* (2013) revealed that functional silo creates barrier in flow and pulling of information in SD projects.

Employees' resistance is another reason which hinders the lean adoption in software development. Lean implementation could lead to staffing reductions is one of the causes of reluctance by some employees (Buesa, 2009).

V Lack of consultants in the company and outside

To create a lean project, it needs tremendous change in culture, habits, attitude of team and management as well as systems. Timans *et al.* (2012) revealed that previous knowledge in process improvement programs can be valued for the skills necessary to adopt lean management. Apart from above, lack of clarity about responsibilities and measurement of deliverables are quite common in SD projects which may be because of tacit knowledge, lack of communication and the existence of silos.

VI Lack of top management commitment

Top management ensures that customer requirements are determined and are met with the aim of enhancing customer satisfaction. In SD project, top management is associated in achieving goal, while in lean it is expected that senior management should support and commit lean adoption and arrange training for employees in lean tools& techniques. Mehta *et al.* (2008) stated that functional manager could be responsible in elimination of waste in coding activities by separating value added task and non- value added task. Scherrer-Rathje *et al.* (2009) revealed that lack of top management commitment could lead to create issues such as limited access to resources, prolonged decision-making processes and communication interruptions in organizations.

VII Lack of availability of required and allocation of resources

Lack of resources (financial, technical, human) is a common barrier for implementing lean (Pedersen and Huniche, 2011a; Achanga *et al.*, 2006; Bateman and Rich, 2003) in SD projects. Resources, mainly financial capabilities include, for example, funds to cover training costs, external consultancies or any other related investments, which play a significant role in lean adoption (Bhasin2008, Dora *et al.*, 2013).These financial resources should be secured to cover implementation cost.

Lessons learned from this chapter can provide the basis on which project managers and software practitioners can design concrete strategies that would enhance the performance of software development to high quality ends.

2.11 Impact of Lean on performance of SD Projects

Measurement of impacts of a management initiative on organizational performance is always a challenge for management. As Lean is increasingly being used in non-manufacturing area as a performance improvement method (Stone, 2012; Esain *et al.*, 2008; Baines *et al.*, 2006; Ziskovsky and Ziskovsky, 2007; Paez *et al.*, 2005) measuring impacts of lean on performance is very important. Some of the researchers have studied impacts of lean on operational performance as discussed below.

Bhasin (2008) reported that non-financial measures such as quality, customer satisfaction and innovation have become increasingly important in knowledge area. Bond (1999) categorized performance parameters: quality; delivery reliability, customer satisfaction, cost, safety and morale of an organization. Gains were evaluated in terms of quality, schedule adherence and lead time. The reduction of cost is equivalent to waste reduction. Womack and Jones (2005), Liker (2006) considered less WIP as performance parameter in adoption of lean in an organization. In one of the studies (Middleton and Joyce, 2012) performance of the software development team was found to improve by adopting a lean approach. Consequently, lead time to deliver software improved, consistency of delivery rose and defects reported by customers fell and then reduced both technical and market risks. Based on the literature review, four performance parameters such as cost, delivery time, customer satisfaction and WIP are taken for considered for investigation in the current research.

2.12 Lean in Manufacturing (LM) v/s Software Development (LSD)

It is noted that Lean was first adopted in manufacturing which later on was extended to other sectors of the economy including SD projects. The review of literature has also revealed that there are differences in applications of lean in manufacturing and SD projects with respect to the applicability of five lean principles and practices. This may be due to differences in the characteristics of the two sectors with respect to the processes or performance measurement. It is thus considered important to identify the difference between these two sectors to understand lean transformation from manufacturing to software development projects. Table 2.6 gives us insight and comparison between LM and LSD.

Table 2.6: Comparison between Lean Manufacturing (LM) and Lean Software Development (LSD)

S. No.	Parameter	LM	LSD
1.	Waste Identified	Visible sources of waste in production flow, Seven wastes (Shah and Ward,2003; Simon and Zokaei,2005; Petersen,2009)	Invisible in SD (Poppendieck & Poppendieck,2005; Ikonen *et al.*,2010; Petersen and Wohlin,2011); Waste in projects occur during the transaction of information (Worrall,2003)
2.	Requirement	Responsive to customer demand (Hines *et al.* 2006); the alignment of production with demand (Andy Lyon 2011 ; Shah and Ward,2003)	Vagueness in requirement (Poppendieck & Poppendieck,2005; Petersen and Wohlin,2011)
3	Inventory	Raw material, WIP, finished goods which are tangible (Melton,2005)	Requirement, design, code, partially done work which is intangible. (Poppendieck & Poppendieck,2006)
4	Flow	Flow of material, cash, resources, and information (Womack&Jones,1994;Lewis,20000;Storch & Lim,1999)	Frequent transfer of preliminary information between development stages. Continuous flow processing(Poppendieck& Poppendieck,2006)
5	Task	Predictable tasks of similar size relatively low variability (Iberale, 2010)	Deals with tasks which are inherently variable and dissimilar (Iberale, 2010).
6.	Deliverables	Tangible (Hines *et al.*,2006, Panwar *et al.*,2015)	Intangible (Poppendieck and Poppendieck,2006)
7	Input	Raw materials, Engineering Specifications, Customer requirement (Hines *et al.*,2006; Jack Cook,2004)	Customer Requirements, System Analyst's Specifications, Programmers's coded object (Jack Cook,2004)
8.	Value	To eliminate wastes and Value addition (Hines,2004) Value creation (Liker,2004) and innovation (Shah and Ward,2003)	Customer defined value through bug fixing and incorporating new features (Mehta *et al.*,2008; Middleton and Joyce,2010)

In SD, process invisibility prevents problems from being identified early enough to be solved efficiently and effectively (Anand *et al.*, 2014; Staats *et al.*, 2011).Comparison between LM and

LSD motivate us to explore lean principles and to examine the adoption level in SD projects through case studies and survey.

2.13 Research Gaps

It has been found that lot of empirical and case studies on lean implementation have been carried manufacturing and other sectors of economy such as product development, construction and service but very little work has been reported for lean in SD projects. It is also not clear which lean principles/practices are more suitable (being applied) to SD projects. There is thus a need to identify lean principles/practices that are applied to SD projects and the issues/barriers that hinder its implementation in SD projects. With this objective, exploratory case studies were needed to conduct in SD organizations.

Only a few (e.g. Stass *et al.*, 2011; Kundu *and* Manohar 2016) empirical studies have been reported about lean in SD projects. Therefore, more empirical studies are needed to investigate implementation of principles/practices and relation between lean implementation and operational performance of SD projects. It provides a sound basis for further empirical research on adoption of principles in software development projects.

Based on literature review following research gaps are identified for research in lean initiatives in SD projects.

i. A plethora of literature is available on adoption of lean in manufacturing, but it is relatively scant for SD projects.
ii. The set of lean principles/practices used in manufacturing have already been well established but it is not so for SD projects. As manufacturing and SD projects have different characteristics it needs more research to identify as to which lean principles/practices are suitable or implemented to SD projects.
iii. There have been some case studies on lean in SD projects but very few empirical studies are observed. As generalization is generally an issue with case studies more empirical investigation is required.
iv. It is always required to examine the effects of management initiatives on organizational performance. To the best of my knowledge no literature is available that directly and

indirectly measures the effect of lean initiatives on organizational performance in SD projects.

2.14 Summary

This chapter has presented a state of the art review literature on lean adoption in software development projects. From literature review it is revealed that very little work is reported on implementation of lean in SD projects. Insights of principles and practices that have been found to be implemented have been obtained. The drivers and barriers to implement lean in SD projects is also identified. There is, however, a need for further research to see whether the lean principles/practices that have been implemented in manufacturing and other sectors of the economy are suitable or only a few of them are applicable as suggested by some of the authors.

Further, it was found that most of the work on adoption of lean in SD projects has been done through case studies and only a few studies have tried to test the same empirically. There is thus a strong requirement for empirical study. It was also found that no effort have been put to assess the effect of lean adoption in SD projects.

Addressing the research gap, adoption level of lean principles/practices is examined through case studies and survey. Addressing the second research gaps, limited lean principles and practices are adopted, hence this motivate to conduct study to apply suitable lean practices in SD projects. A model will be developed to examine the adoption of lean principles and responsiveness in SD projects, which indirectly measures the effect of lean initiatives on organizational performance.

Chapter 3

Research Design and Methodology

Chapter 3

Research Design and Methodology

3.1. Introduction

This chapter focuses on the particulars of research methodology followed to address the research objectives. The research methodology comprises of the research plan, techniques of data collection and analysis to answer the research questions. The research questions considered in the thesis are as follows:

RQ1. How lean is considered in SD projects?
RQ2: Which lean principles/practices are prevalent in SD projects?
RQ3: What are the drivers and barriers of lean implementation in SD projects?
RQ4: Do implementation of lean affect the organizational performance?

To answer these questions the two phase research process is adopted. In first phase, exploratory case studies are conducted to get preliminary insights of adoption level of lean in software development projects. Subsequently, survey is carried out in large number of SD organizations for validation of conceptual model.

3.2 Research Plan

The research plan followed in the current research is given in Figure 3.1. Survey research methodology is discussed.

As a first step a comprehensive review of literature were conducted on fundamentals on lean principles and practices and the drivers and barriers of implementation in Manufacturing and other sectors of the economy and status of lean in software development projects. On the basis of the literature review the research gaps were identified. A details of the literature review has already been presented in the previous chapter.

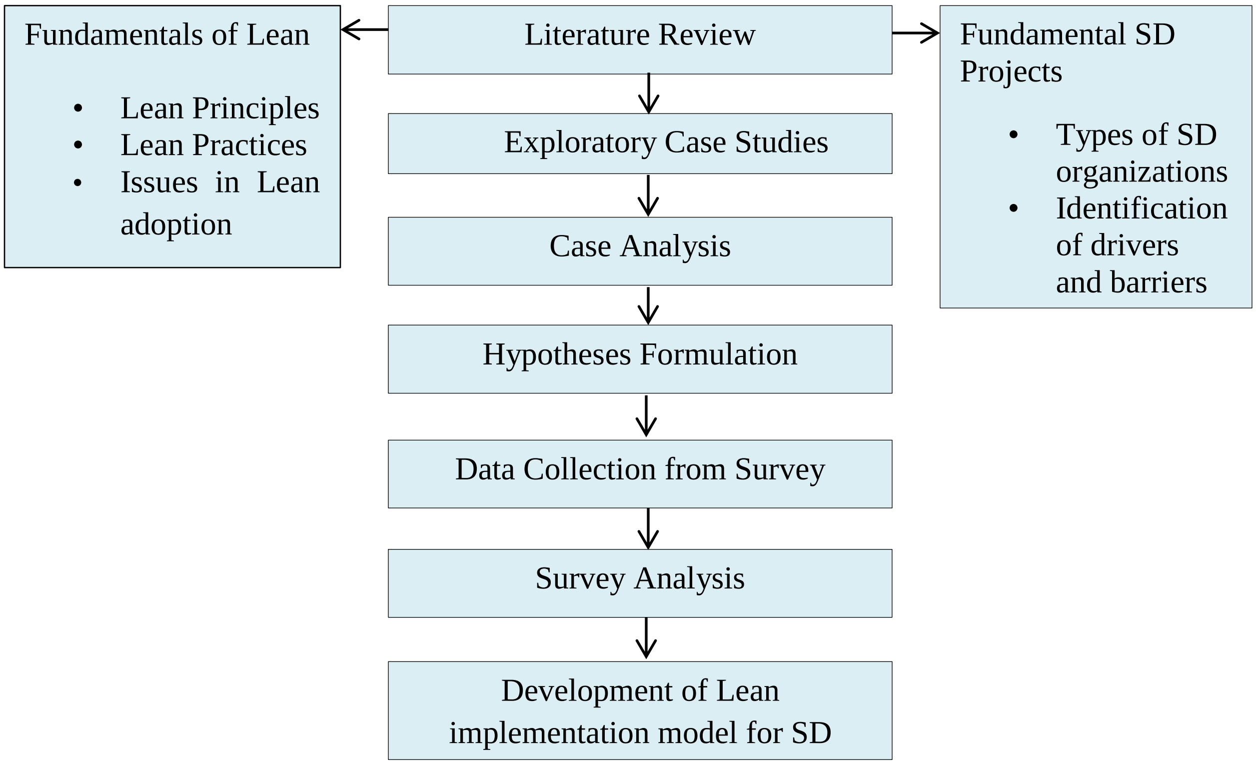


Figure 3.1: Research Plan

3.2.1. Exploratory Case Studies

As there is very scant literature is available on lean in SD projects exploratory study was conducted to understand how lean is considered in SD projects, identify the lean principles/practices that are prevalent in SD projects and the challenges the SD organization face while implementing lean. The purpose of exploratory case studies is to lend early exploratory investigation in which variables are unknown and to gain full understanding of "phenomenon of adoption of lean principles in projects" through observing lean practices in real time. In the early stage examination is needed to develop research ideas and questions for worth pursuing further (Voss *et al.*, 2002).

Case study approach has been adopted for this exploratory study in which five case organizations have been considered. The case study approach is preferred in examining contemporary phenomena, but when the relevant behaviors cannot be manipulated. So in general the case study has a general advantage when a 'how' or 'why' question is being asked about a contemporary set of events over which the investigator has little or no control in which multiple sources of evidence are used Yin (2003). Case study is used when the boundaries between phenomenon and context are not clearly evident. Eisenhardt (1989) argued that case studies can be used to accomplish various aims such as providing a description, generating a theory or testing a theory. In current research adoption of lean as phenomenon and SD projects are context, which are not explicitly defined in literature in contrast to lean manufacturing. Hence exploratory case studies are carried out.

3.2.2. Hypotheses Formulation

Based on the literature review and analysis of the cases following hypotheses were developed for testing in the empirical study. These hypotheses are developed for drivers and barriers of lean implementation in SD Projects and impact of lean principles on operational performance of the SD Organizations. The hypotheses are listed below:

3.2.2.1 Hypotheses related to Drivers of lean implementation

H1a: Lean is implemented in SD projects in order to eliminate wastes (rework/delays)

H1b: Lean is implemented in SD projects in order to minimize project cost

H1c: Lean is implemented in SD projects in order to increase customer satisfaction

H1d: Lean is implemented in SD projects in order to build quality into the product

H1e: Lean is implemented in SD projects in order to get rapid delivery of software with coordination and communication of team

H1f: Lean is implemented in SD projects in order to get solution of customer's problem

3.2.2.2 Hypotheses related to Barriers in Lean Implementation

H2a: Uneven requirement flow inhibits lean implementation in SD Projects
H2b: Lack of information sharing inhibits lean implementation in SD Projects
H2c: Lack of communication inhibits lean implementation inhibits in SD Projects
H2d: Lack of coordination for parallel and sequential task inhibits lean implementation in SD projects

H2e: Lack of availability of resource inhibits lean implementation in SD Projects

H2f: Lack of transparency inhibits lean implementation in SD Projects

H2g: Lack of top management commitment inhibits lean implementation in SD Projects

H2h: Lack of cross functional focus inhibits lean implementation in SD Projects

H2i: Lack of techniques to recognize waste SD Projects inhibits lean implementation in SD

H2j: Lack of prioritization of bug fixing and adding new features inhibits lean implementation in SD Projects

H2k: Lack of lean implementation experts with good leadership and coaching skills inhibits lean implementation in SD Projects

H2l: Lack of proper requirement handover inhibits lean implementation in SD Projects

H2m: Functional silos and Employees' resistance inhibits lean implementation in SD Projects

H2n: Company policy and structure inhibits lean implementation in SD Projects

H2o: Lack of consultants and trainers the company and outside inhibits lean implementation in SD Projects

3.2.2.3 Hypothesis related to adoption of lean in SD projects

H3a: Value principle is significantly adopted in SD projects
H3b: Value stream is significantly adopted in SD projects
H3c: Flow principle is significantly adopted in SD projects
H3d: Pull principle is significantly adopted in SD projects
H3e: Perfection is significantly adopted in SD projects
H3f: Responsiveness is significantly adopted in SD projects

In order to meet the third objectives of the work an empirical model was developed. The hypotheses for the empirical model are as below:

H4: Value principle enables lean thinking in SD Projects
H5: Value stream principle enables lean thinking in SD projects
H6: Flow principle enables lean thinking in SD Projects

H7: Pull creating principle enables lean thinking in SD Projects
H8: Perfection creating principles enables lean thinking in SD projects
H9: Responsiveness enables lean thinking in SD Projects
H10: Lean has positive effects on operational performance

3.3 Survey Research

The hypotheses were tested using data collected through questionnaire survey in the SD organizations. The questionnaire survey consists of three steps described below:

3.3.1 Development of Survey Instruments

The main aim of large scale survey is to gather data appropriate to test theories. Based on the result from preliminary research and literature review a survey questionnaire with ordered and closed ended questions was designed.

3.3.2 Questionnaire Design

A total of 51 questions were designed which were arranged into sections (A, B, C, D, E) with number of questions in each sections as below.

Section A- Demographic information about the firm and respondents (9 questions)
Part B- Degree of adoption of lean principles (27 questions)

- Part B 1: Customer focus value (5) -VL1 to VL5)
- Part B 2: Establishment of Value Stream (VS1 to VS4) which comprises 4 questions
- Part B 3: Flow (FL1 to FL6) which comprises 6 questions
- Part B 4: Pull (PL1 to PL4) which comprises 4 questions
- Part B 5: Striving for perfect value creation (PF1 to PF4) which comprises 4 questions
- Part B 6: Responsiveness (RS1 to RS4)- 4 questions

Part C- Drivers of lean implementation- 6 questions
Part D- Major challenges in lean adoption-15 questions
Part E- Operational performance of projects- 4 questions

Seven point Likert scale was used for measurement in sections B-E described as below:

Section B and E- 1= strongly disagree and 7= strongly agree

Sections C and D-1= not important and 7= Highly Important

3.3.3 Pilot Study

Pilot study is carried out for pre testing the questionnaire before distributing the large scale survey. It is conducted to clarify the complete structure of questionnaire. These responses were used to assess initial reliability and to conduct exploratory data analysis. Dillman's (2000) suggestions were employed with slight modifications to administer the survey. In current research, pilot test was carried out with a sample size of 35 to investigate number of questions, series of questions and awareness of words used in questionnaire to respondents. The goal of utilizing the pilot test is to distinguish a dimensional structure relating to lean software development concept.

3.3.4 Sample design

Sample design is a distinct plan defined before any data are actually gathered to acquire a sample from a given population of SD organizations. Random sampling is adopted to eliminate sampling error. In this kind of sampling each and every each item within the population has equal probability of inclusion in the sample. Hair *et al.*, 2006 recommended that the quantity of survey respondents must be five times to the questionnaire items to apply structural equation modelling for further analysis. Sample design process includes selection of set of population, list of population constituents, procedure of sampling, ascertaining the size of sample.

3.3.5 Administration of Survey

Survey questionnaire was distributed in 1151 IT/ SD companies like product, project and both product and project type organizations were considered in Indian IT industry through Email and off line. Respondents were senior management, team leads, project manager, Business analyst, Team member, tester and developer. Finally 256 responses were received from various SD organizations. Response rate was 22%, which was sufficient for further analysis.

3.4 Data analysis

After completion of data collection, detailed data analysis is carried. The data was analyzed in three steps. SPSS 18.0 version is used to analyze data along with IBM-AMOS version 26.0 for SEM analysis to test the theoretical model. In current research reliability is assessed through internal consistency. According to Flynn *et al.* (1994) reliability is "the degree of inter correlation among the items which builds a scale". Cronbach Alpha was used as reliability coefficient to measure internal consistency. Reliability analysis of survey questionnaire indicates the capability to yield consistent results.

Descriptive analysis is carried out for analysis of the gathered data concerning status of lean adoption, drivers and barriers in SD projects. This analysis covers mean, standard deviation, skewness and Kurtosis, composite reliability. Also it was performed as statistical test by confirmatory factor analysis (CFA) assessing reliability and validities of responses using SPSS18.0. These analyses were conducted about the sample and about the observations that have been made. This analysis was followed to get inferences of gathered data.

Hair *et al.*, (2006) stated that Structural Equation Modelling (SEM) method was used to build the measurement and structural models. SEM was used to understand measurement theory, factor analysis, path analysis and regression in context of lean adoption in SD projects. SEM was used in path analysis for diagrammatic representation of a theoretical model. This path diagram is made with latent variable/constructs. SEM is a multivariate technique into one model fitting framework in which regression equation is established between measured variables. Also covariance among constructs structure analysis is performed. Further, Table 3.1 summarizes the model fit indices used in model analysis.

Table 3.1: Summary of model fit indices

S. No.	Model Fit Indices	Name	Acceptable Level	Comments
1.	χ^2 /df	Chi Square	≤3.00	Likely to be larger when sample size / number of observed variables increases
2.	GFI	Goodness of Fit Index	≥0.90	Ranges of values from 0-1. Higher values of GFI indicates better fit
3.	AGFI	Adjusted goodness of Fit index	≥0.8	Value of 0 (zero) indicates poor fit and 1 indicates perfect fit.
5.	NFI	Normed Fit Index	≥0.9	Range of value is 0(poor fit) and 1(Perfect fit)
6.	RMR	Root Mean Residuals	≤0.14	A mean of the residuals between individual observed and estimated covariance and variance terms. Lower value represents better fit
7.	CFI	Comparative Fit Index	≥0.9	A better version of NFI; Value of 0(zero) indicates poor fit and 1 indicates perfect fit.
8.	RMSEA	Root mean square error of approximation.	≤0.10	Used to correct the impact of sample size or model complexity on chi square

The data analysis helps to see whether or not to accept or reject the hypotheses that are acknowledged supporting the structural model.

3.4 Structural Model Testing

According to Hair *et al.* (2006), after completion of testing of measurement model, the validity of path model along with hypothesized relations are tested. General fit of path model is tested based similar to measurement model. The hypothesized path is significant if model is well fit, then structural model is considered as fit.

Confirmatory factor Analysis is employed to explain to the measurement model testing. This methodology is utilized to check the practicality of selected model and structures, which are supported theory or as research aims and moreover justify whether presented data are reliable with a proposed research model having restrained configuration. The measurement models are of two types like one as factor congeneric and other multi-factor measurement models. One factor congeneric model is utilized to evaluate item reliability, construct reliability whereas multifactor measurement models are liable to investigate the discriminant validity of the individual scales in the latent variables

Chapter-4

Exploratory Case Studies

Chapter-4

Exploratory Case Studies

4.1 Background

It was evident from the previous chapter that scant literature is available on lean in software development projects. In order to get deeper and clear insights of lean in SD projects case study approach was employed. The case study was chosen for the exploratory study to know about the intensive investigation of the application of lean in software development. The case study method allows us to explore the behavior patterns of the SD organizations. The information gathered within the case study helps in building the suitable questionnaire for the survey. This chapter discusses five case studies from software development organizations. The case studies were conducted with the purpose of better understanding of the status of lean adoption in SD organizations. Five case studies were carried out for analytic generalization of lean software development.

The cases were selected that were expected to diverge with relation to the software development process, domain, and deliverables. These are grouped on the basis of the characteristics of software organizations. In present study, case organizations are categorized as product vs project organization based on the level of customization in the software development. Some important pattern of similarities within a group of SD projects and cross-group differences for lean principles are established.

Following five SD organizations are selected to carry out exploratory case studies-

Case 1: Life insurance and financial products

Case 2: Enterprise applications

Case 3: Web applications

Case 4: Banking software solutions

Case5: Healthcare and banking software solutions

Two cases were grouped under product-based organizations, and three cases were grouped under project-based software organizations. The names of firms are not disclosed to keep anonymity hence are coded as Prodbased1, ProdBased2, ProjBased1, ProjBased2, and ProjBased3

The cases are presented in a set descriptive format. First, demographic dimensions of the firm viz. geographical area, type of products, number of employees, type of enterprises and average time span of projects is discussed. Subsequently, lean assessment of each case organizations is presented. The assessment is evaluated based on lean practices and principles adopted. The level of lean adoption and issues in implementing lean practices are also discussed. Finally, a cross-case analysis between product and project-based SD organizations is carried out for implementation of lean principles.

4.2 Plan for conducting exploratory case studies

Before starting this study, a plan has to be made for conducting case studies. The plan adopted in this research is adapted from Yin (2003) which is summarized in Table 4.1.

Table 4.1: Plan for Conducting Exploratory Case Studies

Contents	Details
Purpose	• To find efforts that are made to solve problems of consumer • To find wastes in SD projects • To assess value/ consumer want/ consumer desires • To discover root causes of problems in SD projects • To observe lean activities in SD • To assess the role of lean principles in SD with deep collaboration with customer • To pursue lean path (if followed) in life cycle of SD project • To observe lean practices, if adopted by project managers • Which lean solutions are being used to solve customer's problems permanently • To compare findings of literature review and actions taken by project manager
Key Features of case study method	• Exploratory case study • Multiple Case Studies • Embedded Case Study • Analytic Generalization

Organization of case study Plan	
Procedures	**Details**
Initial Scheduling of field visit	Case1,Case2, Case3, Case4, Case5
Review of preliminary information	Existing software, lean practices, types of projects, lean principles
Special documents	Questionnaire
Determination of persons to be interviewed and other sources of information	Project manager Type of SD Processes/ projects Waste, if any Summary of information gathered
Case Study Protocol and Questions	
• Awareness about lean, lean practices, lean principles • Whether lean is adopted in software organizations or not adopted • How and why related questions were asked • Summary of questions related to lean software development • Application of lean anywhere in organization • Current / future planning for implementation	
Analysis Plan and Case Study Reports	
Individual case studies	
Descriptive information	Outline and format of organization, information management
Cross case report	• Report of case studies within product based SD organizations • Report of case studies within projects based SD organizations • Report of cross case in product and project based SD Organizations

4.3 Data Collection

Data were collected by observation, collecting relevant documents and face to face semi-structured interviews with project managers using questionnaires (See the **Appendix I**). Open-ended questions were asked to the experts in the case companies related to the relevance of lean principles and practices, the status of adoption and issues in implementation. Direct observation and interview enable deeper understanding of the issues involved in lean adoption in a complex reality. The multiplicity of methods results in a more robust and generable set of findings. To verify the statements of project managers, administrative documents like progress reports, minutes of meetings, project proposals and service records of the number of clients served and summaries of advisory committee meetings were used as a source of evidence to ensure triangulation of sources.

Before interviews, a lot of feedback was collected using multiple data sources such as focus groups, designers, developers, testing teams and product owners to understand their viewpoints through field visits, review of preliminary information, verification of access procedures, process documentation and unique documents. Further, observation of company rules, policies, and meeting of team members also gave some insights, and finally, documents also supported the statements of project leader.

In this methodology, the field visit helped to interact with project managers and developers and helped to uncover more issues affecting project performance of SD projects with a clear view and knowledge of developmental stages of software products. Problems encountered by the team in the development are related to process, strategy and type of non-value added activities. The unstructured questions which were floated to the industry representatives were related to the issues of characteristics of SD projects, knowledge of operations, products and different types of wastes in SD projects. The other questions were associated to the pattern of lean in a different organization, management initiatives, lean practices, lean principles, the problem in SD, performance factors of SD projects, success factors, and barriers. Out of five cases, two case studies were conducted on Skype with project managers abroad. Out of these two, one is a product-based company and the other is project-based.

4.4 Description of cases

The five cases are categorized as product-based organizations and project-based organizations. In product-based organizations, deliverables are more or less standard and are created and controlled by developer organizations. On the other hand, project-based organizations develop software as per the requirement of specific customers. The products are sometimes customized to the requirements of the customers which in this study are considered as a project (Poppendieck and Poppendieck, 2006). Project-Based organizations usually produce highly customized products and services (Gann and Salter, 2000). In this research, it is proposed that product-based organizations are more akin to product development teams while project-based organizations are more like service organizations. It is also suggested that a project-based organization is a kind of enterprise where project and application development are carried out. In product-based organizations, production is triggered in response to user needs, while the project-based organizations are demand driven.

While developing lean software for project-based organizations, it was observed that in such organizations, team and organization do not freeze the design because change requests(CRs) from the customer are entertained at all stages, extending even up to the delivery stage. This increases the delay in the release of the end product to the customer. It is observed that a mutual decision between customers and organization as well as feedback from customers is a major criterion in freezing the design. Meanwhile, in product-based organizations, CRs are part of feature enhancements, and deliveries are planned with the inclusion of features prioritized by keeping eyes on the outside world. The product based companies are reacting to the changing needs rapidly to stay competitive by the adoption of continuous improvement in their products. Nowadays, instead of a waterfall model (where design freezes after estimation), iteration (code is designed, developed and tested in a repetitive cycle) and sprint (the set period in which specific incremental work has to be completed) are used. If any challenges occur, then a navigational prototype is developed; like scrum (iteration-code testing) in a web application. Before project requirement clarification, a feasibility study is carried out in project-based organizations to create deliverables. In the product-based organization's performance testing, feature addition and bug fixing are considered to be more critical, while in project-based organizations getting customer requirement and delivery is more critical. The above classification is expected to help us understand as to how value is perceived and induced in these two sets of organizations.

4.4.1 Software Development Stages in Product -based Organizations

The stages in product-based organizations are identified as customer interaction, requirement gathering, development stage, unit testing, quality engineering (QE) testing deployment and client testing. Figure 4.1 presents the stages of product based organizations-

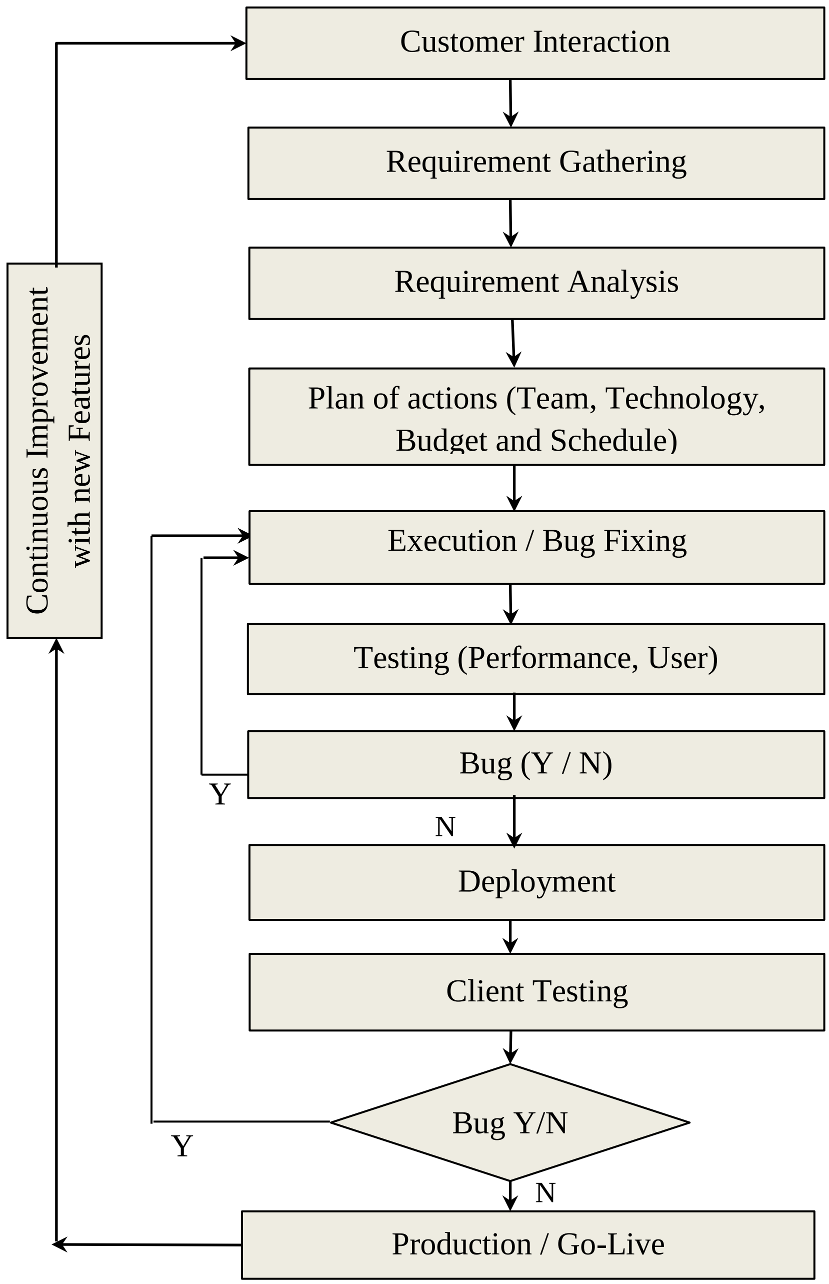


Figure 4.1: Stages of Product Based SD Organization

4.4.2 Software Development Stages in Project -based Organization

The stages in SD of project-based organizations (for 80 per cent new clients) are identified as customer requirement and understanding of these requirements by the execution team, feasibility, product requirement clarification, estimation of costs, design, sprint, navigational prototype (non-functional prototype), scrum, testing (performance), user acceptance testing and delivery. Figure 4.2 presents the stages of project-based organizations.

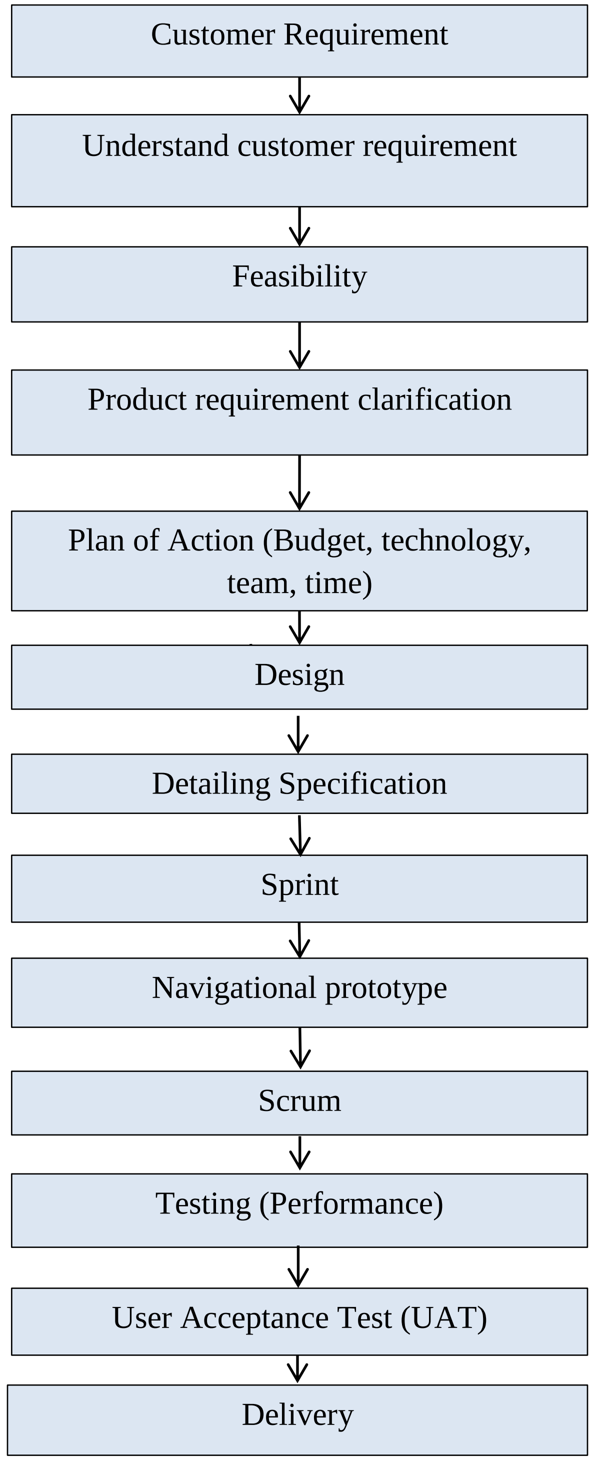


Figure 4.2: Stages of Project Based SD Organization

Womack and Jones (1996) discussed five lean principles: value, value stream, flow, pull and perfection in lean thinking. These case organizations were investigated with respect to these five principles. The five cases are as follows: ProdBased1, ProdBased2, ProjBased1, ProjBased2and ProjBased3 are described next.

4.5 Product-based organizations

4.5.1 Case 1: ProdBased1

Case organization1 is a product based organization. The case company is situated in the central part of Rajasthan (India) and develops sales automation software for life insurance and financial products. It also configures the software to suit the specific needs of its customers. This organization started in 1995, and currently, 260 employees are working for the company. The case organization has a partnership with industry experts and technology leaders like Microsoft to supply Web-based software solutions to cut down customers' expenses and to extend sales productivity which enhances the customer experience and improves promptness to market and service capabilities.

Different stages in product-based organizations are identified as customer interaction, requirement gathering, development stage, unit testing, quality engineering (QE) testing deployment and client testing.

For a new software product, the company first develops prototype programmes, so that software designer can get valuable feedback from the client early in product development. This prototype helps the client and the designer to compare whether the software is developed to match the software specifications. Usually, prototype programmes of SD products are developed within one year, but sometimes virtual prototypes (screens) are supplied by the clients to incorporate their requirements. During the conversation with one of the project managers, it was stated that "Requirement gathering consumes maximum time in software product development". To reduce lead time, project manager, customer and business analyst are involved in the planning phase. Testing for performance and user acceptance is carried out to check process-related issues. Reliable technologies such as Microsoft.NET and SQL Server are used. The product is mostly developed within the timeline.

4.5.1.1 Lean *Assessment*

In product based organization 1, lean assessment is carried out based on the adoption of lean principles based on lean practices. Level of adoption was observed in the case organization. In the current study project manager was asked to report the adoption level in the case organization. The statements were considered for adoption status in the case organization.

Out of the five lean principles, the case company adopted value, value stream, flow and pull. The value is enhanced for the client through bug fixing (correction of code), code revision and testing (QE). The other ways of achieving value include on-time delivery with added functionality and development within budget. Value is created by matching specification (requirement) with the final output. When a project manager (PM) was asked to prioritize between adding new technical features and removing live defects (bugs reported by the customer) at a time as a value-adding activity, he revealed that "priority is given to critical bug removal rather than new technical features being added to create value in product development".

The flow of information is maintained through intranet, portal, e-mail, and meetings. A separate portal is used in which issues are locked and the risk of incorporating incorrect requirements is minimized. Hence, the time taken to refine requirements is dramatically reduced. The work is pulled according to the capacity of the team. Further, small changes are incorporated in the requirement gathering and analysis phases to avoid any delay and major changes are not considered after sign off. The value stream is adopted at every stage of SD by code revision, documentation and correction of codes.

A flexible and cross-functional team, team-based problem-solving, event-driven product development and mistake-proof processes are completely adopted while visual control, standardized work, continuous improvement, JIT information, knowledge innovation visible planning (KIVP), pull and sequencing and collocation are extensively used. On the other hand, VSM, KANBAN/KANBAN board, DSM, cause and effect matrix, Obeya, reverse phase scheduling, Heinjunka, and SPC techniques are not adopted. A lean assessment of Prod based1 is summarized in Table 4.2

Table 4.2: Lean Assessment of Case1 (ProdBased1)

S. N.	Lean practices used	1	2	3	4	5	Degree to which practices are present
1.	Value Stream Mapping (VSM)	■					Not adopted
2.	KANBAN/KANBAN BOARD	■					Not adopted
3.	Visual Control				■		Extensively adopted
4.	Standardized Work				■		Extensively adopted
5.	Continuous Improvement				■		Extensively adopted in a view to process improvement
6.	Event driven product Development					■	Complete adoption
7.	Flexible and cross functional teams					■	Complete Adoption in a view to team empowerment
8.	Team Based Problem solving					■	Complete Adoption
9.	Set base Concurrent Development				■		Extensive Implementation
10.	JIT Information				■		Extensively adopted
11.	Visual Management				■		Extensively adopted
12.	Knowledge Innovation Visible Planning (KIVP)				■		Extensively adopted in knowledge sharing in SD projects
13.	Visual control Board	■					Not adopted
14.	Design Structure Matrix	■					Not adopted
15.	Cause and Effect Matrix	■					Not adopted
16.	Obeya	■					Not adopted due to unawareness of concept
17.	Reverse Phase Scheduling	■					Not adopted
18.	Heinjunka (Work load levelling)	■					Not adopted due to efficient utilization of human resources according to specialization
19.	SPC Techniques	■					Not adopted due to qualitative assessment
20.	Pull and Sequencing				■		Extensive adoption
21.	Collocation (Concurrent and integrated product development practices)					■	Complete adoption
22.	Mistake-Proof-Processes					■	Complete adoption to avoid mistakes in coding
(1-not adopted, 5- Complete Adoption)							

4.5.1.2 Issues in adoption of Lean

Some of the major issues observed in prodbased1 SD organizations in exploratory studies in this section. From the customer side, requirement change is a major problem and process-wise technology change and testing phase (QE) are critical. QE testing is performed on the request from the client after which the software product is frozen. In addition to QE, it is observed that repetitive testing is a major concern in this organization. The issues linked to the integration of planning, design, system, subsystem, testing, and delivery are also raised in this product-based organization. Coordination among different teams dealing with requirements, testing and development of software is another issue.

4.5.2 Case 2: ProdBased 2

This company is situated in Europe. It started its operations in 2016 and develops enterprise applications. The organization has ten employees with an average time span of product development ranging from two-six months. The different stages of product development are specification, design, development and support.

4.5.2.1 Lean Assessment

When asked about the time taken in different stages, the PM remarked that *"The development time is dominant in all stages of software product development"*. The company considers dynamic enterprise needs even upto the support (end of product life cycle) stage. To reduce development time, third-party modules are used. The third-party module saves time because it has already been developed and tested. In conversation, a PM stated that "*The third party module is selected by using cause and effect matrix"*. When it was asked to prioritize quality, cost and delivery, the PM answered that maximum weight is given to quality in comparison to delivery and cost. To improve flow, automation in testing and deployment is used. The value in case organization is created by considering customer requirements up to the final stage of the product development. The customer is allowed to participate in all stages of SD to improve their satisfaction. During the interview, one of the executives at case organization noted: *"different types of non-value activities (NVA) in SD projects are bugs and there is poor documentation in frameworks and third-party software"*. In SD, the maximum time is wasted in developing frameworks or choosing immature technology and documentation. One senior manager realized significant benefits in choosing best practices with *"picking mature frameworks/platforms/third-party modules"*. A mature framework is chosen because it is widely accepted by people. The developers come to know of their progress towards meeting the overall goal by making a feature breakdown. The use of e-mails and Jira software make the information flow smoother in SD. No lean practices are adopted in the company.

The case organization was focusing on agile based practices. Hence lean practices were not widely considered. Some of the lean practices were used in creating responsiveness. Lean principles like value and flow were discussed in the context of the customer perspective. Table 4.3 summarizes the lean assessment done in case 2 (Prodbase2). In agile software development, lean based component is extracted in case studies. The issues in prodbase2 are discussed in the subsequent section.

4.5.2.2 Issues in lean adoption

When asked about the issues in lean implementation in SD process, a PM commented that "team faces the problem of mismatch between requirements and implementation". However, "adoption of immature technology also creates a problem for the customer. The problems arising due to immature technologies are solved by communication, experience, and prototyping". PMs monitor progress up front and revise schedules weekly and biweekly. New features are incorporated into SD and bug fixing is carried out in the Production/ Go-Live environment based on priority and severity of the impact. The balancing of product features, quality and delivery schedule are also major issues. Continuous delivery and automated testing are done to improve project performance.

Table 4.3.Lean Assessment of case 2 (Prod Based2)

S. N.	**Lean practices used**	**1**	**2**	**3**	**4**	**5**	**Degree to which practices are present**
1.	Value Stream Mapping (VSM)						Not adopted ,Agile practices adopted
2.	KANBAN/KANBAN BOARD						Agile practices adopted
3.	Visual Control						Agile practices adopted
4.	Standardized Work						Agile practices adopted
5.	Continuous Improvement						Agile practices adopted
6.	Event Driven Product Development						Agile Practices adopted
7.	Flexible and cross functional teams						Agile practices adopted
8.	Team Based Problem solving						Agile practices adopted
9.	Set base Concurrent Development						Agile practices adopted
10.	JIT Information						Agile methods preferred
11.	Visual Management						Agile methods preferred
12.	Knowledge Innovation Visible Planning (KIVP)						Agile methods are preferred
13.	Visual control Board						Agile methods are preferred
14.	Design Structure Matrix						Agile methods are preferred
15.	Cause and Effect Matrix						Agile methods are preferred
16.	Obeya						Agile methods are preferred
17.	Reverse Phase Scheduling						Agile methods are preferred
18.	Heinjunka (Work load levelling)						Agile methods are preferred
19.	SPC Techniques						Agile methods are preferred
20.	Pull and Sequencing						Agile methods are preferred
21.	Collocation (Concurrent and integrated product development practices)						Not adopted, Only agile practices are preferred
22.	Mistake-Proof-Processes						Agile practices adopted
(1-not adopted, 5- Complete Adoption)							

4.6 Project- based Organizations

4.6.1 Case 3: ProjBased1

Case3 is a project-based organization. Case company is situated at central Rajasthan (India) and develops Web applications, mobile applications and also software for social media and IT support services. This organization is a project-based organization, where the average time span of a project is about one-two years. A total of 550 employees are working for the company.

For a project, the company first develops a non-functional prototype as a proof of concept (POC) to check the project's feasibility and to identify risk areas through interaction with the stakeholder. This prototype gives an overview of what is to be delivered in minimum possible time. Customer feedback at this level is important as it avoids changes/modifications in the deliverables ultimately reducing the project makespan which is considered to be an important aspect of lean. The pre-build solutions and open source technology (Perl/Ruby/PHP rather than expensive technology like Java, Oracle, and .NET) are used to build the deliverables more economical. Cross-functional teams are made that work on different modules leading to rapid development. Continuous process improvement takes place by using new and better-operating methods such as platform upgradation and technology review.

4.6.1.1 Lean Assessment

Out of the five lean principles, value and flow are of major concern in the case company. Value is provided to the customer by reducing time and cost and improving quality. In all SD stages, application development (planning) takes around 70 percent of the time, which is the maximum of all stages. In the planning stage, the design freezes, wherein documents are signed by the customer. In this stage, fix requirement and fix cost are also considered. Questions were asked to identify sources of waste in the SD projects. During the conversation with one of the project managers, it was stated that "Maximum time is wasted in execution stage" which can be reduced by stressing more on requirement gathering and thus "In software development requirement gathering is not considered as waste". Further, value is created through bug fixing and incorporating new features. These two, however, are to be prioritized depending on the customer requirements.

Defects are of two types: technology-related (error or bugs) and customer-related. These defects can be eliminated by using the project management system (PMS) in which time management, task assignment, and best possible resources are taken into consideration. Quality is improved in SD by codes through the model view controller or object-oriented base.

An uninterrupted flow of information in the process of SD is considered to be of importance, and is achieved through "Project Charter." Further, the information system for communication for different modules among project managers, team leaders, functional managers as well as customers is facilitated by PMS. Base Camp and Jira are the tools used to maximize the flow of work and information. It is concluded that providing customer value is of top priority, then comes maximizing flow through tools and finally, debugging. Sometimes a trade-off between value and flow is also made. The clear requirement and better execution reduce the development time, which helps to attain the highest customer value and maximizes workflow. The changes in customer demand are accommodated by the adoption of a flexible platform (technical/framework).

VSM, Visual Management, Kanban/ Kanban board, DSM, mistake-proof-processes are completely adopted, while flexible and cross-functional teams, team-based problem solving, JIT Information, Pull and Sequencing, Continuous improvement, visual control are extensively used in the company. On the other hand, standardized work, KIVP, set-based concurrent development and cause and effect matrix are adopted to a lesser extent, while VCB, SPC Techniques, event-driven product development and Obeya are not used at all. Table 4.4 summarizes the lean assessment in case3

4.6.1.2 Issues in adoption of Lean

The major issues in SD processes in the company include resource constraints (team availability) and technology constraints. From the customer perspective, major issues are vague requirements or changes in customer requirements. Customer-related problems are solved by requirements analysis, in which software requirement documentation (SRD) is used. SRD is developed through mutual discussion and then finalized, after which no major changes are allowed.

.

Table 4.4: Lean assessment of Case 3 (ProjBased1)

S. No.	Lean practices used	1	2	3	4	5	Degree to which practices are present
1.	Value Stream Mapping (VSM)					■	Complete Adoption
2.	KANBAN/KANBAN BOARD					■	Complete Adoption
3.	Visual Control				■		Extensively adopted
4.	Standardized Work				■		Extensively adopted
5.	Continuous Improvement				■		Extensively adopted
6.	Event Driven Product Development	■					Not adopted
7.	Flexible and cross functional teams				■		Extensively adopted
8.	Team Based Problem solving				■		Extensively adopted
9.	Set base Concurrent Development			■			Some adoption
10.	JIT Information				■		Extensively adopted
11.	Visual Management					■	Complete Adoption
12.	Knowledge Innovation Visible Planning (KIVP)			■			Some adoption
13.	Visual control Board	■					Not adopted
14.	Design Structure Matrix					■	Complete Adoption
15.	Cause and Effect Matrix			■			Some adoption
16.	Obeya	■					Not adopted
17.	Reverse Phase Scheduling		■				Little adoption
18.	Heinjunka (Work load levelling)	■					Not adopted
19.	SPC Techniques	■					Not adopted
20.	Pull and Sequencing				■		Extensive adoption
21.	Collocation (Concurrent and integrated product development practices)					■	Complete adoption
22.	Mistake-Proof-Processes					■	Complete Adoption
(1-not adopted, 5- Complete Adoption)							

4.6.2 Case 4: ProjBased2

This SD Company is situated in Europe with 200 employees on the roll. The company is engaged in developing banking software solutions.

4.6.2.1 Lean assessment

Out of five lean principles, value and flow are used in the case organization. Bugs and errors and information asymmetry are considered to be major NVA in projects in this case company. Thus, bug counts and delay periods are reduced to increase customer value. Additionally, the company considers problems occurring during the development and use of the deliverable as NVA. Regarding the same, a project manager observed that "*Requests, services, accounts, passwords, and installation of software are non-value activities, which amount to 50 percent of total activities"*.

Flow is achieved through automation tools such as VSTS, Jenkins, Powershell and Chef and Windows Containers/Docker. Docker is used in SD projects for packaged deployment. DevOps is used to achieve flow and integrated processes; ultimately resulting in reduced repetitive testing and technology upgradation in SD. DevOps also creates value in the project by reducing project makespan. However, DevOps process is adopted by the combined use of lean/agile practices: Kanban and scrum along with operations. This process achieves continuous integration and continuous development in which real-time knowledge sharing takes place. Flow is also improved by automation testing such as Salenium (software testing framework) and team foundation server (TFS). TFS represents a set of collaborative SD tools that incorporate with the prevailing editor, thus enabling the cross-functional team to work efficiently on software projects of all sizes. For improvement of flow, cumulative flow diagrams (CFD) have been a standard part of reporting in TFS. CFD plot an area graph of cumulative work items in every state of workflow and are rich in information for calculation of cycle time and throughput rate.

Lean practices such as VSM, Kanban Board, flexible and cross-functional teams and team-based problem-solving are completely used in the case organization. Meanwhile, lean practices like visual control, standardized work, continuous improvement, visual management, set-based concurrent development, JIT Information, KIVP, DSM, cause and effect matrix and pull and sequencing are also extensively adopted. Table 4.5 presents the lean assessment of case 4.

Table 4.5: Lean Assessment of Case 4 (ProjBased2)

S. N.	Lean practices used	1	2	3	4	5	Degree to which practices are present
1	Value Stream Mapping (VSM)					■	Complete adoption
2	KANBAN/KANBAN BOARD					■	Complete adoption
3	Visual Control				■		Extensive adoption
4	Standardized Work				■		Extensive adoption
5	Continuous Improvement				■		Extensive adoption
6	Event Driven Product Development	■					Not adopted
7	Flexible and cross functional teams					■	Complete adoption
8	Team Based Problem solving					■	Complete adoption
9	Set base Concurrent Development				■		Extensive adoption
10	JIT Information				■		Extensive adoption
11	Visual Management				■		Extensive adoption
12	Knowledge Innovation Visible Planning (KIVP)				■		Extensive adoption
13	Visual control Board	■					Not adopted
14	Design Structure Matrix				■		Extensive adoption
15	Cause and Effect Matrix				■		Extensive adoption
16	Obeya	■					Not Adopted
17	Reverse Phase Scheduling	■					Not adopted
18	Heinjunka (Work load levelling)	■					Not Adopted
19	SPC Techniques	■					Not adopted
20	Pull and Sequencing				■		Extensive adoption
21	Collocation (Concurrent and integrated product development practices)					■	Complete Adoption
22.	Mistake-Proof-Processes					■	Complete Adoption
(1-not adopted, 5- Complete Adoption)							

4.6.2.2 Issues in Lean Adoption

When asked about the issues in SD projects, a PM described: [. . .] *vagueness in the requirement, pinpointing the problems, difficulty in convincing customers, additional effort in change requests, bugs removal, ineffective progress measurement, and ineffective requirement gathering. Sometimes effort duplication is also observed as an issue. The effort duplication is reduced using interface, coding, limited sharing, high-level sharing software, automated code review, and open source movement code.*

4.6.3 Case 5: ProjBased3

It is an IT service company situated in India. There are 150 employees working for the company. The company is engaged in SD, consulting and providing software solutions to businesses worldwide. The projects are related to services such as healthcare and banking.

Contrary to other cases, this company does not develop prototypes for requirements gathering and risk analysis, except when there is a challenge that calls for the same. The project team decides about days of working, methodologies, technical architecture and criticality of the client, budgeting of the project, checking and application of security in its kick-off meeting. The practices such as code libraries, automation tools as well as continuous integration and easy test are adopted to make the deliverables more economical. Feedback from the customer is taken and root cause analysis is carried out. The technical problem is resolved and any misunderstanding is removed at regular intervals to improve customer value.

4.6.3.1 Lean Assessment

The company creates value to the customer by instant delivery, increased variety and low cost. Customer requirements are gathered by incorporating customer expectations, data requirement, suggestions by customer and ideas from intelligent customers regarding technical and business aspects and management information system (MIS) through business analysts.

One of the project managers, during the interview, revealed:

In our organization, 30 percent time is consumed in requirement gathering, 40 percent time in coding and 30 percent in testing (system, integration). Sometimes customers demand additional *tests such as load test, integration, and automation test, which take upto 50 percent of project time.* Hence, testing is a very time-consuming activity in SD projects. The other things that consume time include the wrong selection of methodologies which reduces quality and increases rework. Project management tools are employed to monitor the process and root cause analysis is performed to identify NVA in SD projects. Requirement analyses along with code review are adopted to reduce NVA in SD projects.

The flow in SD is achieved by information management, continuous information, and dedicated software tools. The flow of uninterrupted (continuous) information in linking processes of SD is maintained thorough flow creating tools such as Jira, MS Project and video conferencing.

Visual control, standardized work, continuous improvement and JIT information are completely adopted, whereas VSM, VCB, visual management, set-based concurrent development, KIVP, cause and effect matrix and pull and sequencing are extensively used. On the other hand, KANBAN, reverse phase scheduling, Heinjunka and SPC techniques were adopted to a lesser extent. On the other hand, event-driven product development, DSM, Obeya and mistake-proof processes are not adopted at all. Table 4.6 summarizes the lean assessment of Case 5.

4.6 .3.2 Issues in the adoption of Lean

The customers' concerns about delivery and project status reports are considered to be major issues in this case organization. To capture accurate customer requirements and then to integrate them into design and development is another issue. Bug leakages are an issue where the customer finds bugs. The techniques such as code review, peer review, automatic code review, testing, delivery and bug leakages are used as validation and verification of SD process and to identify issues.

Problems do not surface until long after the code is written. The problems of customers are solved by weekly calls and monthly calls informing customers well ahead of time regarding delivery and project status. The parts of the software are delivered from time to time to the customers as per their requirements.

Table 4.6: Lean Assessment of Case 5 ((ProjBased3)

S. N.	Lean practices used	1	2	3	4	5	Degree to which practices are present
1.	Value Stream Mapping (VSM)				■		Extensive adoption
2.	KANBAN/KANBAN BOARD			■			Some adoption
3.	Visual Control					■	Complete Adoption
4.	Standardized Work					■	Complete Adoption
5.	Continuous Improvement					■	Complete Adoption
6.	Event driven product development	■					Not adopted
7.	Flexible and cross functional teams					■	Completely
8.	Team Based Problem solving					■	Completely
9.	Set base Concurrent Development				■		Extensive adoption
10.	JIT Information				■		Extensive adoption
11.	Visual Management				■		Extensive adoption
12.	Knowledge Innovation Visible Planning (KIVP)				■		Extensive adoption
13.	Visual control Board				■		Extensive adoption
14.	Design Structure Matrix	■					Not adopted
15.	Cause and Effect Matrix				■		Extensive adoption
16.	Obeya	■					Not adopted
17.	Reverse Phase Scheduling			■			Some adopted
18.	Heinjunka (Work load levelling)			■			Some adoption
19.	SPC Techniques			■			Some adoption
20.	Pull and Sequencing				■		Extensive adoption
21.	Collocation (Concurrent and integrated product development practices)	■					Not adopted
22.	Mistake-Proof-Processes	■					Not adopted
(1-not adopted, 5- Complete Adoption)							

4.7 Case discussion and Analysis

4.7.1 Level of adoption of lean principles in software development projects

It is observed that in case organizations out of five lean principles and responsiveness/agility, value and flow are adopted in the majority of organizations. Value and flow enhancing lean practices are discussed in the next section. Prodbased2 case organization adopts responsiveness, which is an indicator of lean and agility. Remaining four case organizations focus on value and flow principles. The important outcomes of lean initiatives in product and project-based organization SD organizations are summarized in Table 4.7. The comparison among cases is based on the adoption level of lean principles in software development.

Table 4.7: Level of adoption of lean principles

Case Organizations	Lean Principles					Responsiveness
	Value	Value Stream	Flow	Pull	Perfection	
Case1(ProdBased1)	✓	✓	✓	x	x	✓
Case 2 (ProdBased2)	✓	x	✓	x	x	✓
Case 3 (ProjBased1)	✓	x	✓	x	x	X
Case 4 (ProjBased2)	✓	x	✓	x	x	✓
Case 5 (ProjBase3)	✓	x	✓	x	x	X
✓ represents Adoption of lean			x- Not adopted			

Value is achieved using technology/tool like project management tool and root cause analysis Flow is achieved through automation, third-party module, mature framework, project charter, Basecamp, Jira, TFS, CFD, Microsoft project, prebuild solutions and open source technology. Out of five cases, three cases focus on adoption of responsiveness.

4.7.2 Level of adoption of lean practices in software development projects

Apart from identifying the prominent lean practices in SD projects, the level of adoption of the same is also examined in this research. The respondents in the case organizations were asked to indicate the level of adoption of lean practices in terms of whether the practice is being used completely, extensively, to some extent, little or not at all. Figure 4.3 shows the level of adoption in the case organizations.

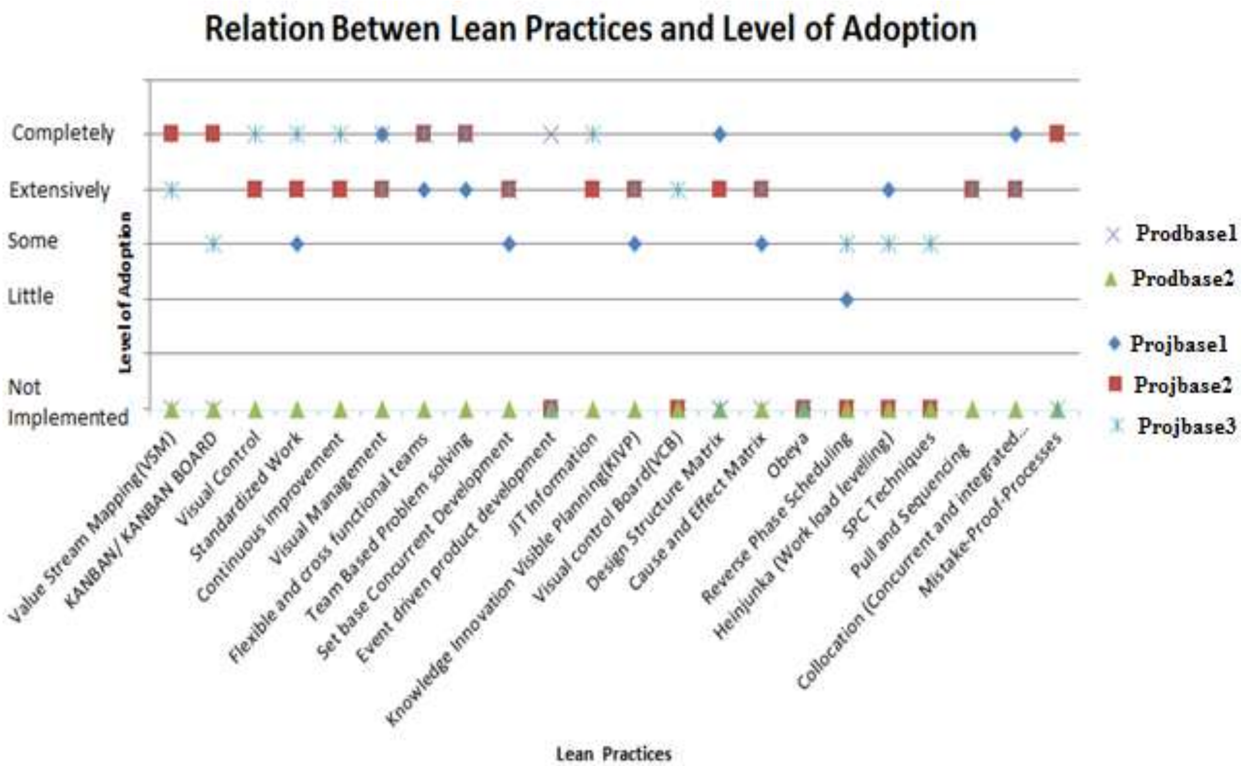


Figure 4.3: Relationship between level of adoption and lean practices

It was observed that visual control, standardized work, continuous improvement, flexible and cross-functional teams, team-based problem-solving, mistake proofing, set-based concurrent development, JIT information, KIVP, pull and sequencing are extensively used in both product and project-based organizations. Flow creating practices such as Kanban, DSM, concurrent and integrated product development, Heijunka and cause and effect are extensively adopted in project-based organizations but are not conspicuous in product-based organizations. Adoption of flow creating practices in software product development may be less due to its distributed and asynchronous nature which obstructs implementation of flow creating practices (Maglyas *et al.*, 2012). Further, adoption of Kanban in product-based organizations creates a bigger challenge in achieving end-to-end flow due to the number of decisions involved in the processes which create handovers (Rodríguez *et al.*, 2013). Similarly, concurrent and integrated product development, Heijunka and cause and effect matrix are not adopted extensively in product based organizations. This may be due to difficulty in the transfer of knowledge (like knowledge about the customer and the code) across the team. It is also noted that lean practices such as event-driven product development, visual control board, reverse phase scheduling, and obeya were not popular in almost in all the cases.

4.7.3 Cross case comparisons

In both product and project SD organizations, some important lean practices viz. visual control, standardized work, continuous improvement, flexible and cross-functional teams, team-based problem solving, visual management, mistake proofing, pull and sequencing were observed. These practices are graphically represented in the previous section.

The value and flow in the SD projects are found to be achieved through a number of lean practices borrowed from manufacturing. However, there are few practices which are typical to SD projects and not common in manufacturing.

The notion of value to the customer is the utility provided to the customer at the correct time at an appropriate price as outlined by the customer. The value in software projects is delivered through bug fixing, new features and security features (non-functional).

Flow in SD projects emphases on the flow of information between the project team and the customer rather than physical flow and information flow in manufacturing. Poppendieck and Poppendieck (2006) argued that great software products are the result of frequent transfer of preliminary information among different development stages of software.

The lean practices being used in SD projects for enhancing value and flow along with literature support are summarized in Tables 4.8 and Table 4.9, respectively.

Table 4.8: Value Enhancing Lean Practices

Value enhancing lean practices	References	Case Organizations
Mistake Proofing (developing mistake proof code)	Poppedieck & Poppedieck, 2006	ProjBased1,ProjBased2, ProdBased1
VSM for determining waste	Middleton and Joyce, 2010;Stass *et al.*,2011;Anand *et al.*,2014	ProjBased1,ProjBased2, ProjBased3
Team based problem solving	Stass *et al.*,2011	ProjBased1,ProjBased2, Projbased3;ProdBased1
Continuous Improvement	(Anderson2010, Poppedieck&Poppedieck, 2006;Samanta and Mani,2015)	ProjBased1, ProjBased2, ProjBased3;ProdBased1
Standardized work	Stass *et al.*, 2011 and Middleton, 2010	ProdBased1, ProjBased1, ProjBased2, and ProjBased3
Visual Control	(Middleton,2010: Karvonen *et al.*, 2012 and ,Pernstal *et al.*,2013)	ProjBased1,ProjBased2, ProjBased3, and ProdBased1

The lean practices being used in SD projects for enhancing flow along with literature support are listed in Table 4.9.

Table 4.9: Flow Enhancing Lean Practices

Flow enhancing lean Practices	References	Case Organizations
Visual Management for task prioritization	Pernstal *et al.*(2013)	ProdBased1, ProjBased1, ProjBased2, and ProjBased3
KANBAN (Continuous flow of work) KANBAN BOARD (Virtual KANBAN system and Visual Control)	Anderson(2003,2010); Middleton ,2010 ;Ikonen *et al.*,2011; Rodríguez *et al.*,2013	ProjBased1,ProjBased2, ProjBased3
DSM to optimize information flow	Stass *et al.* (2011)	ProjBased1, ProjBased2
JIT Information	David Anderson's 2010; Ikonen *et al.*,2011	ProjBased1,ProjBased2, ProjBased3,Prodbased1
Flexible and cross functional team	Middleton and Joyce, 2012;Samanta and Mani,2015	ProjBased1,ProjBased2, ProjBased3;ProdBased1
Cause and effect matrix (current state of process)	Stass *et al.* (2011)	ProjBased2, ProjBased3
Set base Concurrent Development	(Hafer, 2011; Middleton, 2005;Poppendieck and Poppendieck,2006).	ProjBased2,ProjBased3; ProdBased1

4.7.4 Issues in implementing lean in software development

Various issues during the implementation of lean in SD projects were identified. The issues are classified into customer-related, process-related and people-related issues and are shown in Table 4.10. These issues will be helpful for project managers to develop a robust lean implementation plan.

Customer focus is central to lean philosophy. The success of any SD organization depends on customer satisfaction and close collaboration with the customer. In SD, two kinds of customers are considered: external customer or end user and internal customer. Vague requirements and frequent changes in the requirements are observed to be the most important issues in lean implementation. The other important issue is managing technical specifications. Project manager of Prodbase2 stated that *"Team faces the problem of mismatch between requirements and implementation"*. The change is an enemy in lean rather than being an ally in agile (Boehm and Turner, 2005). In ProjBased 3, a project manager categorically stated that *"customer related issue is a major issue because 30 percent of the time is consumed in requirement gathering"*.

Repetitive testing and debugging were found to be the major process-related issues as they increase the development time and delay the release. In ProjBased 3, a project manager observed that "50 percent of project time is consumed in testing". The other major process related issues include the prioritization of bug fixing and the addition of new features in SD. If the requirement changes are frequent, then technology constraints such as platform and hardware issues become prominent. In ProjBased 2, a PM stated that *"50 percent of non-value activities in a project have been found to be related to requests, services, after delivery account, password and installation of software"*.

People-related issues are also vital to the adoption of lean in SD projects due to due to its collaborative nature. Like other management initiatives, developing an effective cross-functional team is a challenge in lean implementation. Poppendieck and Poppendieck (2006) argued that excellent software products start with highly competent technical experts in many areas: architecture, object-oriented technologies, coding strategies, data structures, and test automation. However, the teams are usually involved with working in silos on tasks assigned by PMs and an "I did as was told to" way of working is common. Further, lack of clarity about responsibilities and measurement of deliverables are quite common in SD projects which may be because of tacit knowledge, lack of communication and the existence of silos. Due to a high level of tacit knowledge in SD (Poppendieck and Poppendieck, 2006; Staats *et al.*, 2011; Anand and Kodali, 2010a), it is challenging to communicate and hand off knowledge to other people. Table 4.10 summarizes the comparison of issues in SD projects among five case organizations.

Table 4.10: Comparison of Case Organizations for Issues in SD

Issues	ProjBase1	ProjBase 2	ProjBase 3	ProdBase1	ProdBase 2
Customer Related	Change in customer requirement and vague requirement	Vague requirement	Capturing accurate customer requirement and integrating them into product design and development.	Change requests after product delivery	Mismatching between requirements and implementation
Process Related	Testing, bug fixing and adding new features, managing non-functional requirements	Non- value activities such as with requests, services, After delivery-Account, password and installation of software.	Bugs leakages	Repetitive testing, prioritization between bug fixing and adding new features	Developing frameworks and non-availability of mature technology
People Related	-Resource Constraints (team) -Coordination among Cross-functional Team	Coordination among cross-functional team	Communication among organization, Business Analyst and customer	Resource Constraint	Communication between team and customer

4.8 Summary

This research provides an overview of implementation of lean principles in SD organizations. Case research method was applied to examine the level of adoption of lean in SD projects. Further, the issues related to the implementation of lean in SD projects have been highlighted. Five case studies were undertaken, which were categorized into project-based and product-based organizations. Lean has its roots in discrete manufacturing with five lean principles: value, value stream, flow, pull and perfection. The proponents of lean have claimed these principles to be universally applicable. The case studies undertaken in this research provide some evidence that lean principles are being adopted in SD projects, but the application of all five principles is not visible. This research reveals that value and flow are more relevant in SD organizations while value stream, pull and perfection are not conspicuous. This may be due to intangible deliverables, non-repetitiveness, distributed and asynchronous nature of the SD process.

This study identifies the major lean practices adopted in SD organizations corresponding to product-based and project-based organization clarifying which practices are driven by the value principles and which practices are driven by flow principles. The findings suggest that some of the lean practices such as visual control, standardized work, continuous improvement, flexible and cross-functional teams, team-based problem-solving, mistake proofing, set base concurrent development, JIT information, KIVP, pull and sequencing which do not depend on the characteristics of the organization and are largely applicable to both types of organizations. On the other hand, practices like event-driven product development, VCB, Obeya, reverse phase scheduling, and SPC techniques are not being applied in SD organizations. Further, in flow-oriented practices such as VSM, Kanban, DSM, concurrent and integrated product development, Heijunka and cause and effect matrix have been applied to project-based organizations with limited applications in SD product-based organizations.

Finally, the issues related to the implementation of lean in SD have been studied. These issues are categorized as customer-related, process-related and people, i.e. team-related. The mismatch between requirements and specifications, repetitive testing and debugging, prioritization between bug fixing and adding new features, organizational silos inhibiting communication are found to be major issues.

Although all care has been taken in this research to avoid overlooking any detail, yet due to the nature of case methods, the findings may not be applicable in general for other SD organizations. To generalize, it would be worthwhile to conduct empirical studies in larger number of organizations. In the next chapter, empirical studies are taken up to investigate the lean adoption with all core lean principles.

Chapter 5

Descriptive Analysis

Chapter 5

Descriptive Analysis

5.1 Introduction

The gathered data from various software organizations are investigated, which comprises of a statistical analysis, management of missing data, examination of normality, investigation of outliers, reliability and validity of the variables. The descriptive analysis is carried out to discover the broad notices about the data gathered concerning the lean principles and responsiveness in software development projects.

Apart from investigation of data, analysis is continued with concerned study employing structural equation modeling (SEM) to covering the research hypotheses.

5.2 Data Coding

According to Coakes *et al*, (2005), codification of data is mandatory and prerequisite of any inferential statistical computation. The codification covers appropriate numbering of constructs, their stages and values. Table5.1 presents the main variable, item details and their coding. The code of every item is required for descriptive analysis.

Table 5.1: Coding of variables and references

Construct	Variable Code	Variable	References	Mean	SD
Value	VL1	We use standardized tasks/parts clearly defining substance, order, timing, and desired result and processes for continuous improvement.	Spear and Bowen, 1999; Liker,(2004); Meiling, (2010); Staats *et al.*, (2011); Olivella *et al*., 2008; Malmbrandt and Ahlstrom, 2013)	5.99	1.035
	VL2	Whenever defects are identified, it is logged and immediate actions are taken for fixing the problem and to obtain the desired quality on the first try/pass.	Malmbrandt and Ahlstrom,(2013); Poppendieck and Poppendieck (2007)	6.00	1.110

	VL3	We update, prioritize the backlog and incorporate new features throughout the entire programme of SD	Rodrieguez,2013	5.82	1.008
	VL4	We frequently remain in close contact with our customer to know about their exact requirements.	Spear and Bowen, 1999; Anand and Kodali, 2009; Malmbrandt and Ahlstrom, 2013; Spear and Bowen, 1999	6.05	1.005
	VL5	Our customers give feedback on quality and delivery performance of the software.	Anand and Kodali, 2009; Malmbrandt and Ahlstrom, 2013)	6.06	1.086
Value Stream	VS1	We determine those activities that do not contribute value to the product or service and redesign work process to eliminate these tasks/processes	Mujtaba et al. (2010); Musat and Rodríguez (2010)	5.55	1.245
	VS2	We frame current and future development process diagram capturing processing time, delay time, number of people involved	Peterson & Wohlin (2010) and Anand et al. (2014)	5.44	1.309
	VS3	We identify all the sources of waste in SD such as delays, unnecessary rework due to errors, defects, extra features and partial work done and track them	Staats and Upton, (2011); Malmbrandt and Ahlstrom,(2013)	5.33	1.442
	VS4	We analyse and design the workflow required to bring a software to a customer	Anderson (2010); Peterson & Wohlin (2010)	5.35	1.293
Flow	FL1	We have developed subsystems to handle dependencies of the tasks/modules and their ordering	Stass et al. (2011), Middleton(2005)	5.88	1.149
	FL2	We consider conceptual solutions in parallel and narrow to develop a single focused idea for achieving the final specifications	Poppendiecks and (2003);Poppendieck and Poppendieck (2007) , Jonsson (2012)	5.82	1.177
	FL3	We have developed direct pathway and connections among people and tasks in order to have smooth and timely flow of information	Staats and Upton, (2011); Staats et al., (2011)	5.75	1.213
	FL4	Workload is leveled and balanced according to available resource and features to meet sprint demand (scrum /stand up for assigning task)	Meiling, (2010); Malmbrandt and Ahlstrom, (2013)	5.79	1.192
	FL5	Team members collaborate with each other for achieving goal	Meiling, 2010	5.27	1.216
	FL6	Process parameters are displayed on workplace on visual dashboard/ electronic dashboards to improve the process and progress of work	Pernstål et al. (2013)	5.43	1.421

Pull	PL1	Development of software workplace is pulled by the demand of the next stakeholder in the organization	Petersen and Wohlin, (2011)	6.04	1.086
	PL2	We use signal cards which contains specification of job, detailed designs, verifications and implementation of the story and help to know what to do next task within stipulated time	Anderson(2003), Ikonen et al.(2010); Anderson (2010); Middleton et al.(2010)	5.64	1.288
	PL3	Work is assigned according to available capacity in my area of operations	Hopp and Spearman, 2004	5.94	1.092
	PL4	Team members pull the items/ tasks from the backlog according to priority	Poppendieck and Poppendieck,2003	5.46	1.277
Perfection	PF1	Our organization adopt continuous improvement and continuous learning techniques	Alves et al. (2012)	5.77	1.232
	PF2	Employees are actively involved in problem solving	Spear and Bowen (1999); Liker (2004)	5.59	1.536
	PF3	Information for the execution of my work is displayed visually at my work and visible for every team member	Rodríguez et al., 2013; Malmbrandt and Ahlstrom (2013)	5.06	1.332
	PF4	We capture online/documented feedback from customers	Liker, (2004); Meiling, (2010)	5.57	1.483
Responsiveness	RS1	We are highly capable of responding to customized requirements to handle dynamicity	Poppendieck and Poppendieck (2007); Rodrı´guez et al. (2013)	5.98	1.143
	RS2	We use practices such as iterative, standardized error code, continuous integration, test driven development, and daily standup meetings for expediting the SD project	Stass et al., 2011; Wang et al. (2012)	5.76	1.332
	RS3	We use scrum in SD project to track progress	Wang et al. (2012); Rodrı´guez et al.(2013)	5.79	1.440
	RS4	We use periodic builds and code reviews	Stass et al. (2011)	5.84	1.197
Operational Performance	OP1	We have less work in progress (WIP) when a set of features are designed, coded and tested.	Anderson (2010)	4.98	1.863
	OP2	We have fewer complaints from customers regarding desired quality and service	Middleton and Joyce (2012)	5.20	1.631
	OP3	We generally complete the project within its estimated cost		5.52	1.683
	OP4	Generally, we supply product/ deliverable on the agreed upon time.	Middleton and Joyce (2012)	5.46	1.630

Table 5.2 represents the questionnaire with the specific lean principles/practices. Questions were designed based on lean practices and responsiveness captured from literature and exploratory case studies.

Table 5.2: Questionnaire with specific lean principles/practices

Q. No.	Related Lean principles	Principle code	Specific Lean practices
Q16	Value	VL4	Direct customer connection
Q13	Value	VL1	Standardized work, task, stable and predictable process
Q14	Value	VL2	Mistake proof
Q17	Value	VL5	Customer feedback
Q15	Value	VL3	Update and prioritize the backlog, incorporate new feature
Q33	Perfection	PF2	Online feedback capturing
Q22	Flow	FL1	Design Structure Matrix (DSM)
Q23	Flow	FL2	Set Based Concurrent Development SBCD)
Q25	Flow	FL4	Heijunka ,Workload is leveled and balanced/ Workload balancing
Q29	Flow	FL6	Visual management: Visual dashboards/ electronic dashboards
Q28	Flow	FL5	Cross-functional teams (CFT)
Q20	Value Stream	VS3	Waste identification
Q18	Value Stream	VS1	Determination of NVA and elimination
Q24	Flow	FL3	Direct Pathway and connection (JIT)
Q26	Perfection	PF1	Continuous improvement and continuous learning
Q29	Perfection	PF4	Online Documentation
Q27	Perfection	PF2	Employee involvement in problem-solving in product and process
Q26	Flow	FL5	Team collaboration
Q36	Responsiveness	RS3	SCRUM
Q39	Responsiveness	RS4	Periodic Builds and periodic Code Reviews
Q36	Responsiveness	RS1	Responding on customized requirements
Q37	Responsiveness	RS2	Iterative method
Q19	Value Stream	VS2	Development of Current and Future process diagram
Q21	Value Stream	VS4	Analyze and design the workflow
Q28	Perfection	PF3	Transparent processes with visibility of information
Q23	Pull	PL2	Kanban Board
Q22	Pull	PL1	Pulling according to stakeholder
Q25	Pull	PL4	Pulling of task
Q24	Pull	PL3	Work distribution according to tasks

These practices represent various lean principles. As per outcome of exploratory case studies, value and flow was adopted in major case organizations. Hence the adoption level of core lean principle along with responsive was investigated in larger number of SD projects.

5.2.1 Missing Value analysis

Missing data, where valid values on one or more variables are not available for analysis. From a practical standpoint, the missing data can become quite problematic in terms of reducing the sample size. The data imputation is used to manage missing data value. IBM AMOS 26 Software presumes that a data value that is missing entirely at random, permits with substituting approximates value that is effective and logical. Hair *et al.*, 2006 proposed “If remedies for missing data not applied, any observation with missing data on any of the variables will be excluded from the analysis”.

5.2.2 Outliers

Outliers are observations with a unique combination of characteristics identifiable as distinctly different from the other observations. Leech *et al.* (2006) revealed that the existence of outlier may considerably influence on the model estimation for model fit and reliability. Hair et el. (2006) proposed “The critical levels (Mahalanobis distance D2) for the measure should be less than 3 or 4 in larger samples”. In current study no outlier was detected when tested with SPSS 26.0. The Pearson correlation coefficient can be severely affected by a single outlier on *data set* (Anderson & Schumacker, 2003)

5.2.3 Non response bias

In non- response bias the comparison between early and late respondents are done for significant difference. Lambert and Harrington (1990) categorically stated “Comparison should be made based on the assumption that the late respondents were considered as non–respondents”. In this study, a total of 256 respondents were divided in 136 (53%) and late 120 (46.87%) respondents. Performing t-test in early and late respondent, it was observed that comparison was insignificant at 5% significance level. Hence in current study, problem of non -response bias did not exist.

In survey methodology, a survey instrument was developed based on literature and experts from IT industries/ SD organizations. The questionnaire was devised to probe the research objectives. A set of 51 questions was used.

5.3 Survey Observations

The survey was carried out in Software development organizations. The questionnaire was administered to a sample of 1151 in SD organizations through Email and post. Respondents were senior management, team leads, project manager, business analyst, team member, tester and developer. A total of 256 responses were received (response rate 22%) which were used for further analysis.

5.4 Responses rate and Respondent's characteristics

Descriptive analysis is carried out to test the rate of responses, respondent's profile and to examine the mean and standard deviation which will benefit in further statistical analysis. The data has been gathered from 256 software development organizations together with product and project base SD organizations. These organizations comprises consulting, web services, and E-commerce covering domain of enterprises, banking and insurance. All the demographic profile for the respondents and data are shown in Table 5.3, Figure5.1, Figure 5.2; Figure 5.3; Figure 5.3; Figure 5.4, Figure 5.5, Figure 5.6; Figure 5.7; and Figure5.8.

Table 5.3: Demographic profile for the respondents

Demographic data	
Type of SD Organization	**Percentage of Total Sample**
Product Based	27.9
Project Based	25.7
Both Product and project	39.0
Other	
Software Services	
Consulting	30.8
Web Services	25
E- commerce	12.5
Other	

Domain Specific	
Enterprise Application	52.4
Banking Software Solution	8.1
Insurance	4.8
Other	
IT Support Services	
Business Processes	60.3
Information Security	19.8
Knowledge Management	7.8
Other	
Deliverable of your project	
Complete standalone software	78.8
Partial software product	15.2
Other	
Position of the Respondents in organization	**Percentage of total sample**
Senior Management	10.7%
Team Leads	15.2%
Project Manager	12.3%
Business Analyst	4.9%
Team Member	13.6%
Tester	8.0%
Developer	18.4%
Annual Turn over	
< 5 Million	12.9 %
5-10 Million	8.1 %
10-1 Billion	16.1 %
Over 1 Billion	58.9%
Number of Employees	
<100	12.6
101-250	8.1
251-500	8.0
501-1000	7.6
>1000	63.7

Section A represents the demographic information about the firm and respondents

I. Respondents according to type of SD organizations

Figure5.1 represents the types of software Organizations with percentage responses.

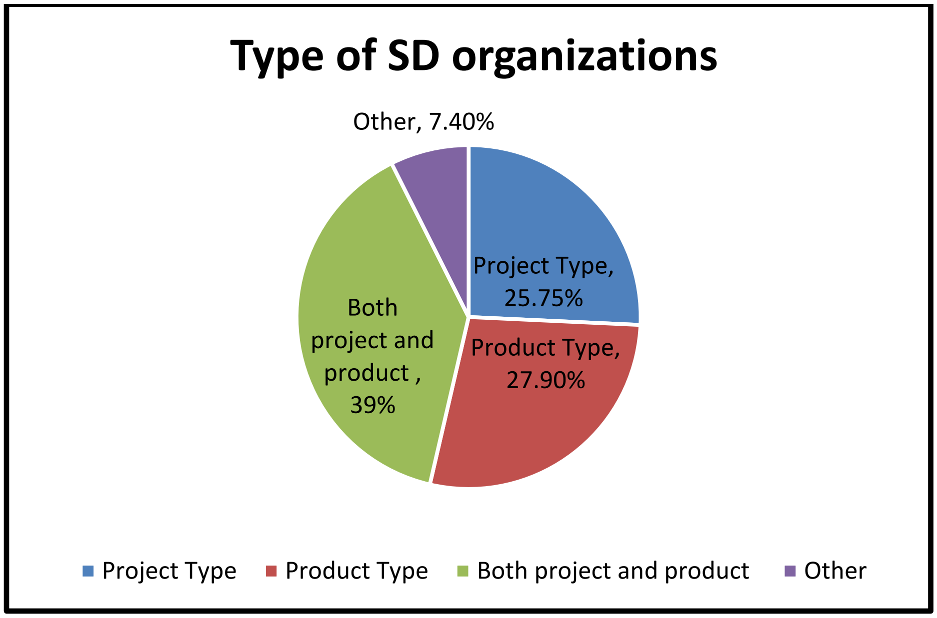


Figure 5. 1: Types of software Organizations

II. Software Services

Figure5. 2 represents types of software Services with percentage responses

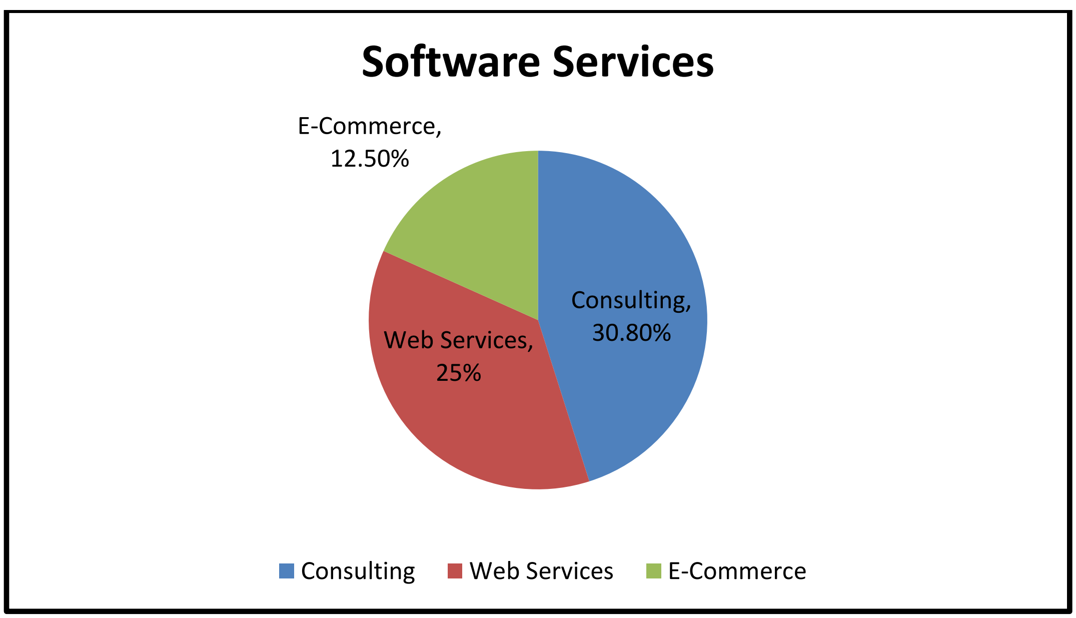


Figure 5. 2: Types of software Services

III. Domain Specific

Figure 5. 3 represent the types of software based on specific domain.

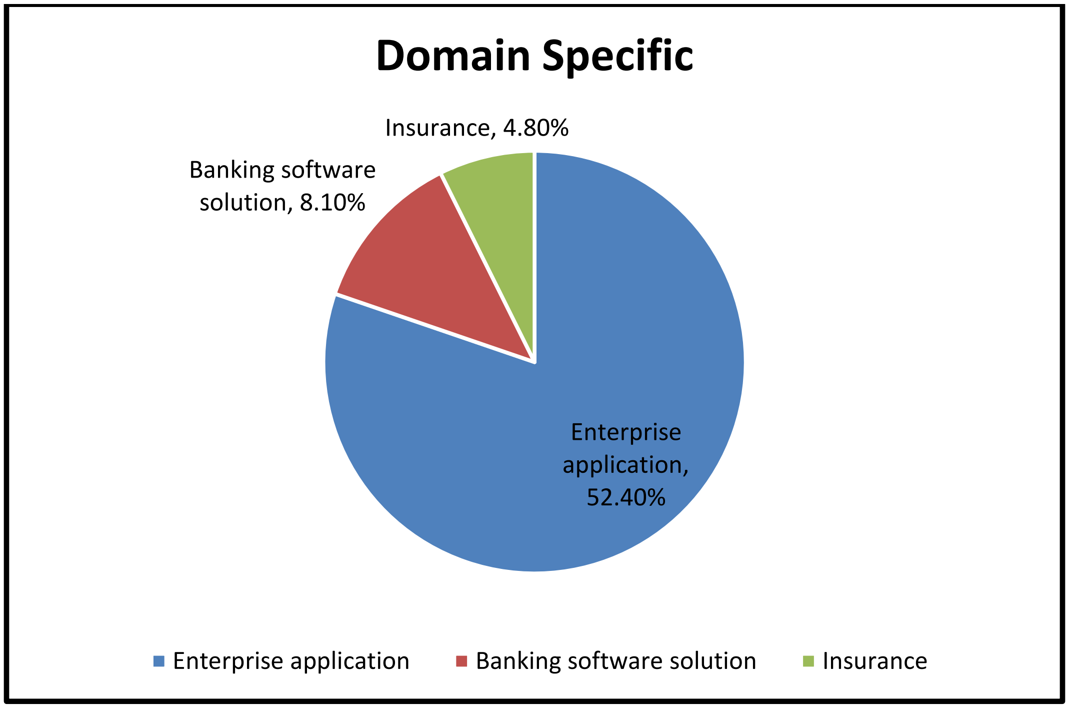


Figure 5. 3: Types of software based on domain

IV. IT Support Services

Figure 5.4 depicts the types of IT Support Services

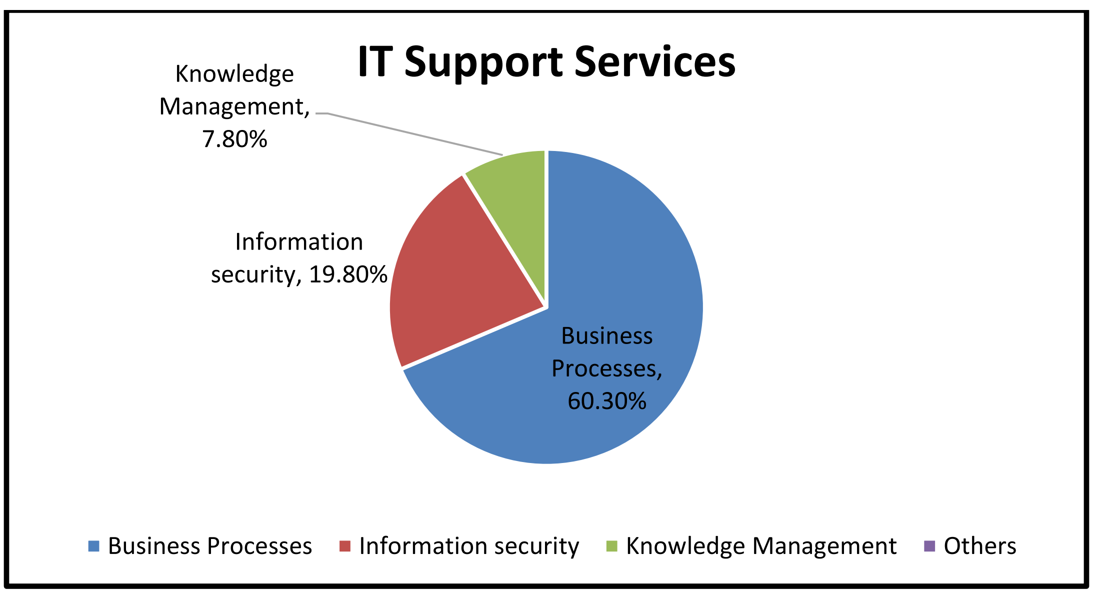


Figure 5.4 Types of IT Support Services

V. ***Deliverable of SD project*** - Figure 5.5 represents the deliverables of SD projects. Highest deliverable (78.8%) was those which had complete stand- alone software.

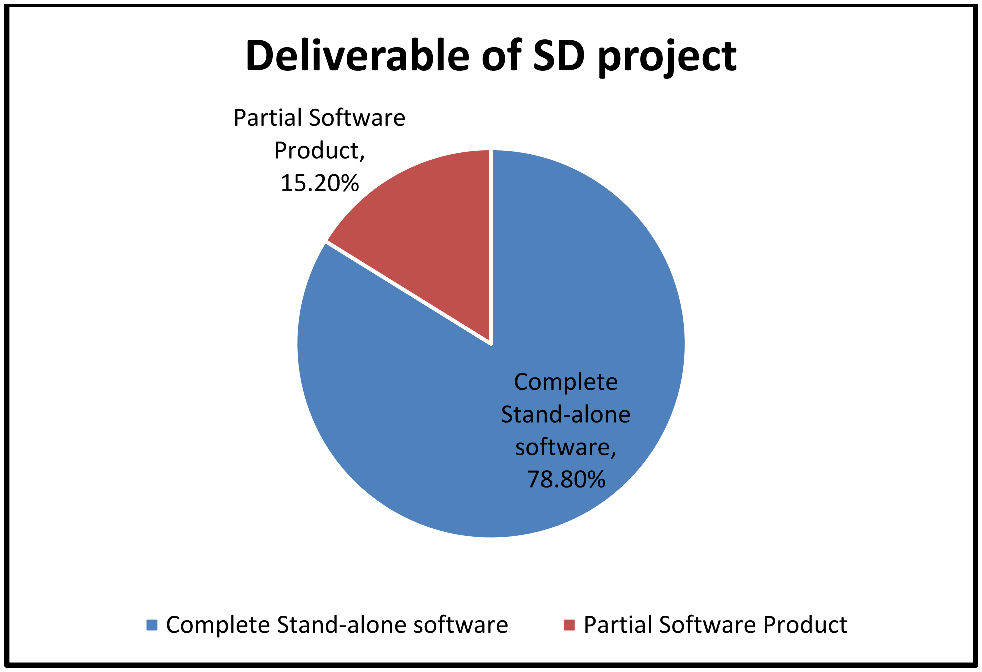


Figure 5. 5: Types of deliverables in projects

VI. ***Position of the Respondents in organization*** - Figure 5.6 represents the position of employees in SD organizations. Highest responses were received from developers, then team leads, team members, project manager and senior management.

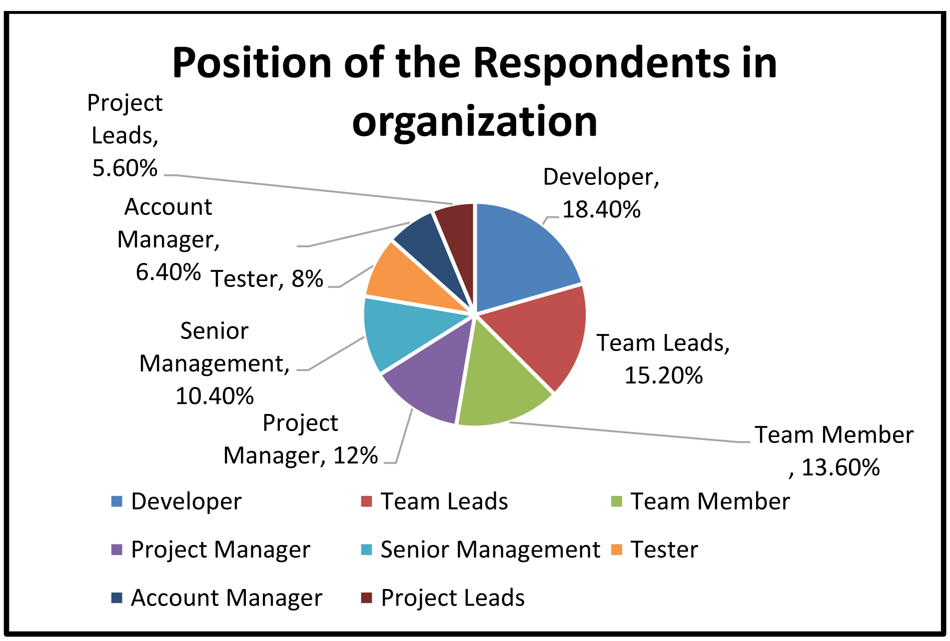


Figure 5. 6: Positions of employees

VII. ***People Employed*** -. Figure 5.7 represents the percentages number of employees.

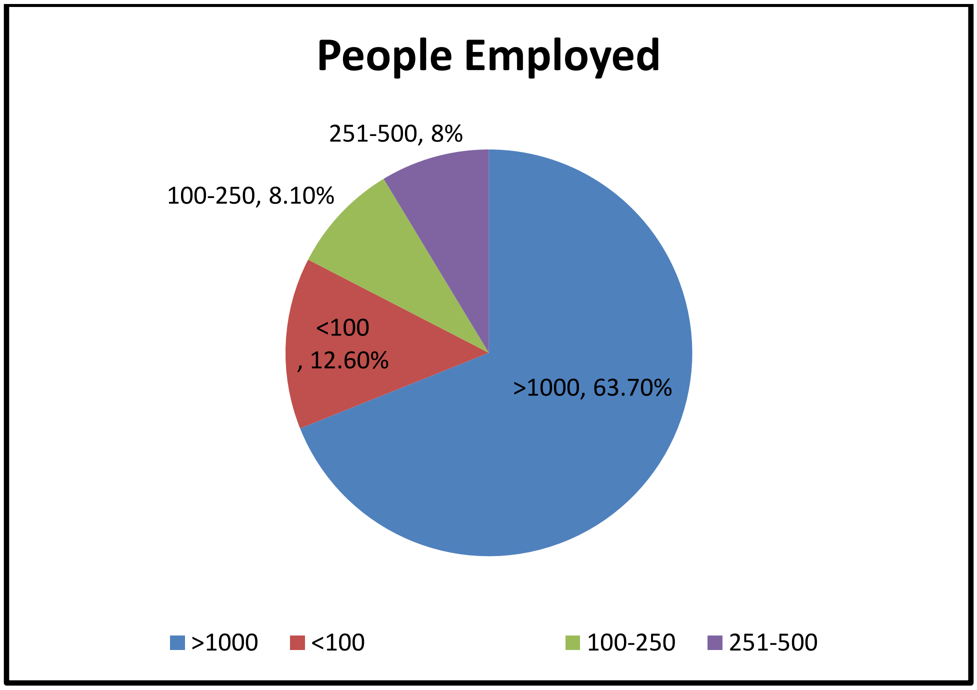


Figure 5.7: Number of employees

VIII. ***Annual Turnover*** **-** Figure 5.8 represents the annual turnover of SD organizations.

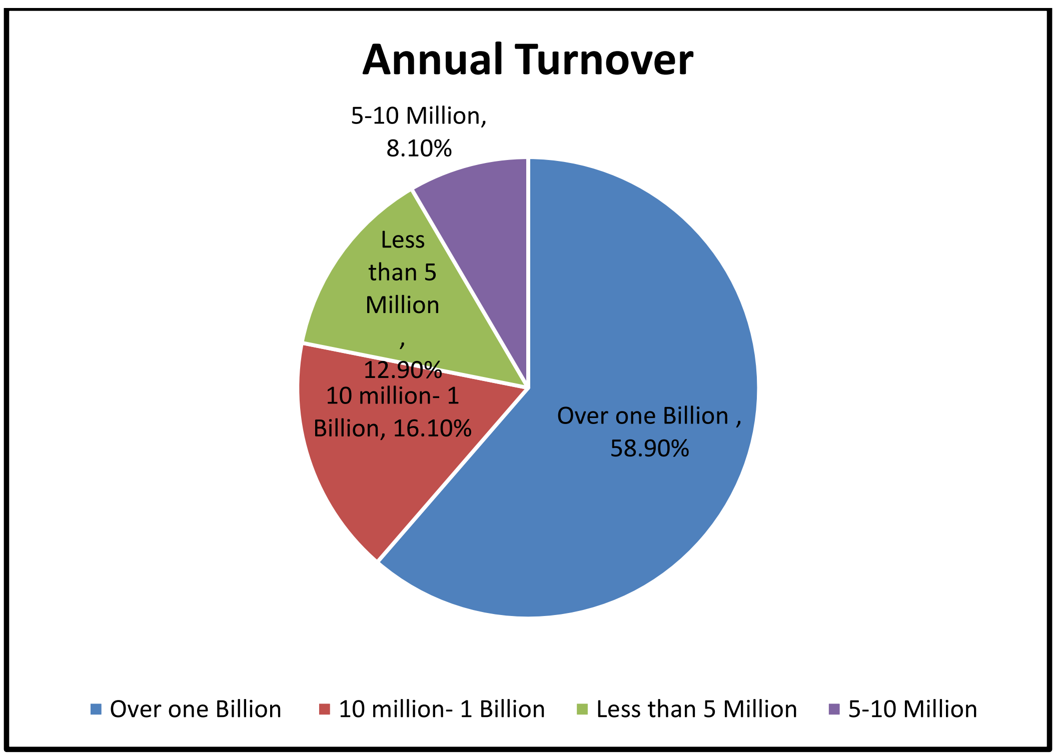


Figure 5. 8: Annual turnover

5.5 Responses of adoption of lean principles in SD organizations

The responses of SD organizations for degree of lean adoption is collected by the questions enquired in part B of survey questionnaire. In current study the lean principles and their effect are examined by the view and perceptions on implementation.

5.5.1 Response Analysis of enablers of value principles (VL)

Based on the analysis of data relating to part B.1: (VL1 to VL5), response rate of enablers of value principles /result is summarized in Figure 5.9.

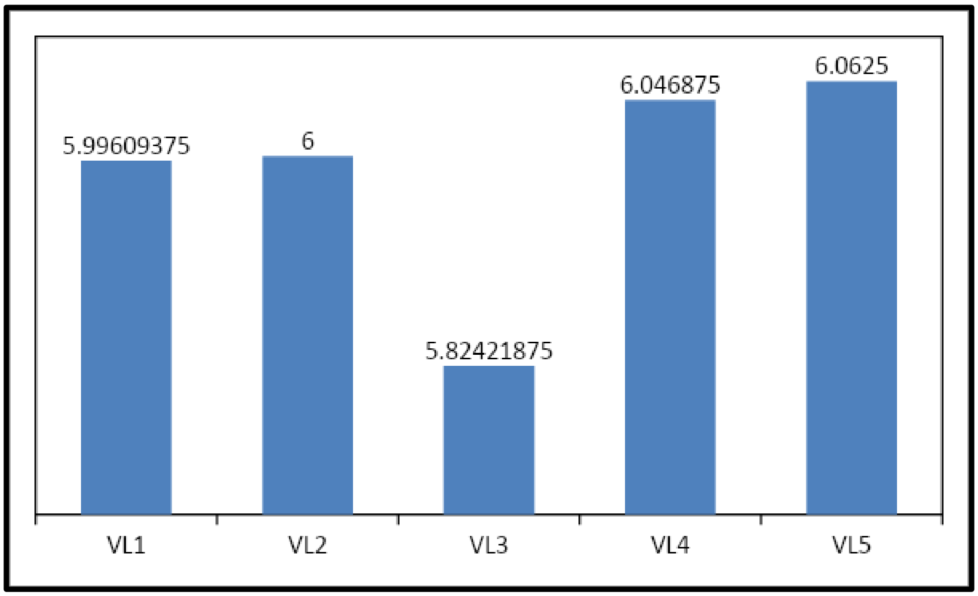


Figure 5.9: Responses on adoption level of Value

5.5. 2. Response analysis of enablers of Value Stream (VS)

Based on the analysis of data relating to part B.2: (VS1 to VS4), response rate of enablers of value stream principles /result is summarized in Figure 5.10.

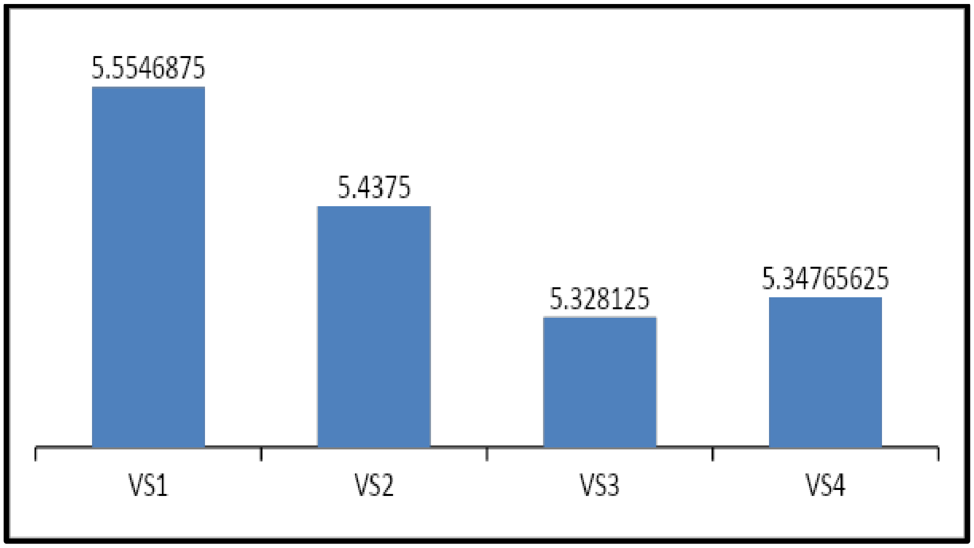


Figure 5.10: Responses on adoption level of Value Stream

5.5 3. Response analysis of enablers of flow (FL) principle

Based on the analysis of data relating to part B.3: (FL1 to FL6), response rate of enablers of flow principles /result is summarized in Figure 5.11.

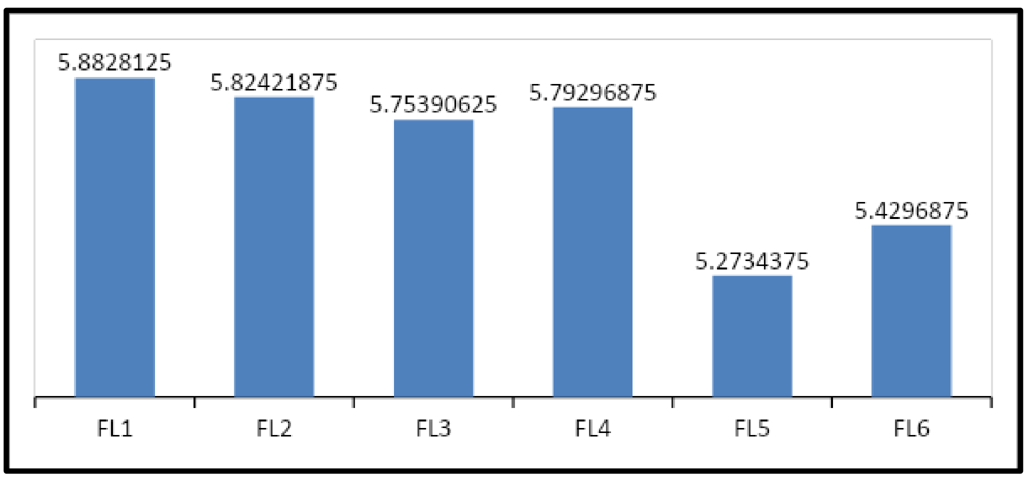


Figure 5.11: Responses on adoption level of Flow principles

5.5.4 Response analysis of enablers of Pull (PL) principle

Based on the analysis of data relating to part B.4: (PL1 to PL4), response rate/result of enablers of pull principles is summarized in Figure 5.12.

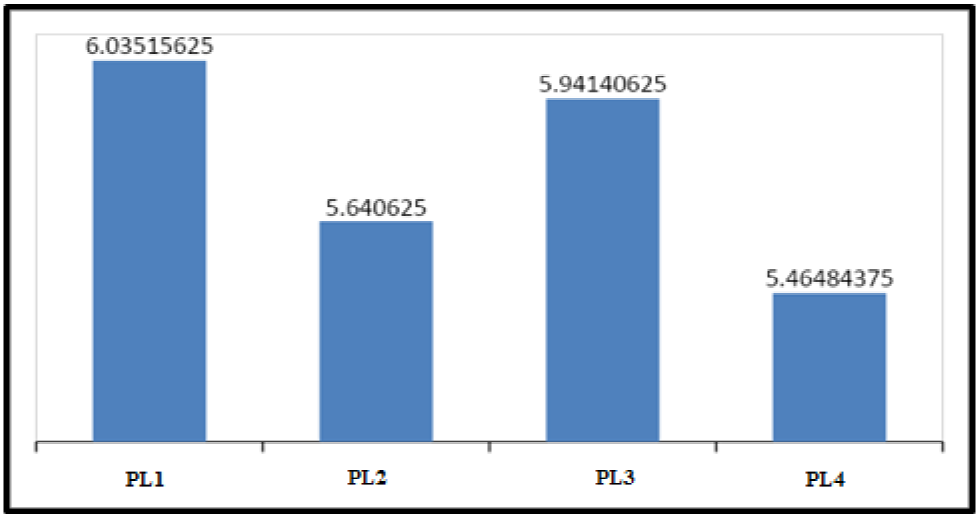


Figure 5.12: Responses on adoption level of Pull

5.5.5 Response Analysis of enablers of Perfection (PF) principle

Based on the analysis of data relating to part B.5: (PF1 to PF4), response rate/result of enablers of perfection principles is summarized in Figure 5.13.

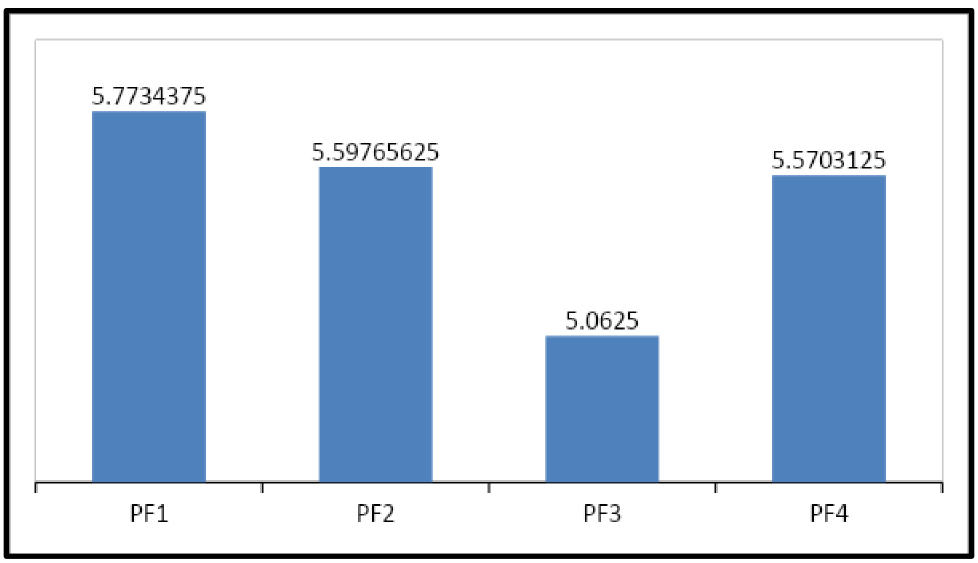


Figure 5.13: Responses on Value Stream

5.5.6 Response Analysis of enablers of Responsiveness (RS)

Based on the analysis of data relating to part B.3: (RS1 to RS4), response rate/result of enablers of responsiveness principles is summarized in Figure 5.14.

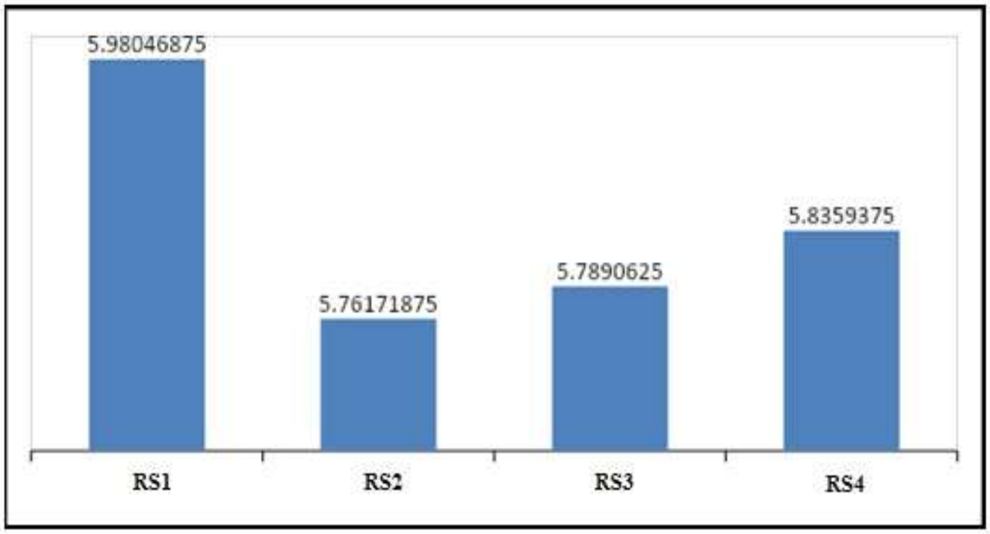


Figure 5.14: Responses on Responsiveness

5.6 Response Analysis of enablers of Operational performance (OP)

Based on the analysis of data relating to part E: (OP1 to OP4), response rate/result of enablers of operational performance is summarized in Figure 5.11. The trends of responses in descending order as OP3 (estimated cost), OP4 (agreed upon time),OP2 (quality),and OP1(WIP).

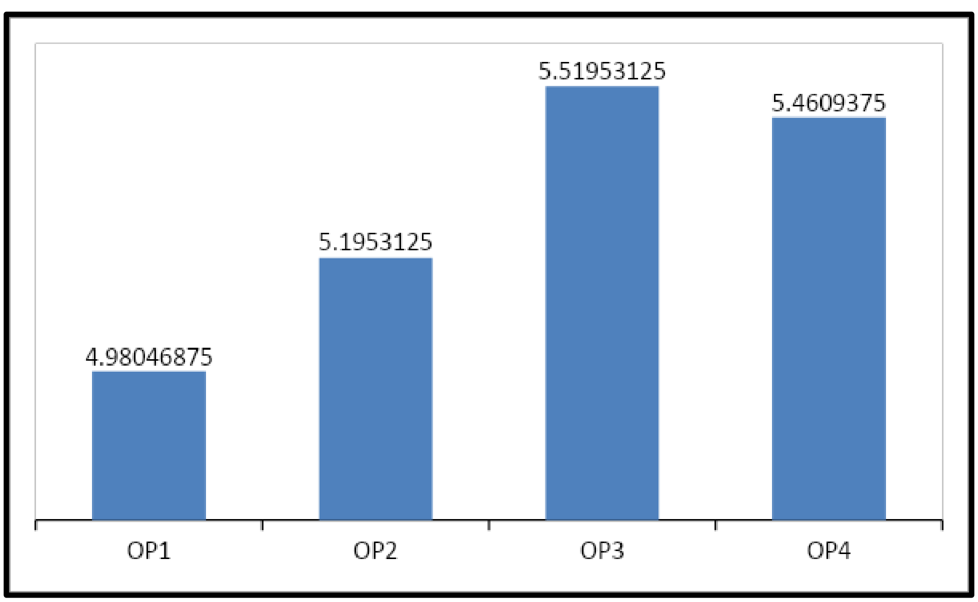


Figure 5.15: Responses on Operational performance

5.7 Drivers of Lean adoption

The purpose of this research was to get understandings of important issues like drivers of lean adoption. Major Drivers/Success Factor were identified for SD Projects-

A few drivers are identified in adoption of lean in SD projects. These are considered as employee involvement, top management support, performance management, mistake proofing, communication, and training and education for customer satisfaction, to reduce cost and fast delivery.

To overcome challenges in lean adoption, drivers are identified. These are called as success factors or drivers or enablers of lean adoption in SD projects. Drivers identified in lean adoption in SD projects as follows-

Cronbach's alpha test was employed for reliability of the drivers. It was found that the value of Cronbach is quite high ($\alpha = 0.813$) to confirm the internal consistency and reliability of construct of driver. Table 5.4 summarizes the types of drivers along with mean and standard deviation.

Table 5.4: Types of drivers with mean and standard deviation

Driver No.	Name of Drivers	N	Mean	Std. Deviation
D1	To eliminate waste (rework/delays/ WIP inventory)	256	5.61	1.312
D2	To minimize project cost	256	5.77	1.168
D3	To increase customer satisfaction	256	6.02	.986
D4	To build quality into the product	256	5.80	1.176
D5	Rapid delivery of software with coordination and communication of team	256	5.16	1.065
D6	To get solution of problem of customers	256	5.63	.949

Hypotheses for the drivers of lean adoption in SD Projects

Based on the literature, it is expected that there are significant drivers in adoption of lean in SD projects. The hypotheses for the drivers were proposed as given below.

H1a: Lean is implemented in SD projects in order to eliminate wastes (rework/delays)

H1b: Lean is implemented in SD projects in order to minimize project cost

H1c: Lean is implemented in SD projects in order to increase customer satisfaction

H1d: Lean is implemented in SD projects in order to build quality into the product

H1e: Lean is implemented in SD projects in order to get rapid delivery of software with coordination and communication of team

H1f: Lean is implemented in SD projects in order to get solution of customer's problem

Hypotheses *H1a, H1b, H1c, H1d, H1e, and H1f were tested using t-test.*

Table 5.5 represents the t- test for significant drivers at 0.05 significance level.

Table 5.5 : t- test for significant drivers in adoption of lean

One-Sample t- Test			
	Test Value = 4		
Drivers	**t- statistics**	**df**	**Sig. (2-tailed)**
D1	19.679	255	0.00**
D2	24.185	255	0.00**
D3	32.706	255	0.00**
D4	24.494	255	0.00**
D5	17.492	255	0.00**
D6	27.539	255	0.00**

It is evident that, the hypotheses *H1a, H1b, H1c, H1d, H1e, and H1f are accepted.*

Significant drivers to adopt lean in SD projects are to increase customer satisfaction by customer focus, to build quality into the product by mistake proofing, to minimize project cost by top management support and to get solution of problem by team involvement.

Table 5.6 represents the rank of drivers for implementing the lean in SD projects. The ranking was derived based on their mean score of responses.

Table 5.6: Ranking of drivers in lean implementation in SD projects

Code	Drivers	Mean	Rank	Std. Deviation
D3	To increase customer satisfaction	6.02	1	.986
D4	To build quality into the product	5.80	2	1.176
D2	To minimize project cost	5.77	3	1.168
D6	To get solution of problem by	5.63	4	.949
D1	To eliminate waste (rework/delays/ WIP inventory)	5.61	5	1.312
D5	Rapid delivery of software	5.16	6	1.065

In this empirical research customer focus problem and process oriented problem are solved by adopting lean in various SD projects to increase customer satisfaction (D3, D4 and D2). Customer focus, waste reduction, work stream efficiency and continuous improvement were the five core principles of lean in product development (Ebert *et al.*, 2012). Other drivers like delivery, waste reduction and minimization of project cost were also considered as other drivers in adoption of lean in SD projects.

The most important drivers in SD projects are already discussed. These are to increase customer satisfaction by focusing on customer, build quality into the product by mistake proofing, minimize project cost with top management support, to get solution of problem by employee involvement.

5.8 Barriers in adoption of lean in SD projects

The important barriers in adoption of lean in SD projects were investigated after intensive literature review. Fifteen barriers are summarized in Table 5.7. The responses were based on seven point Likert scale. The value of Chronbach Alpha (0.952) ensures high reliability of scale.

Table 5.7: Barriers in adoption of lean with reliability analysis in SD projects

Barrier/ Challenge code	Items	Cronbach Alpha
CH1	Uneven requirement flow and change requests.	0.**952**
CH2	Lack of techniques to recognize waste in SD Projects.	
CH3	Lack of information sharing or communication between organization and customers	
CH4	Lack of lean implementation experts with good leadership and coaching skills	
CH5	Lack of top management commitment	
CH6	Lack of communication and coordination within the team and among cross functional team	
CH7	Difficult to create a cross functional focus.	
CH8	Lack of coordination for parallel and sequential tasks.	
CH9	Lack of prioritization of bug fixing and adding new features	
CH10	Company policy and structure creates obstruction	
CH11	Lack of consultants and trainers in the company and outside	
CH12	Lack of proper requirement handover /We have unnecessary handovers	
CH13	Lack of transparency	
CH14	Functional silos and Employees' resistance inside the organization	
CH15	Lack of availability of required and allocation of resources	

The hypotheses for the barriers were proposed as given below.

Hypotheses for barriers

H2a: Uneven requirement flow inhibits lean implementation in SD Projects

H2b: Lack of information sharing inhibits lean implementation in SD Projects

H2c: Lack of communication inhibits lean implementation inhibits a barrier in lean implementation in SD Projects

H2d: Lack of coordination for parallel and sequential task inhibits lean implementation

H2e: Lack of availability of resource inhibits lean implementation in SD Projects

H2f: Lack of transparency inhibits lean implementation in SD Projects

H2g: Lack of top management commitment inhibits lean implementation in SD Projects

H2h: Lack of cross functional focus inhibits lean implementation in SD Projects

H2i: Lack of techniques to recognize waste SD Projects inhibits lean implementation in SD

H2j: Lack of prioritization of bug fixing and adding new features inhibits lean implementation in SD Projects

H2k: Lack of lean implementation experts with good leadership and coaching skills inhibits lean implementation in SD Projects

H2l: Lack of proper requirement handover inhibits lean implementation in SD Projects

H2m: Functional silos and Employees' resistance inhibits lean implementation in SD Projects

H2n: Company policy and structure inhibits lean implementation in SD Projects

H2o: Lack of consultants and trainers the company and outside inhibits lean implementation in SD Projects

Statistical analysis is carried out to test hypotheses of barriers. Hypotheses from H2a to H2o (15) were tested by t –test. The median score was taken as 4 and compared with observed mean of each item. Table5.8 summarized the significant barriers (at 0.05 significance level).

Table 5.8: t- Test for significant barriers in SD Projects

One-Sample t-Test			
Test Value = 4			
Barrier/Challenge	**t statistics**	**P value at α level of 0.05**	**Mean Difference**
CH1	18.468	0.00**	1.49
CH2	9.908	0.00**	1.012
CH3	11.643	0.00**	1.18
CH4	10.698	0.00**	1.091
CH5	9.088	0.00**	1
CH6	10.931	0.00**	1.184
CH7	11.689	0.00**	1.142
CH8	11.541	0.00**	1.145
CH9	11.017	0.00**	1.129
CH10	6.007	0.00**	0.708
CH11	4.883	0.00**	0.573
CH12	8.863	0.00**	0.957
CH13	11.034	0.00**	1.122
CH14	6.613	0.00**	0.755
CH15	11.88	0.00**	1.161

As reported in survey observation, all the barriers are significant; hence hypotheses from H2a to H2o are accepted.

Table 5.9 illustrates ranks and their mean scores of barriers while implementing lean in SD projects. The rank is decided by mean scores. It is evident from Table 5.8 that most important challenges/barriers lean implementation in SD projects are uneven requirement and change requests, lack of communication and co-ordination , and lack of information sharing.

Table 5.9: Ranking of barriers in adoption of lean in SD projects

Name of Barrier/Challenge	Barriers Number	N	Mean	Rank
Uneven requirement flow and change requests.	CH1	255	5.49	1
Lack of information sharing or communication between organization and customers	CH3	255	5.18	2
Lack of communication and coordination within the team and among cross functional team	CH6	255	5.18	3
Lack of availability of required and allocation of resources	CH15	255	5.16	4
Lack of coordination for parallel and sequential tasks	CH8	255	5.15	5
Difficult to create a cross functional focus	CH7	253	5.14	6
Lack of prioritization of bug fixing and adding new features	CH9	255	5.13	7
Lack of transparency	CH13	255	5.12	8
Lack of lean implementation experts with good leadership and coaching skills	CH4	253	5.09	9
Lack of techniques to recognize waste in SD Projects.	CH2	253	5.01	10
Lack of top management commitment	CH5	253	5	11
Lack of proper requirement handover (unnecessary handovers)	CH12	253	4.96	12
Functional silos and Employees' resistance inside the organization	CH14	253	4.75	13
Company policy and structure create obstruction	CH10	253	5.49	14
Lack of consultants and trainers in the company and outside	CH11	255	5.18	15

5.9 Overview of adoption Level of Lean Principles

Adoption level was investigated based on seven-point Likert-type scale in a survey study, Where 1 – Low adoption; 4 – partial adoption; 7 – complete adoption. An independent analysis was carried out for each of the five principles and responsiveness and observed some statistically significant results.

Table5.10 summarizes the reliability analysis of lean principles

Table 5.10: Reliability analysis of lean principles

Lean Principle	Variable	Source	Cronbach' s Alpha
Value	We use standardized tasks/parts clearly defining substance, order, timing, and desired result and processes for continuous improvement.	Spear and Bowen, (1999); Liker, (2004)	0.849
	Whenever defects are identified, it is logged and immediate actions are taken for fixing the problem and to obtain the desired quality on the first try/pass.	Staats *et al.* (2011),	
	We update, prioritize the backlog and incorporate new features throughout the entire programme of SD		
	We frequently remain in close contact with our customer to know about their exact requirements.	Olivella *et al.*(2008), Malmbrandt and Ahlstrom, (2013)	
	Our customers give feedback on quality and delivery performance of the software.		
Value Stream	We determine those activities that do not contribute value to the product or service and redesign work process to eliminate these tasks/processes	Staats *et al.* (2011), Malmbrandt and Ahlstrom, (2013)	0.847
	We frame current and future development process diagram capturing processing time, delay time, number of people involved		
	We identify all the sources of waste in SD such as delays, unnecessary rework due to errors, defects, extra features and partial work done and track them		
	We analyse and design the workflow required to bring a software to a customer		

Flow	We have developed subsystems to handle dependencies of the tasks/modules and their ordering	Staats *et al.* (2011) Meiling, (2010); Malmbrandt and Ahlstrom, (2013)	0.858
	We consider conceptual solutions in parallel and narrow to develop a single focused idea for achieving the final specifications		
	We have developed direct pathway and connections among people and tasks in order to have smooth and timely flow of information		
	Workload is leveled and balanced according to available resource and features to meet sprint demand (scrum /stand up for assigning task)		
	Team members collaborate with each other for achieving goal		
	Process parameters are displayed on workplace on visual dashboard/ electronic dashboards to improve the process and progress of work		
Pull	Development of software workplace is pulled by the demand of the next stakeholder in the organization	Andersson (2003) Ikonen *et al.*(2010)	0.803
	We use signal cards which contains specification of job, detailed designs, verifications and implementation of the story and help to know what to do next task within stipulated time		
	Work is assigned according to available to available capacity in my area of operations		
	Team members pull the items/ tasks from the backlog according to priority		
Perfection	Our organization adopt continuous improvement and continuous learning techniques	Rodrigues *et al.* (2013)	0.840
	Employees are actively involved in problem solving		
	Information for the execution of my work is displayed visually at my work and visible for every team member		
	We capture online/documented feedback from customers		
Responsiveness	We are highly capable of responding to customized requirements to handle dynamicity	Popendieck and Poppendieck (2007)	0.825
	We use practices such as iterative, standardized error code, continuous integration, test driven development, and daily standup meetings for expediting the SD project		
	We use scrum in SD project to track progress		
	We use periodic builds and code reviews		

Following hypotheses were proposed for lean adoption in SD projects.

Hypotheses for Lean Adoption

H3a: Value principle is significantly adopted in SD projects/Value principle has high adoption level in SD projects

H3b: Value stream is significantly adopted in SD projects

H3c: Flow principle is significantly adopted in SD projects

H3d: Pull principle is significantly adopted in SD projects

H3e: Perfection is significantly adopted in SD projects

H3f: Responsiveness is significantly adopted in SD projects

One sample t- test is performed to find evidence of a significant difference between the population mean and a hypothesized value (1-sample t test). It is used to determine whether the difference is statistically significant, the t-test calculates. The *p*-value is used in the context of null hypothesis testing in order to quantify the idea of statistical significance of evidence. Table 5.11 depicts the t test for lean principles.

Table 5.11: t test for lean principles

Lean Principle	**t- statistcs**	**Df**	**p value Sig.(2-tailed)**
Value (VL)	38.33993	255	0.00
Value Stream (VS)	20.68641	255	0.00
Flow (FL)	28.19095	255	0.00
Pull (PL)	30.04398	255	0.00
Perfection (PF)	20.92353	255	0.00
Responsiveness (RS)	28.37215	255	0.00
Significance level at 0.05			

It is evident that Hypotheses H3a, H3b, H3c, H3d, H3e, H3f are accepted

It results that lean principles are adopted in SD projects. It also concludes that SD organizations are adopting lean principles in the order of value, responsiveness, pull, flow, perfection and value stream. The ranking is based on mean value. Table 5.12 summarizes the ranking of adoption level of lean principles. These are described below-

I. ***Value principle*** - Value may be created by lean tools such as VSM, Mistake Proofing, KIVP, team based problem solving, continuous improvement, standardized work and visual control. Value principle is presented by VL. One sample t-test is applied. Differences are statistically significant among the companies and t- Statistics (38.33) is significant at p=0.05, mean M=5.9859.

II. ***Value stream*** – It detects non- value activities with help of current and future development diagram. It is represented by VS. One sample t- test is used to confirm statistically significance of VS. Its mean value is M=5.417

III. ***Flow principle***-Petersen and Wohlin (2009) revealed that improvement of flow is done by shorter lead time means timely delivery of product to the customer. Flow principle covers DSM, JIT information, set base concurrent development, Heinjunka and visual management. It is represented by FL and mean value is M=5.6597.

IV. ***Pull principle*** -Pull principle includes the Kanban, pulling is executed according to capacity and stockholder’s demand. It is represented by PL. Its mean value is 5.7705

V. ***Perfection***- It covers continuous improvement, continuous learning, team involvement in problem solving, and feedback from customer It is represented by PF and its mean value is M=5.501

VI. ***Responsiveness***-Responsiveness has more focus on the just in time aspects of lean. Petersen and Wohlin, 2009 supported that agile software development aims at being highly focused and responsive to the needs of customer. Responsiveness is represented by RS and its mean value M=5.8418

Table 5.12: Ranking of lean principles adopted in SD projects

Lean Principles	Std. Deviation	Mean	Rank
Value (VL)	0.82877	5.9859	1
Responsiveness (RS)	1.03865	5.8418	2
Pull (PL)	0.94289	5.7705	3
Flow (FL)	0.94197	5.6597	4
Perfection (PF)	1.14778	5.501	5
Value Stream (VS)	1.09598	5.417	6

5.10 Summary

In this chapter, drivers and barriers were identified in lean adoption in SD projects after survey analysis. According to the result of current study the important drivers are to increase customer satisfaction by customer focus, to build quality into the product by mistake proofing, to minimize project cost by top management support and to get solution of problem by team involvement.

These important barriers are further categorized as customer-related, process-related, and people-related barriers. The prominent barriers observed in survey are-customer related barriers viz. uneven requirement flow and change requests, lack of information sharing or communication between organization and customers. Likewise, people- related barriers viz. lack of communication and coordination within the team and among cross functional team, lack of availability of required and allocation of resources, difficult to create a cross functional focus, lack of lean implementation experts with good leadership and coaching skills were significant.

Further, process related barriers viz. lack of coordination for parallel and sequential tasks, lack of prioritization of bug fixing and adding new features, lack of transparency, and lack of techniques to recognize waste in SD Projects were found significant.

The major barriers are as follows-uneven requirement flow and change requests, lack of information sharing/communication or lack of transparency, cross functional focus/lack of coordination among team, lack of availability and allocation of resources, and lack of lean experts to coach the employees.

This research investigated that uneven requirement flow and change requests create challenge in adoption of lean principles in SD projects. Nord *et al.* (2012) suggested that by making architecture visible, flow management in SD projects are improved.

A framework is proposed by Ebert *et al.* (2012) in lean adoption by prioritizing requirements. Change management system should be established to manage change requests. Software development is considered as product development (Poppendieck and Poppendieck, 2007) in which output is information. Reinertsen, (2005); Czabke *et al.*, (2008) revealed that communication is a key element to share all necessary and valuable information in the process of lean transformation. The smooth flow of information is required to gain knowledge of SD process and improvement in strategic and operational activities of SD projects. Ebert *et al.*2012 examined that the intangible nature of software, developers as knowledge worker found difficulty in defining flow in software development.

According the result of present study, it was found that all core lean principles along with responsiveness are being adopted in SD organizations. Value, pull and flow are regularly adopted but perfection and value stream are adopted by low percentages of SD organizations. Responsiveness which represents the lean and agile characteristics, also frequently used in more SD companies.

Next chapter discusses the relationship among lean principles, lean thinking and operational performance.

Chapter 6

Impact of Lean on Operational Performance

Chapter 6

Impact of Lean on Operational Performance

6.1 Introduction

In this chapter, data was analyzed to test the hypotheses developed in chapter 3 for establishing relationship in adoption of lean principles and lean thinking and also its effect on operational performance of the SD organizations. A model was developed for this and tested using Structural Equation Modelling (SEM). The steps involved in SEM analysis are as below:

1. *Development of conceptual model*- it comprises measurement model and Structural model
2. *Elements for Model evaluation*—it includes estimation method, selection of model fit indices, and model identification
3. *Measurement of Model specification and evaluation*-Single and multifactor approach, measurement of reliability and validity of model
4. *Modification of Structural model and rectification*- Estimating the coefficient and hypotheses testing

6.2 Development of Model constructs

Based on literature review on lean, a conceptual model for adoption of lean principles in SD projects has been developed. The model is given in Figure 6.1. The components of proposed model are grouped into 6 main constructs: Value, Value stream, flow, pull system, perfection and Responsiveness. These six constructs are the bottom line lean principles. The suggested lean constructs can be used at the organizational level for performance of software development projects.

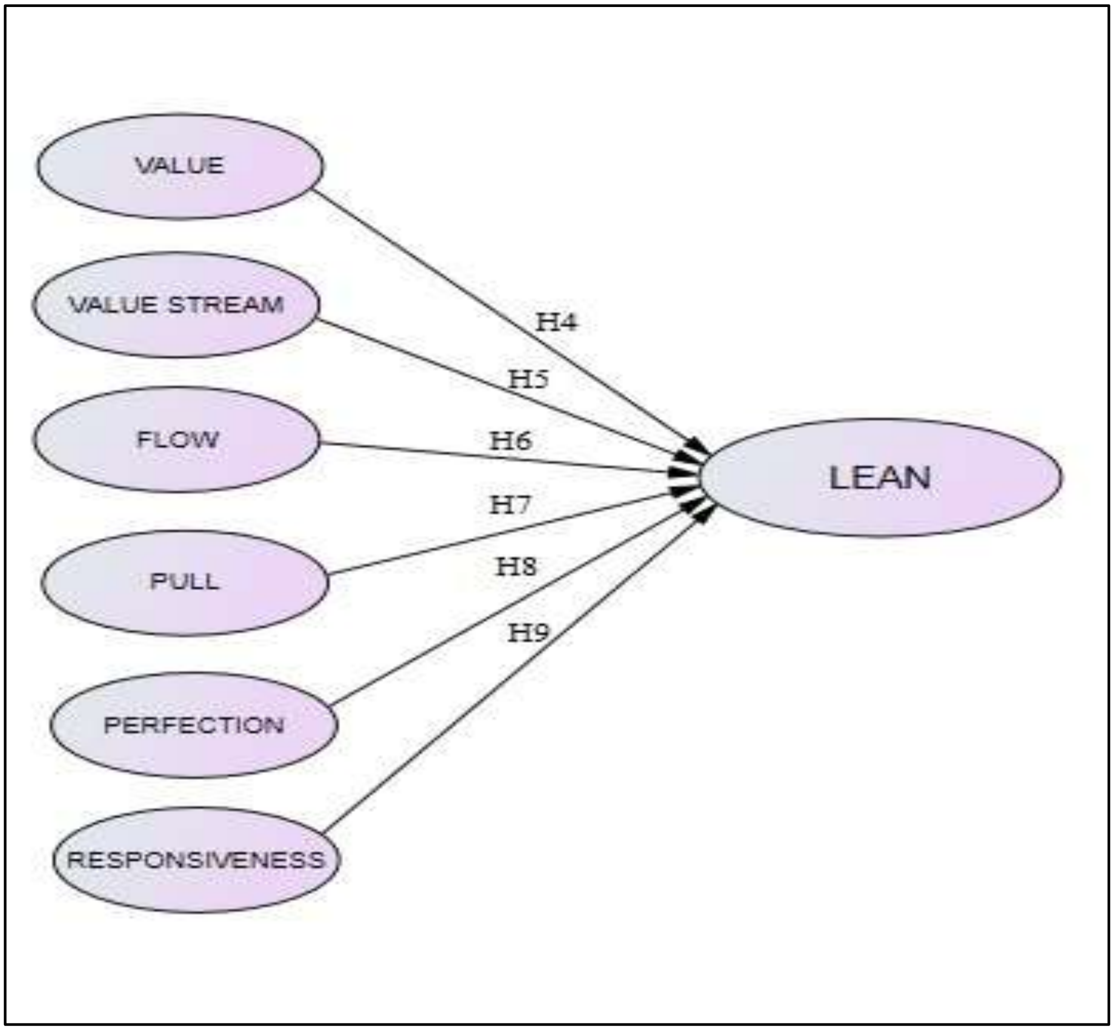


Figure 6.1: Conceptual Model

6.3 Choice of model estimation Method and Model Indices

Structural coefficients in SEM may be calculated with various methods such as maximum likelihood estimation (MLE), un-weighted least square (ULS), Scale free least square (SLQ) and asymptotically distribution free (ADF), generalized least square (GLS) methods of coefficients estimations.

Maximum likelihood estimation (MLE) is most frequently used method of estimation. Since sample size is not large, MLE method is suitable for current study. Kline (2005) revealed that MLE is the iterative method that estimates the fairly accurate value for each parameter. Likewise, Ullman (2003) reported that MLE establishes estimates supported on maximizing the probability(likelihood) that the observed covariance are drawn from a population presumed to be the similar as that reflected in the coefficient estimates. IBM AMOS 26.0 Software was used for MLE (Maximum likelihood estimation).

6.4 Model Fit Indices

The fit indices for evaluation of the model are described below-

χ^2 /df - Normed Chi Square- Chi square ($\chi^{2)}$ test linked with p value is employed to compare the observed and estimated covariance matrices. At least four test like chi square, goodness of fit index(GFI),normed fit index (NFI) or comparative fit index (CFI) and root mean square residual (RMR) are proposed by Kline (2005) to deal with spreaded statistical predictions. Further, other frequently used indices like root mean square error of approximation (RMSEA) and chi square are also suggested to study. The acceptance limit of model fit indices is depicted in Table 6.1.

Table 6.1: Acceptance Limit of Model fit Indices

Model Fit Indices	**χ^2 /df**	**GFI**	**AGFI**	**RMSEA**	**CFI**	**RMSR**
Acceptable Limit	≤3.00	≥0.90	≥0.8	≤0.10	≥0.90	≤0.14

6.5 Factor Analysis

In the present research, factor analysis is used for reduction of dimensions and identification of research constructs concerned with lean principles and performance measures of SD projects. There exist significant correlations among various variables. Originally 31 items were used to gain insight to respondent’s perception. Principal Component Analysis was used as extraction method and rotation method is used as Varimax with Kaiser Normalization. Six factors (lean principle) were extracted with eigen value more than one explaining with 66.39% of variance.

Confirmatory factor Analysis is used to describe to the measurement model testing. This method is employed to check the viability of selected model and structures, which are based on theory or as research objectives and also justify whether presented data are reliable with a proposed research model having restrained configuration. The measurement models are of two types such as one factor congeneric and other multi-factor models. One factor congeneric model is utilized to assess item reliability, construct reliability whereas multifactor measurement models are liable to investigate the discriminant validity of the individual scales in the construct.

6.5.1 KMO and Bartlett's Test

The Kaiser-Meyer-Olkin (KMO) measure of sampling adequacy is a statistic that indicates the proportion of variance in variables that might be caused by underlying factors. High values (close to 1.0) generally indicate that a factor analysis may be useful with the data. If KMO is ≤0.5, then results of factor analysis are not useful. Bartlett's tests sphericity in which small values (less than 0.05) of the significance level indicate that a factor analysis may be useful with our data. Table 6.2 depicts the result of KMO and Bartlett's test.

Table 6.2: KMO and Bartlett's Test

KMO and Bartlett's Test		
Kaiser-Meyer-Olkin Measure of Sampling Adequacy		.856
Bartlett's Test of Sphericity	Approx. Chi-Square	3.821E3
	Df	465
	Sig.	.000

The value of KMO is 0.856 and Bartlett's test (0.000) reflects that the data is appropriate for performing factor analysis.

6.5.2 Eigen Value

Eigen value for the given factor measures the variance in the all variables, which is accounted for by that factor (Costello, 2009). Eigen values are the sum of squared values of factor loadings related to factors. Table 6.3 depicts the Eigen values associated with each factor. It is evident that first seven factors have large amount of variance, whereas remaining factors explain small amount of variance. The components having Eigen values less than one can be dropped.

Table 6.3: Extraction sum of squared loadings

	Initial Eigen values			**Extraction sums of squared Loading**			**Rotation sums of squared Loading**		
	Total	% of Variance	Cumulative	Total	% of Variance	Cumulative %	Total	% of variance	Cumulative %
1	**7.750**	**25.001**	**25.001**	**7.750**	**25.001**	**25.001**	**3.645**	**11.758**	**11.758**
2	**2.904**	**9.367**	**34.368**	**2.904**	**9.367**	**34.368**	**3.193**	**10.299**	**22.057**
3	**2.445**	**7.887**	**42.255**	**2.445**	**7.887**	**42.255**	**2.830**	**9.130**	**31.188**
4	**2.151**	**6.940**	**49.195**	**2.151**	**6.940**	**49.195**	**2.817**	**9.088**	**40.276**
5	**1.998**	**6.444**	**55.639**	**1.998**	**6.444**	**55.639**	**2.767**	**8.927**	**49.203**
6	**1.744**	**5.625**	**61.265**	**1.744**	**5.625**	**61.265**	**2.661**	**8.583**	**57.786**
7	**1.579**	**5.094**	**66.359**	**1.579**	**5.094**	**66.359**	**2.658**	**8.573**	**66.359**
8	.801	2.582	68.941						
9	.762	2.457	71.398						
10	.741	2.391	73.788						
11	.666	2.149	75.937						
12	.645	2.082	78.019						
13	.586	1.892	79.911						
14	.537	1.731	81.642						

15	.505	1.629	83.270						
16	.486	1.567	84.838						
17	.465	1.501	86.339						
18	.424	1.367	87.706						
19	.423	1.364	89.069						
20	.382	1.233	90.302						
21	.363	1.172	91.474						
22	.342	1.103	92.577						
23	.334	1.077	93.654						
24	.306	.989	94.643						
25	.282	.910	95.553						
26	.270	.870	96.424						
27	.263	.847	97.271						
28	.231	.746	98.017						
29	.218	.703	98.720						
30	.205	.662	99.382						
31	.192	.618	100.000						
Extraction Method: Principle component Analysis (PCA) Method									

Scree Plot

Those factors which have Eigen value more than one are retained according to scree plot. Six factors were selected as lean adoption factors. This shows their contribution to total variance as depicted in Figure 6.2.

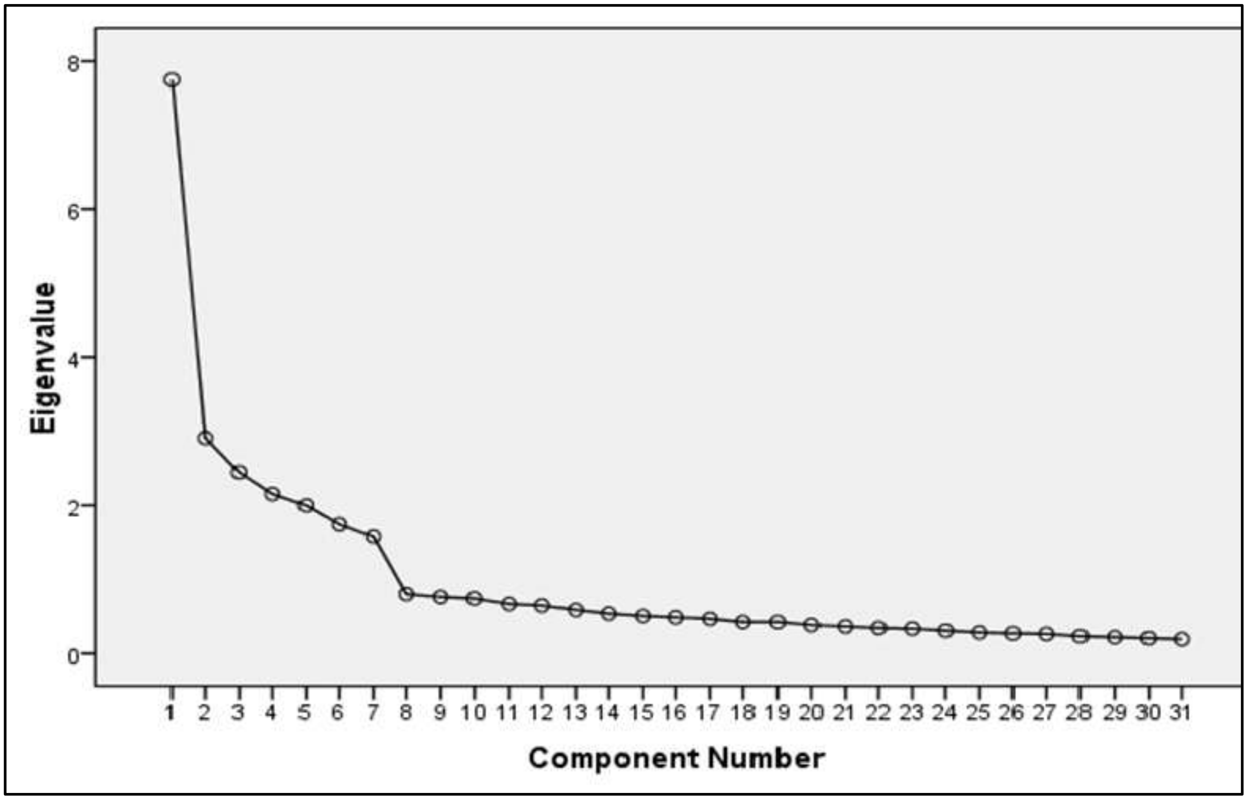


Figure 6.2: Scree plot

6.5.3 Factor Loading and Rotation

According to Hair *et al.* (2006) items with standardized loading should be 0 .5 or higher (ideally 0.7 or higher). In current research, item with factor loading greater than 0.6 factors were considered for further analysis. Rotation is a method used to simplify interpretation of a factor analysis. It changes the pattern of factor loading and hence can improve the interpretation. With the un-roated factors, it is difficult to interpret. Hence rotated component matrix can be helpful to simplify the structure with Varimax rotation approach to maximum extent. Table 6.4 lists the Varimax rotated component matrix, with the items that are high loading factor on each of the extracted factors.

Table 6.4: Varimax Factor Rotated Component Matrix

	VL	VS	FL	PL	PF	RS	OP
VL1	0.846						
VL2	0.737						
VL3	0.646						
VL4	0.76						
VL5	0.77						
VS1		0.771					
VS2		0.784					
VS3		0.778					
VS4		0.777					
FL1			0.779				
FL2			0.761				
FL3			0.657				
FL4			0.729				
FL5			0.678				
FL6			0.698				
PL1				0.802			
PL2				0.737			
PL3				0.829			
PL4				0.714			
PF1					.880		
PF2					.857		
PF3					.727		

PF4					.800		
RS1						0.749	
RS2						0.748	
RS3						0.785	
RS4						0.771	
OP1							.783
OP2							.814
OP3							.796
OP4							.798

6.5.4 Internal consistency analysis

Cronbach Alpha was used as reliability coefficient to measure internal consistency. The internal consistency is an estimate of reliability of test scores. Cronbach's alpha is widely believed to indirectly indicate the degree to which a set of items measures a single unidimensional latent construct. Table 6.5 depicts the reliability of constructs.

To investigate the significance of important lean principles in SD projects to relate with lean thinking, hypotheses were formulated as discussed in chapter 3.

In order to meet the third objectives of the work an empirical model was developed. The hypotheses for the empirical model are as below:

H4: Value principle enables lean thinking in SD Projects

H5: Value stream principle enables lean thinking in SD projects

H6: Flow principle enables lean thinking in SD Projects

H7: Pull principle enables lean thinking in SD Projects

H8: Perfection principle enables lean thinking in SD projects

H9: Responsiveness enables lean thinking in SD Projects

H10: Lean has positive effects on operational performance

Table 6.5: Reliability of constructs

S. No.	Constructs	Number of Items	Cronbach Alpha Index
1.	Value Principle (VL)	5	0.849
2.	Value Stream Principle (VS)	4	0.847
3.	Flow Principle (FL)	6	0.858
4.	Pull Principle (PS)	4	0.803
5.	Perfection Principle (PF)	4	0.840
6.	Responsiveness (RS)	4	0.825

6.6 Structural Equation Model

Structural equation modeling (SEM) encompasses such diverse statistical techniques as path analysis, confirmatory factor analysis, causal modeling with latent variables, and even analysis of variance and multiple linear regressions.

Hair, 2013 suggests that SEM delivers the measurement model, which specifies the rules of correspondence between measured and latent variables (constructs). The measurement model enables the researcher to use any number of variables for a single independent or dependent construct. Once the constructs are defined, then the model can be used to assess the extent of measurement error (known as reliability*)*.

SEM is a collection of statistical techniques that allow the examination of relationships among multiple predictor and response variables. These variables can be observable (directly measured, also referred to as manifest variables) or unobservable (Bagozzi andYi 1988).

Vinod and Joy (2012) revealed that SEM consists of two types of models:

a. *Measurement model*: the measurement model represents the theory that specifies how measured variables come together to represent the theory.

b. *Structural model:* the structural model represents the theory that shows how constructs are related to other constructs.

In SEM two types of variables are used, namely endogenous and exogenous variables. Endogenous variables are equivalent to dependent variables and exogenous variables are equal to the independent variables.

6.7 One Factor Congeneric Model

One factor congeneric model is defined as a model of single latent variable (Construct/ Factor) which is measured by various observed variables (item/indicators). Through survey, observed variables are directly observable.

6.7.1 One Factor measurement model for value principles (VL)

The latent variable of Value principle holds five indicators VL01 to VL05. The ratio of chi square to degree of freedom (χ^2 /df) is 1.283 suggesting the good fit to the data .The values of other indices of model fit are CFI= .998, NFI= .990, RMR= .017, GFI=.992, RMSEA= .033 and AGFI= .972 ($p \leq 0.001$) are well within standard limits. The measurement model of value principle is found statistically significant as shown in Figure 6.3, table 6.6 and table 6.8. These result shows that value based principles presented a good fit. Figure 6.3 depicts the CFA results for Value principles.

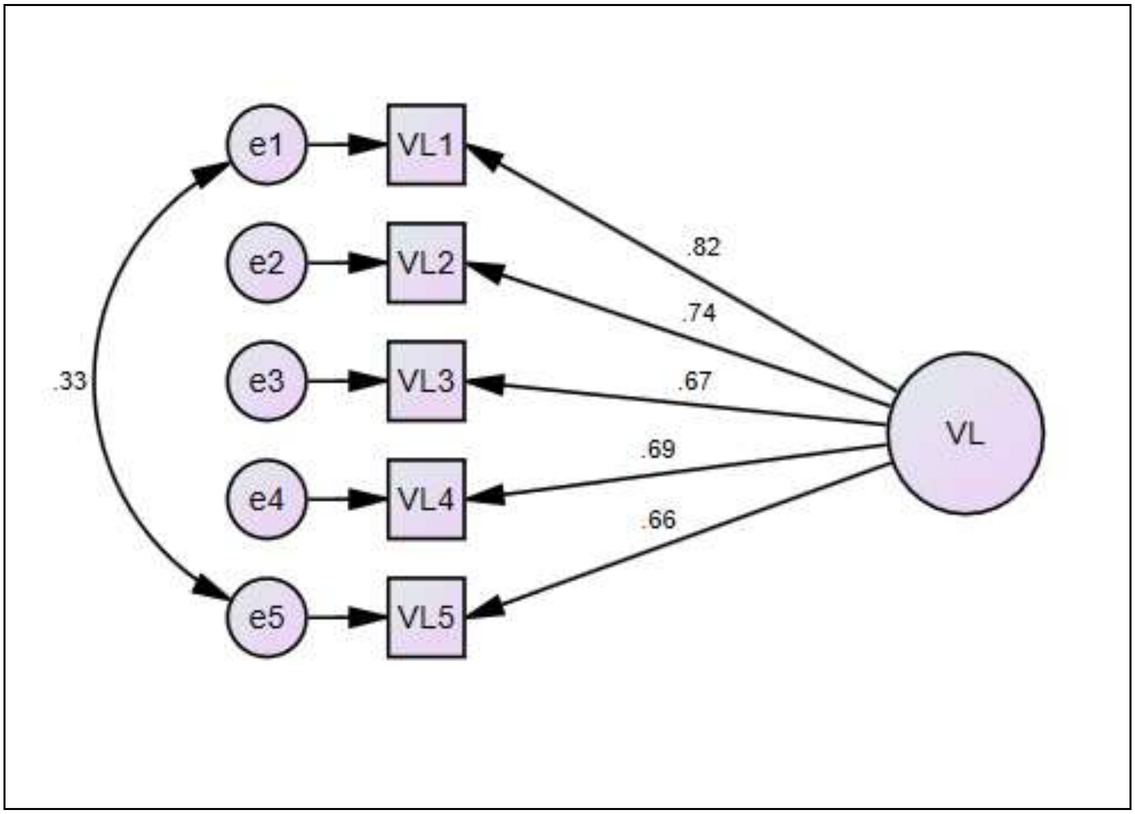


Figure 6.3: CFA results for Value principles

Table 6.6: Regression weights for Value principle

	Estimates	Estimate Std.	Standard Error	Critical Ratio	P
VL1 <--- VL	1.000	.825			
VL2 <--- VL	.967	.743	.085	11.347	***
VL3 <--- VL	.795	.673	.077	10.325	***
VL4 <--- VL	.817	.694	.077	10.655	***
VL5 <--- VL	.842	.662	.068	12.376	***

***p ≤0.001

Table 6.7 represents the acceptance limit of model fit indices.

Table 6.7: Acceptance limit of model fit Indices

Model Fit Indices (Hair *et al.*,2006)	**χ^2 /df**	**GFI**	**AGFI**	**RMSEA**	**CFI**	**RMSR**
Acceptable Limit	≤3.00	≥0.90	≥0.8	≤0.10	≥0.90	≤0.14

Where χ^2 /df - Normed chi square, RMR- Root mean square residual, GFI- Goodness of Fit Index, AGFI- Ajusted goodness of fit index, CFI- Comparative Fit Index, NFI- Normed Fit Index, RMSEA- Root mean square error of approximation.

Table 6.8: Model fit Indices for Value (VL) principle

S. No.	**Model Fit Indices**	**Measurements**
1.	χ^2 /df	1.283
2.	GFI	.992
3.	AGFI	.972
4.	NFI	.990
5.	RMSEA	.033
6.	CFI	.998
7.	RMR	.017

6.7.2 One factor measurement model for value stream principles

The latent variable of VS holds four items VS01 to VS04. The ratio of chi square to degree of freedom test (χ^2 /df) is .255 (p ≤0.001) which specifies the data are good fitted. The values of other model fit indices like CFI= 1.000 , NFI=.999, RMR=.010, GFI=.999, AGFI=.995, and RMSEA=.000 are well within standard limits. The measurement model of value stream is found statistically significant as shown in Figure 6.4, table 6.9 and table 6.10. These outcomes recommended that the measurement model of value stream principles found statistically good. Figure 6.4 depicts CFA results for Value Stream principles.

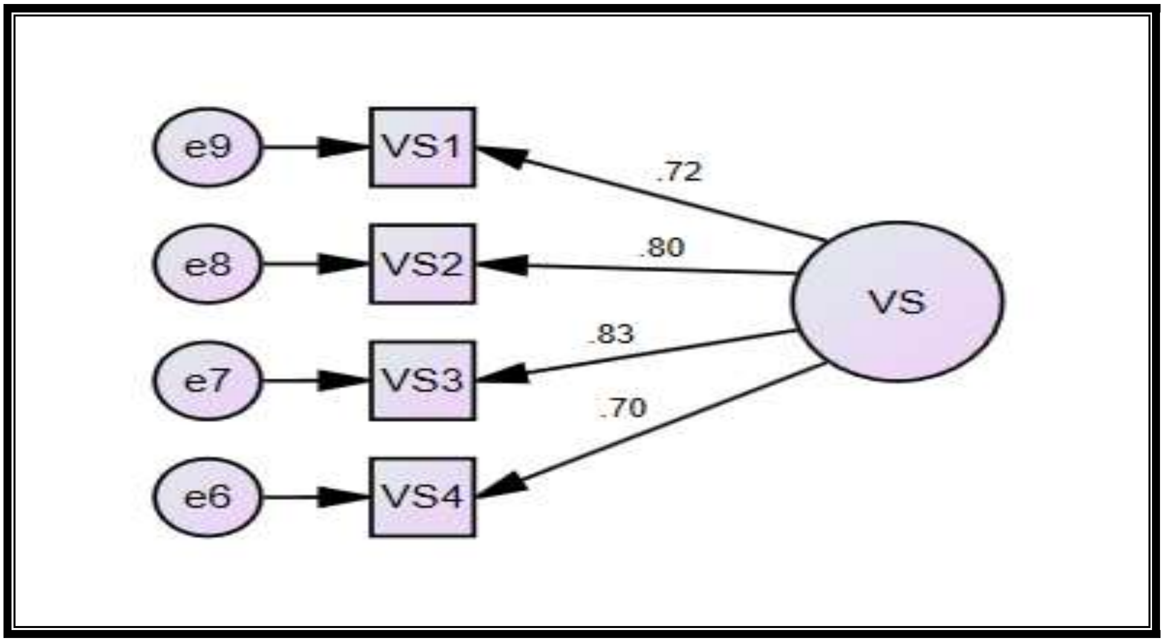


Figure 6.4: CFA results for Value Stream principles

Table 6.9: Regression weights for value stream principles

			Estimates (Unstandardized)	Estimates (Standardized)	Standard Error	Critical Ratio	P
VS1	<---	VS	0.748	.720	0.064	11.715	***
VS2	<---	VS	0.876	.802	0.067	13.056	***
VS3	<---	VS	1.000	.831			
VS4	<---	VS	0.749	.697	0.067	10.115	***

***p ≤0.001

Table 6.10: Model Fit Indices for value stream (VS) principle

S. No.	Model Fit Indices	Measurements
1.	χ^2 /df	.255
2.	GFI	.999
3.	AGFI	.995
4.	NFI	.999
5.	RMR	.010
6.	CFI	1.000
7.	RMSEA	.000

6.7.3 One factor measurement model for Flow principles

The latent variable of FL holds four items FL1 to FL6. The ratio of chi square to degree of freedom test (χ^2 /df) is 1.917, which specifies the data are good fitted. The values of other model fit indices like CFI= .986, NFI=.972, RMR=.040, AGFI= .949, GFI= .978, and RMSEA=.060 are well within standard limits. The measurement model of is found flow statistically significant as shown in Figure 6.5,table 6.11 and table 6.12. These outcomes recommended that the measurement model of flow principles found statistically good. Figure 6.5 depicts the CFA results for Flow principles.

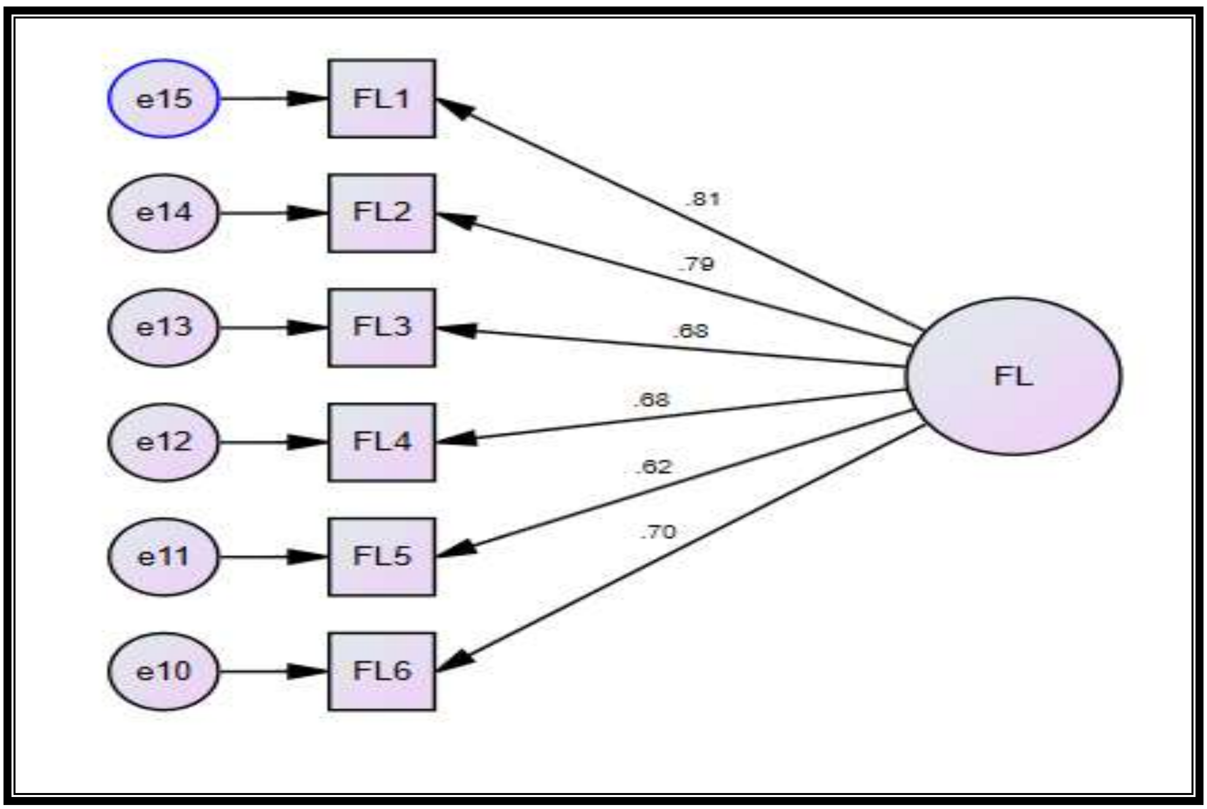


Figure 6.5: CFA results for Flow principles

Estimates and model fit indices are given below -

Table 6.11: Regression weights for Flow (FL) principles

			Estimates (Unstandardized)	Estimates (Standardized)	Standard Error	Critical Ratio	P
FL1	<---	FL	.938	.812	.082	11.475	***
FL2	<---	FL	.937	.791	.083	11.239	***
FL3	<---	FL	.832	.682	.085	9.846	***
FL4	<---	FL	.815	.679	.083	9.816	***
FL5	<---	FL	.754	.616	.084	8.963	***
FL6	<---	FL	1.000	.699			

***p $\leq$0.001

Table 6.12: Model fit indices for flow (FL) principle

S. No.	Model Fit Indices	Measurements
1.	χ^2 /df	1.917
2.	GFI	.978
3.	AGFI	.949
4.	NFI	.972
5.	RMR	.040
6.	CFI	.986
7.	RMSEA	.060

6.7.4 One factor measurement model for Pull Principle (PL)

The latent variable of PS holds four items PL01 to PL04. The ratio of chi square to degree of freedom test (χ^2 /df) is 2.741 (p $\leq$0.001) which specifies the data are good fitted. The values of other model fit indices like CFI= .991, NFI=.986, RMR=.044, AGFI=.948, GFI=.990,and RMSEA=.083 are well within standard limits. The measurement model of pull is found statistically significant as shown in Figure 6.6, table 6.13 and table 6.14. These outcomes recommended that the measurement model of Pull principles found statistically good. Figure 6.6 depicts the CFA results for Pull principle.

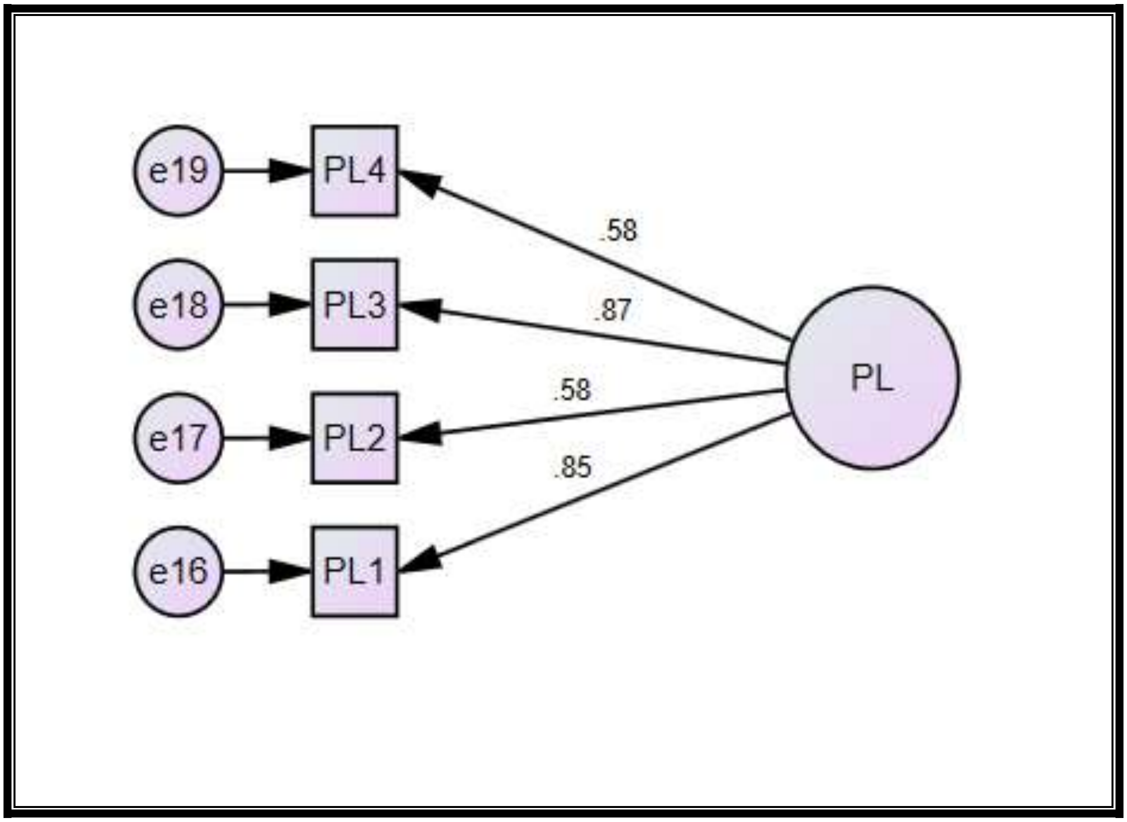


Figure 6.6: CFA results for Pull principle

Table 6.13: Regression weights for Pull (PL) principle

			Estimates (Unstandardized)	**Estimates (Standardized)**	**Standard Error**	**Critical Ratio**	**P**
PL1	<---	PL	.973	.853	.072	13.575	***
PL2	<---	PL	.780	.576	.084	9.309	***
PL3	<---	PL	1.000	.872			
PL4	<---	PL	.771	.575	.083	9.285	***

***p ≤0.001

Table 6.14: Model fit indices for pull (PL) principle

S. No.	Model Fit Indices	Measurements
1.	χ^2 /df	2.741
2.	GFI	.990
3.	AGFI	.948
4.	NFI	.986
5.	RMR	.044
6.	CFI	.991
7.	RMSEA	.083

6.7.5 One factor measurement model for Perfection (PF)

The latent variable of PF holds four items PF1 to PF4. The ratio of chi square to degree of freedom test (χ^2 /df) is 2.957(p $\leq$0.001) which specifies the data are good fitted. The values of other model fit indices like CFI= .991, NFI= .987, RMR= .049, AGFI= .942, GFI= .988, and RMSEA= .088 are well within standard limits. The measurement model of perfection is found statistically significant as shown in Figure 6.7, table 6.15, and table 6.16. Figure 6.7 depicts the CFA results for perfection principle.

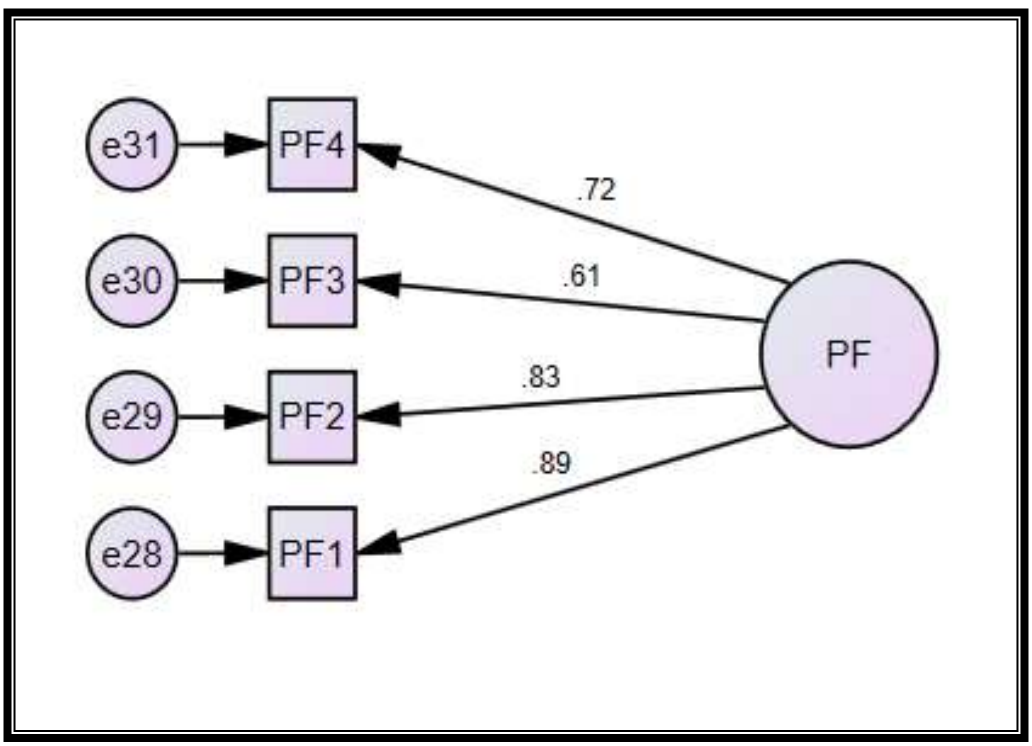


Figure 6.7: CFA results for perfection principle

Table 6.15: Regression weights for Perfection (PF)

			Estimates Unstandardized	Estimates Standardized	Standard Error	Critical Ratio C.R	P
PF1	<---	PF	.865	0.89	.059	14.761	***
PF2	<---	PF	1.000	0.83			
PF3	<---	PF	.710	0.61	.072	9.917	***
PF4	<---	PF	.744	0.72	.061	12.141	***

***$p \leq 0.001$

Table 6.16: Model fit indices for Perfection (PF)

S. No.	Model Fit Indices	Measurements
1.	χ^2 /df	2.957
2.	GFI	.988
3.	AGFI	.942
4.	NFI	.987
5.	RMR	.049
6.	CFI	.991
7.	RMSEA	.088

6.7.6 One factor measurement model for Responsiveness (RS)

The latent variable of RS holds four items RS1 to RS4. The ratio of chi square to degree of freedom test (χ^2 /df) is .214 ($p \leq 0.001$) which specifies the data are good fitted. The values of other model fit indices like CFI= 1.000, NFI= .999, RMR=.007, AGFI= .996, GFI=1.000, and RMSEA= .000 are well within standard limits. These outcomes recommended that the measurement model of responsiveness principle found statistically good. The factor loading of each variable is above 0 .507(standard) which support the construct validity of construct (RS). The measurement model of responsiveness is found statistically significant as shown in Figure 6.8, table 6.17 and table 6.18. Figure 6.8 depicts the CFA results for responsiveness.

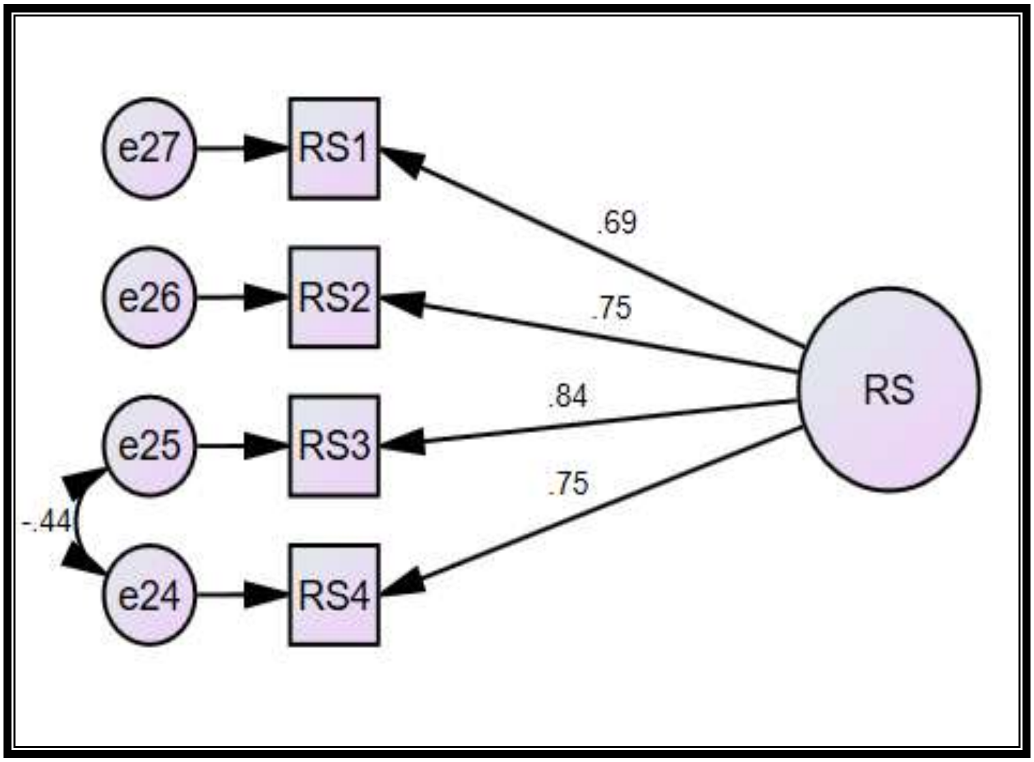


Figure 6.8: CFA results for Responsiveness

Table 6.17: Regression weights for Responsiveness (RS)

			Estimates Unstandardized	**Estimates Standardized**	**Standard Error**	**Critical Ratio**	**P**
RS1	<---	RS	.655	.691	.066	9.861	***
RS2	<---	RS	.827	.748	.080	10.386	***
RS3	<---	RS	1.000	.837			
RS4	<---	RS	.741	.746	.074	10.045	***

***p ≤0.001

Table 6.18: Model fit indices for Responsiveness (RS)

S. No.	Model Fit Indices	Measurements
1.	χ^2 /df	.214
2.	GFI	1.000
3.	AGFI	.996
4.	NFI	.999
5.	RMR	.007
6.	CFI	1.000
7.	RMSEA	.000

6.7.7 One factor measurement model for Operational Performance (OP)

The latent variable of OP holds four items OP1 to OP4. The ratio of chi square to degree of freedom test (χ^2 /df) is .414 ($p \leq 0.001$) which specifies the data are good fitted. The values of other model fit indices like CFI=1.000, NFI= .998, RMR= .023, AGFI= .992, GFI= .998, and RMSEA= .000 are well within standard limits. The measurement model of operational performance is found statistically significant as shown in Figure 6.9, table 6.19 and table 6.20.These outcomes recommended that the measurement model of operational performance principles found statistically good. Figure 6.9 depicts CFA results for Operational Performance.

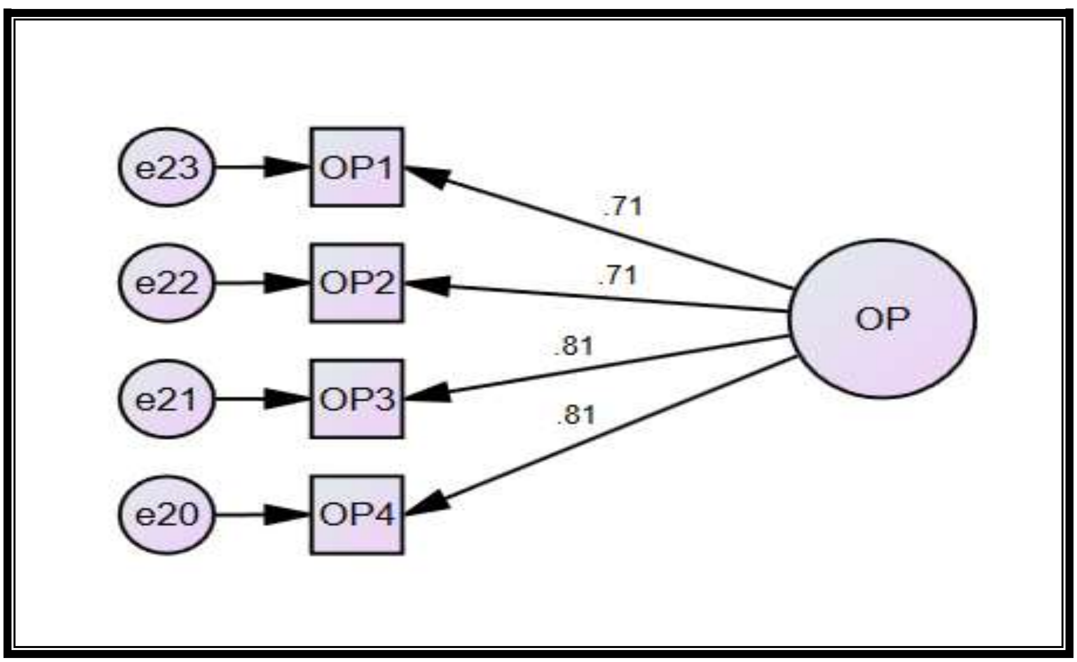


Figure 6.9: CFA results for Operational Performance

Table 6.19: Regression weights for operational Performance (OP)

			Estimates (Unstandardized)	Estimates (Standardized)	Standard Error	Critical Ratio	P
OP1	<---	OP	.962	.708	.085		***
OP2	<---	OP	.846	.711	.074		***
OP3	<---	OP	1.000	.815			
OP4	<---	OP	.963	.810	.075		***

***p ≤0.001

Table 6.20: Model fit indices for Operational performance (OP)

S. No.	Model Fit Indices	Measurements
1.	χ^2 /df	.414
2.	GFI	.998
3.	AGFI	.992
4.	NFI	.998
5.	RMR	.023
6.	CFI	1.000
7.	RMSEA	.000

6.8 Multifactor measurement model for confirmatory factor analysis

Multifactor measurement model is further developed with prime target to investigate the discriminant and construct validity as confirmatory factor analysis for the prior specification of items to their corresponding latent construct and uniqueness of constructs.

6.8.1 First order measurement model

The analysis is done by using trial version IBM AMOS 26.0. First order model for lean SD project is developed by confirmatory factor analysis (CFA) as depicted in Figure 6.10.The first order CFA model proposes that there are six lean principles (VL, FL,VS,PL,PF, and RS).These constructs are measured by five, six, four, four, four and four items respectively.

6.8.2 Reliability and validity of measurement models

In this research, validity and reliability of reflective constructs are tested using standardized estimates, AVE (Average variance extracted), R^2 (Squared multiple correlations) and CR (Composite reliability). AVE is a dimension of convergent validity. Composite reliability (CR) is assessed by Cronbach's Alpha to measure internal consistency. Communality is measured to find degree of variation in an item which is described by the construct and which is mentioned to as variance extracted from the item.

Shah and ward (2007) suggested that CFA provides a test of convergent and discriminant validity. Convergent validity test is conducted to assess how a particular item behaves within the block of items intended to measure a latent variable.

Hair *et al.*, 2013 defined discriminant validity as the degree to which a factor is really different from the other item also the degree by which an item is related to a construct. Shah and ward (2007) revealed that discriminant validity is assessed by constructing models for all possible pairs of latent variables.

The quality of the measurement model is assessed by convergent validity, composite reliability and discriminant validity. All standardized loading estimates are found above 0.5 value. Nunnally (1978) asserted that an alpha score bigger than 0.7 is commonly satisfactory and show adequate accuracy for a construct. After building up the CFA model for each of the construct, convergent validity is assessed based on significance level of factor loadings.

Average variance extracted (AVE) reflects the measure of convergence among set of indicators representing a latent construct. Average variance extracted (AVE) is a measure of the amount of variance that is captured by a construct in relation to the amount of variance due to measurement error. AVE is calculated based on a congeneric measurement model. Composite reliability, AVE and Cronbach's alpha are used for validity of constructs. The relation to calculate AV E is given in equation 1 and 2 as given below-

$$\textbf{Composite Reliability}(\textbf{CR}) = \left(\sum_{i=1}^{n} \lambda_i\right)^2 / \left[\left(\sum_{i=1}^{n} \lambda_i\right)^2 + \left(\sum_{i=1}^{n}\left(1 - \lambda_i{}^2\right)\right)\right] \qquad (1)$$

$$\textbf{Average Variance Extracted } (\textbf{AVE}) = \left(\sum_{i=1}^{n} \lambda_i{}^2\right)/n \qquad (2)$$

Where λ is the standardized factor loading of items and n is the number of items connected with particular construct.

It is observed that VL, VS, FL, PL, PF, and RS have AVE more than 0.5 which is found satisfactory. Hair *et al.*, 2013; Shah and Goldstein, 2006 suggested that AVE≥0.5 is considered as reference value. AVE values under 0.5 which demonstrates a lack of convergent validity. Composite reliability is better option than Cronbach Alpha for path analysis (Anderson& Gerbing, 1988).

According to Hair *et al.* (2006) discriminant validity is the degree to which a factor is actually different from other factors. In discriminant validity, AVE is greater than MSV (Maximum shared variance in the range 0.16-0.28). The CFA diagram is shown in Figure 6.10 which specifies the number of factors (latent variables). The co-variances among latent factors can be estimated through CFA diagram and table 6.21 represents the CFA results for measurements model.

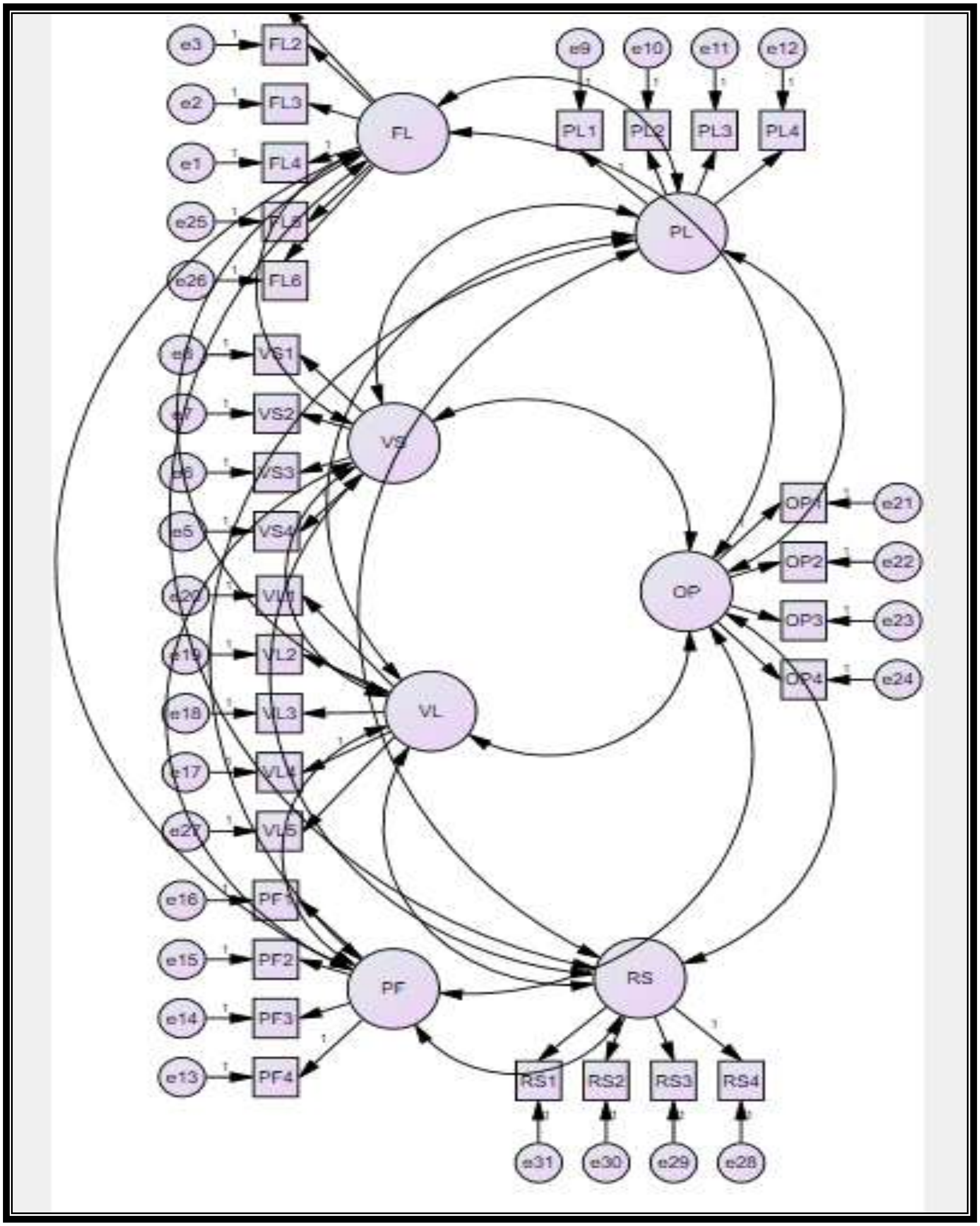


Figure 6.10: CFA Diagram for constructs

Table 6.21: CFA results for measurements model

First order latent variable	Manifest variable	Manifest variable loading	R^2	AVE	CR Composite Reliability	A
VL	VL1	0.884	.354	**0.538**	**0.851**	**0.849**
	VL2	0.701				
	VL3	0.65				
	VL4	0.67				
	VL5	0.74				
VS	VS1	.720	.423	**0.583**	**0.848**	**0.847**
	VS2	.802				
	VS3	.831				
	VS4	.695				
PL	PL1	.853	.233	**0.537**	**0.817**	**0.803**
	PL2	.576				
	PL3	.872				
	PL4	.575				
FL	FL1	.812	.574	**0.513**	**0.862**	**0.858**
	FL2	.791				
	FL3	.682				
	FL4	.679				
	FL5	.616				
	FL6	.699				
PF	PF1	.720	.022	**0.592**	**0.852**	**0.840**
	PF2	.610				
	PF3	.830				
	PF4	.890				
RS	RS1	.713	.400	**0.546**	**0.827**	**0.825**
	RS2	.789				
	RS3	.776				
	RS4	.672				

Where R^2 represents squared multiple correlations and α implies Chronbach's alpha.

6.8.3 Construct Validity

Table 6.22 depicts the summary of factor matrices for lean constructs. This represents KMO and Eigen value of lean constructs. Figure 6.11 depicts the SEM diagram.

Table 6.22: Summary of Factor matrices for higher level lean constructs

Construct	KMO	% Variance	Eigen Value
Value (VL)	0.833	62.58	3.128
Value Stream (VS)	0.815	68.57	2.743
Flow (FL)	0.879	59.17	3.550
Pull (PL)	0.752	64.19	2.568
Perfection (PF)	0.789	68.54	2.742
Responsiveness (RS)	0.779	65.82	2.633

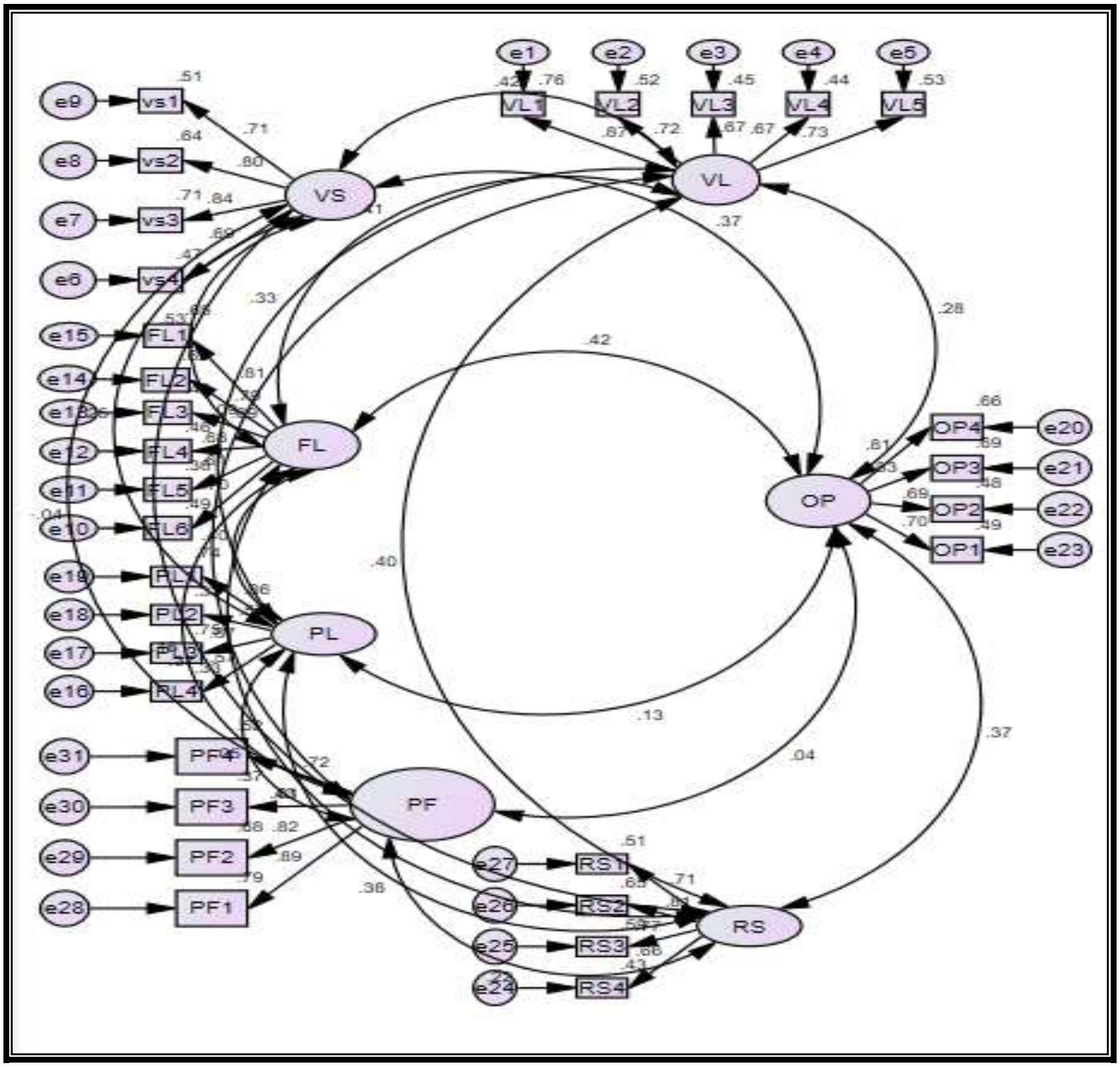


Figure 6.11: SEM Diagram

6.8.4 Co-variance among constructs

The primary requirements for this type of modelling are that all first-order constructs should have a significant correlation. The role of independent variable like VL, VS, FL, PL, PF and RS are examined. As it is observed that critical ratio are found more than 1.96 for 95% confidence level. These values are 5.586 between FL& VS; FL and VL have 4.736; FL & RS comprise 4.835; VS and VL have C.R value 4.828; VS and RS has 4.296; VL and RS has 4.577; PL and VL has 4.112; VS and RS has 4.296; VL and RS has 4.577; PL and VL has 4.112; PL and RS has 4.586; FL and PL have 4.827; VS and PL has 3.269. This shows that relationship are positive and significant at the $p<0.001$ level. This shows that constructs are positively associated. Table 6.23 depicts the covariance and their significance.

Table 6.23: Covariance of constructs

			Estimate	S.E.	C.R.	P
FL	<-->	VS	.376	.067	5.586	***
FL	<-->	VL	.220	.047	4.736	***
FL	<-->	PF	.118	.056	2.116	.034
FL	<-->	RS	.275	.057	4.835	***
VS	<-->	VL	.250	.052	4.828	***
VS	<-->	PF	-.035	.060	-.582	.560
VS	<-->	RS	.256	.060	4.296	***
PF	<-->	VL	.058	.045	1.268	.205
VL	<-->	RS	.211	.046	4.577	***
PL	<-->	VL	.202	.049	4.112	***
PL	<-->	PF	.040	.063	.639	.523
PL	<-->	RS	.280	.061	4.586	***
PF	<-->	RS	.164	.057	2.861	.004
FL	<-->	PL	.297	.062	4.827	***
VS	<-->	PL	.206	.063	3.269	.001

6.8.5 Second Order Measurement Model

Second order structural model was created to assess the relationship between lean principles and operational performance. Reflective construct model was used to create Lean. All first order correlation should be significant. Table 6.24 represents the Pearson correlations between lean constructs. It can be seen that correlation is significant at the level of 0.01.Thus the analysis offers that second order latent construct exists.

Table 6.24: Pearson Correlations between lean constructs

	VL	**VS**	**FL**	**PL**	**PF**	**RS**
VL	1.00	0.36**	0.36**	0.27**	0.09	0.35**
VS	0.36**	1.00	0.44**	0.23**	-0.01	0.299**
FL	0.36**	0.44**	1.00	0.33**	0.17	0.364**
PL	0.27**	0.23**	0.33**	1.00	0.07	0.308**
PF	0.09	-0.01	0.17	0.07	1.00	0.188**
RS	0.35**	0.30**	0.364**	0.31**	0.19	1
**. Correlation is significant at the 0.01 level (2-tailed).						

6.9 Multi Structural Congeneric model to analyse the impact of Lean principles on operational Performance

Multi Structural Equation Model for Operational Performance is depicted in figure 6.12. To test the research model, a structural equation modeling approach was used.

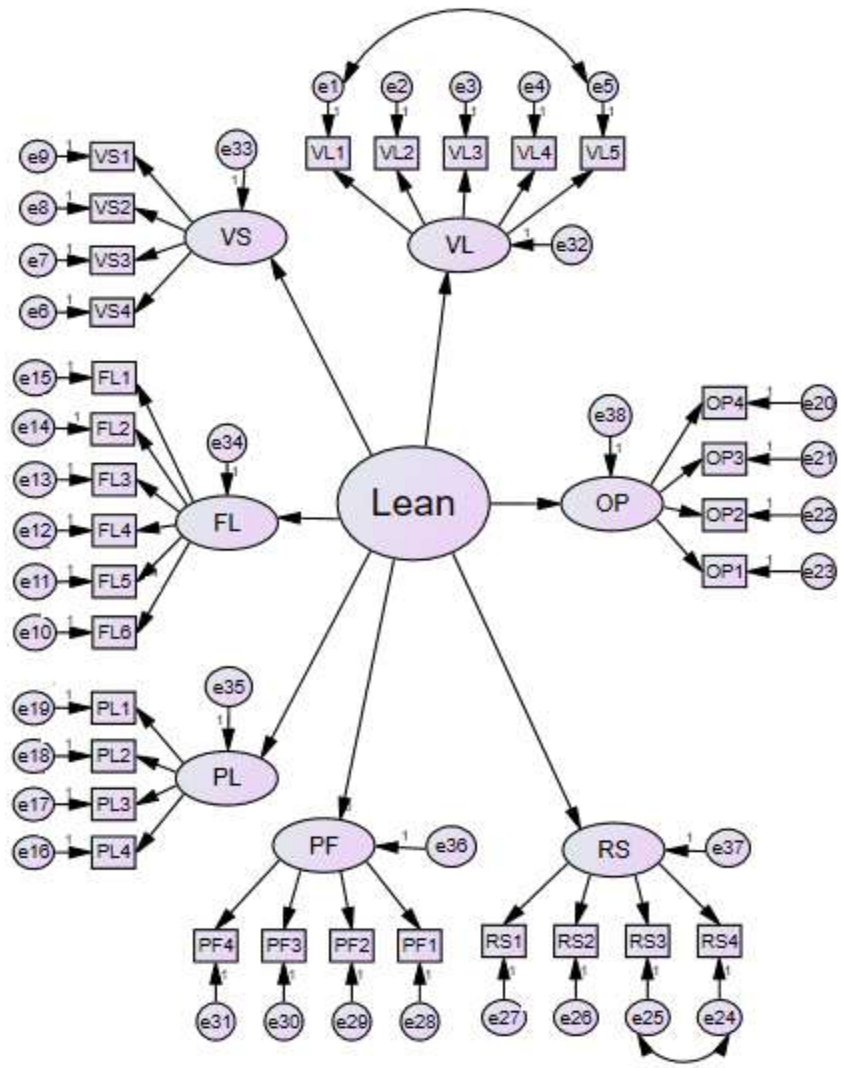


Figure 6.12: Multi Structural Equation Model for Operational Performance

All structural equations modeling analyses were carried out employing IBM AMOS 26.0, within SPSS16.0. First, the overall model fit is evaluated by χ^2 /df and root mean square error of approximation (RMSEA). Both these measures assess how well a priori model replicates the sample data. The proposed structural model: combined measurement and path model fits well as shown by the model fit statistics: RMSEA=0.039, χ^2 /df = 1.385, while the suggested cutoff points are RMSEA < 0.08 and χ^2 /df < 3(Hair *et al.*, 2006). Table 6.25 presents model fit indices of constructs.

Table 6.25: Model fit indices of constructs

S. No.	Model Fit Indices	Measurements
1.	χ^2 /df	1.385
2.	GFI	.876
3.	AGFI	.851
4.	NFI	.857
5.	RMR	.088
6.	CFI	.955
7.	RMSEA	.039

After presenting the second-order construct (lean), the goodness-of-fit indices for the structural model were χ^2 /df=1.385, GFI =.876 which is very near to 0.9, Likewise, AGFI =.851, RMR =0.088, NFI =.857 are very close to 0.9; CFI =.955 and RMSEA = .039 and were found to be within acceptable limits. The operational performance construct was also subjected to CFA and was also found to be satisfactory. For this reduced model, all the fitness indices had acceptable values and the standardize coefficient found significant ($p<0.01$). Subsequently, the full structural model was subjected to the goodness of fit tests. All the fitness indices for the structural model found acceptable. Figure 6.13 represents the relationship among lean principles,lean thinking and Operational Performance.

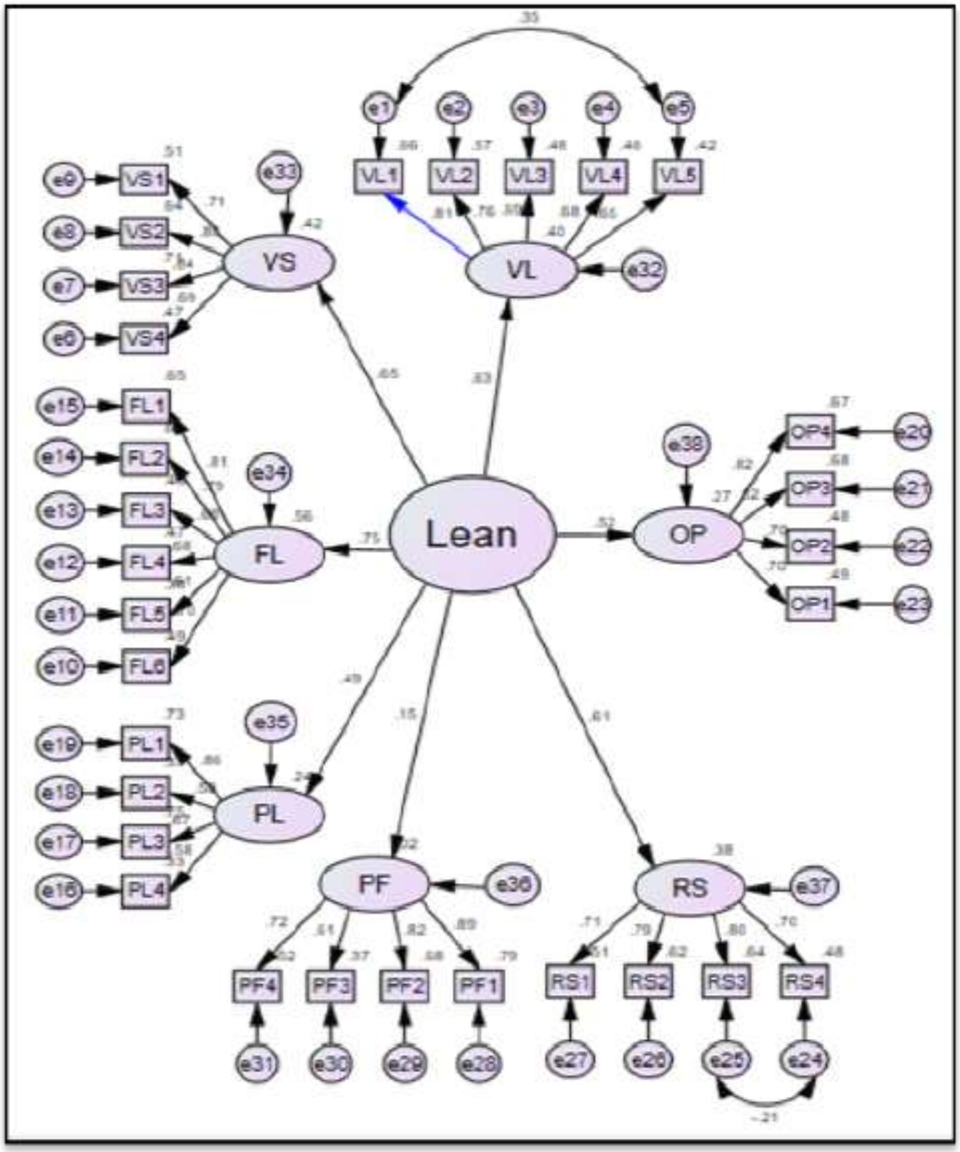


Figure 6.13: Relationship among lean principles, lean thinking and Operational Performance

The path between Lean and operational performance was significant. Structural equation modeling is used to test the research model. From the statistical analysis, it was observed that the proposed model is fit according to the model fit indices. From the model, it was found that all lean principles are positively associated with lean, except perfection..

6.10 Discussion of hypotheses among the research constructs

As it is evident from table 6.26, the relationship between lean principles and lean thinking is statistically significant. Hypotheses H4, H5, H6, H7, and H9 are accepted and it implies that the relationship between lean principles and lean thinking is significant. H10 is accepted which means lean thinking is positively associated with operational performance of SD projects

Table 6.26: Result of hypothesized relationship in structural model

Hypotheses				Estimate Unstandardized	Estimates Standard	S.E.	C.R.	P	Remarks
H4	VL	<---	LEAN	.713	.595	.113	6.325	***	Supported
H5	VS	<---	LEAN	.769	.650	.125	6.162	***	Supported
H6	FL	<---	LEAN	1.000	.758				Supported
H7	PL	<---	LEAN	.473	.483	.096	4.930	***	Supported
H8	PF	<---	LEAN	.218	.150	.113	1.925	.054	Not Supported
H9	RS	<---	LEAN	.663	.633	.112	5.933	***	Supported
H10	OP	<---	LEAN	.924	.523	.163	5.671	***	Supported

H4 is considered to analyze the role of VL on Lean.

It means Value (VL) principle enables lean thinking in SD projects and significant at $p<0.001$.

H5 is considered to analyze the role of VS on Lean.

The relationship between VS and LEAN is positive and significant at $p<0.001$ levels, which implies that hypothesis H5 is strongly supported. It implies that Value Stream (VS) enables lean thinking in SD projects. It means Value Stream (VS) principle enables lean thinking in SD projects.

H6 is considered to analyze the role of FL on Lean.

The relationship between VS and LEAN is positive and significant at p<0.001 levels, which implies that hypothesis H6 is strongly supported. It implies that Value Stream (VS) enables lean thinking in SD projects. It means Flow (FL) principle enables lean thinking in SD projects.

H7 is considered to analyze the role of PL on Lean.

The relationship between PL and LEAN is positive and significant at p<0.001 levels, which implies that hypothesis H7 is strongly supported. It implies that Pull (PL) enables lean thinking in SD projects. It means Pull (PL) principle enables lean thinking in SD projects.

H8 is considered to analyze the role of PF on Lean.

The relationship between PL and LEAN is positive and significant at p<0.001 levels, which implies that hypothesis H8 is not supported. It implies that Perfection (PF) enables lean thinking in SD projects.

H9 is considered to analyze the role of RS on Lean.

The relationship between RS and LEAN is positive and significant at p<0.001 levels, which implies that hypothesis H9 is strongly supported. It implies that Responsiveness (RS) enables lean thinking in SD projects. It means responsiveness (RS) principle has positive effect on operational performance of SD projects.

H10 is considered to analyze the role of Lean thinking on OP.

The relationship between LEAN on Operational performance is positive and significant at p<0.001 levels, which implies that hypothesis H10 is strongly supported. It implies that lean thinking has positive effect on Operational performance (OP) in SD projects. The result shows how lean principles are correlated with lean thinking and help in improving the organizational performance among the SD projects being surveyed.

6.11 Conclusions

A model was developed in implementing lean principles in SD projects to help IT experts. It is investigated that application of lean principles are positively associated with the operational performance of SD projects. In manufacturing, all core five lean principles (value, value stream, flow, pull and perfection) have been applied, while in SD organizations value, pull and flow are found to be more visible. A little impact of VSM, pull and perfection on operational performance was observed. It implies that lean is applied partially in the SD projects rather than full in the manufacturing industries. In SD projects, value and flow are focused as lean principles.

The success of SD projects depends largely on a set of lean practices and principles which practitioners consider important for success. Therefore, it is necessary to investigate the degree of lean adoption of the lean practices/principles from the perspective of the practitioners for a successful process improvement program and to assess the actual status of such practices/principles. Most of the SD projects focus on value, flow, and pull from lean principles and responsiveness. In contrast to lean manufacturing, lean is adopted partially in software development projects.

This study instigates the status of lean principles in practitioners' perception toward the adoption of the lean in SD projects. Thus, most of the SD organizations put more focus on value, flow, pull and less focus on VSM; hence perfection is not completely adopted as in manufacturing organizations. Kundu and Manohar (2016) also supports this result that more focus is observed in value oriented practices to enhance value principle.

Impact of lean principles on operational performance is investigated through empirical analysis. It is observed that value, flow; VSM is positively associated with lean. However, pull and perfection are not closely associated and not creating more impact on operational performance. Our results are in line with exploratory studies carried out in five case organizations, in which value and flow creating principles are dominating and creating positive influence on operational performance. The findings also reveals that adoption of lean principles improves customer satisfaction, reduce lead time, WIP and cost.

Chapter 7

Conclusion and Discussions

Chapter 7

Conclusion and Discussions

7.1 Background

The final chapter outlines the conclusion of the thesis with contribution of the thesis to the body of research on lean adoption in software development projects. It also highlights practical and managerial implications of the research for practicing managers. Lastly, limitations of the research and recommendations for future research are presented.

It has been found that a lot of empirical studies as well as case studies have been carried out on lean implementation in manufacturing and other sectors such as product development, construction, healthcare, and insurance and banking, but very little work have been reported for lean in SD projects. There have been a few case studies on lean in SD projects, but very few empirical studies are observed. A set of lean principles/practices used in manufacturing have already been well established, but it may not be applicable to SD projects in their existing form. It is always required to examine the effects of management initiatives on organizational performance. Stass *et al.* (2011) found that lean in software services can gain operations based advantages. The importance of SD projects is illustrated by Bhatnagar (2006), who estimates that the software and service sector accounts for over 20 percent of India's total exports and 2.6 percent of GDP. Next section presents the summary of the research.

7.2 Summary of the Research

A background of lean in software development was presented and need for further research was identified in the introductory chapter. Second chapter provided a comprehensive review of literature on adoption of lean in SD projects/ IT support service which shows that very little work has been done in the area. There is a lack of research concerning the identification of specific lean principles/practices that are suitable or implementable in SD projects. There have been a few multi-case studies on lean in SD projects in the literature (Kupiainen *et al.*, (2015) and few empirical studies principally focusing on adoption of lean practices (Kundu and Manohar, 2016).

However, no study was found that directly or indirectly measured the effect of lean initiatives. Therefore, it was decided to study the status of lean implementation in SD projects.

By reviewing literature, gaps were identified in existing knowledge. Analysis of currently available literature on lean software development (LSD) was done. The following were the objectives of present study:

- To perform exploratory study to examine the implementation of lean in SD projects.
- To identify drivers and barriers of implementing lean in SD projects.
- To develop a model for lean implementation in Software Development projects.

Case study was chosen for exploratory study for intensive investigation of application of lean in software development. LSD focuses on lean thinking in software development organizations. From five exploratory case studies it was found that lean was adopted in different forms in SD projects. Further, it was found from research that value and flow principles are more relevant in SD organizations, while value stream, pull, and perfection are not conspicuous. This may be due to intangible deliverables and non-repetitiveness.

After reviewing literature and findings from exploratory studies, hypotheses were formulated. These hypotheses were tested by constructing a survey instrument. The questionnaire was designed and segregated into five sections viz. demographic, degree of lean adoption, drivers, barriers, and operational performance. A pilot study was carried out for testing the questionnaire in 35 organizations.

Further, descriptive analysis of data gathered and survey analysis of adoption of lean principles, barriers and drivers in SD projects were carried out. Confirmatory factor analysis (CFA), and SEM were employed using IBM AMOS 26.0 to test the impact of lean thinking on operational performance. Validity and model fit indices were evaluated via confirmatory factor analysis (CFA) measurement model having an initial phase of SEM. The structural model was formulated and applied to carry out the final step of SEM. After this, limitations and directions for future research were proposed.

It is observed that limited numbers of lean principles are adopted in SD projects from the perspective of creating an impact on operational performance. This implies that lean is adopted partially in SD projects. Lean is positively associated with operational performance. Finally, a set of guidelines is presented to implement lean principles in SD projects. Lastly, all the research objectives are achieved in current studies.

7.3 Concluding Remarks

This study identifies the lean practices in product and project-based organizations driven by value and flow principles. SD project-oriented organizations adopted mainly VSM, Kanban, DSM, concurrent and integrated PD, and cause & effect matrix.

It was found from research that three types of challenges are identified, which hinder the adoption of lean in SD projects. These are categorized as customer-oriented, process-oriented, and people-oriented challenges. In process-oriented challenges, repetitive testing and debugging are prominent. On the other hand, prioritization between bug fixing and adding new features are major process-oriented challenges. It is hard to integrate planning, design, system, subsystem, testing, and delivery. In customer-oriented challenges, it was identified that vague requirements and frequent changes in the requirements are found to be the most important issues in lean implementation. Likewise, in people (team) oriented challenges, it was observed that developing an effective cross-functional team, lack of communication among cross-function team and the existence of silos are major challenges in lean implementation. After exploring important lean practices and challenges, other important issues like drivers and barriers and adoption of lean principles were investigated in a large number of SD organizations.

7.4 Theoretical Contribution

The research provides a vital evaluation of the existing literature on lean software development. Based on an extensive review of literature, research gaps were identified and an abstract model on the adoption of lean in SD projects was formulated. This model represents one of the first endeavors to combine lean principles with the responsiveness view in lean adoption in SD projects. It was investigated that interrelationship among lean constructs, lean thinking, and organizational performance was a worthy contribution for improvement of performance in SD project. The outcome of the model formulated in this study confirmed that lean principles are strongly enabled with lean thinking, which directly affects the organization's performance. The empirical results verify that the lean principles are effective in lean adoption in SD projects. This research builds a contribution to the literature by examining the measurement model in SD projects.

7.5 Research Implications

In manufacturing, all core lean principles (value, value stream, flow, pull, and perfection) are applied, while in SD organizations, value, responsiveness and flow are found to be more visible.

When a larger set of SD companies was investigated, value, flow, and responsiveness are found more visible. A little impact of VSM, pull and perfection was visible. This implies that lean is applied partially in the SD projects rather than full lean as found in manufacturing. The research may be advanced towards developing strategies for improving lean implementation in SD projects by focusing on these principles. Value and flow could be used by the agile research community in agile SD under the umbrella of scrum and sprint to achieve a transformation to "Leagile."

7.6 Practical Implications

The research reveals that value and flow are the most relevant lean principles for SD projects. Thus, the software industry interested in lean implementation should focus on the implementation of value and flow enhancing practices. Value enhancing practices include mistake proofing, team-based problem solving, VSM, standardized work, continuous improvement, and visual control. Some of the flow enhancing practices are flexible and cross-functional teams, Kanban/Kanban board, JIT information, cause and effect matrix, and set base concurrent development. These practices can be implemented according to the nature of the organization, i.e., product-based, and project-based SD organizations. Value enhancing practices are useful in the improvement of products and processes. Flow enhancing practices contribute to the reduction of WIP and help to detect bottlenecks early in the dynamic environment and in rapidly changing SD products. The Lean practices may be effective in bug tracking, testing, and to reduce release time. These practices can also be useful in supporting software process improvements.

A model was developed in implementing lean principles in SD projects to help IT experts. Further, this research also provides the empirical evidence for the relationship of lean principles, lean thinking, and performance determinants, which may help in SD organizations to improve the desired performance level. A Strategy can be adopted that aims to achieve the performance targets. It is recommended that value-oriented practices, flow-oriented practices, and responsiveness measure should be significantly focused for performance improvement as compared to their competitor.

7.7 Set of recommendations: guidelines in implementing lean in software development Projects

It was observed from research that lean practices are adopted on the basis of characteristics of organizations. Companies should implement different lean practices depending on the characteristics of the organization, e.g., product-based or project-based organization. We believe that value enhancing practices should be the first step in SD organizations. The core value enhancing practices are "waste elimination practices," such as mistake proofing, visual control, continuous improvement, and VSM. It should be followed by systematic implementation of flow enhancing practices such as work standardization, Kanban/Kanban Board, JIT information, and set-based concurrent development. Similar to manufacturing, pull, and sequencing and work load levelling can be applied. This study suggests the implementation of value enhancing practices and flow enhancing lean practices in SD organizations. There may be more practices that can be included depending upon the characteristics and performance improvement objectives of an SD organization.

Customer-related issues such as a change in customer requirements and vague requirements should be handled before lean implementation in SD organizations. Customer-oriented issues can be resolved by limiting the inventory of requirements and early test to overcome mistakes in requirements along with faster detection of bugs. Further, complex requirements and large SD projects can be broken down into sprints. The sprints may be broken into stories which comprise of several features in the development process of SD products. Likewise, process related issues such as non-value activities, difficulties in prioritizing bug fixing, and adding new features should also be taken care of. The process-oriented issues may be resolved by using standardized procedures. People related issues such as resource constraints and communication among the organization, business analyst, and customer should be resolved while implementing lean. The team-oriented issues can be resolved by using multi-skilled employees to balance workload and to eliminate bottlenecks.

Our observations of the details of the implementation will hopefully provide the beginnings of a roadmap for other SD projects seeking to apply lean.

7.8 Managerial Implications

Lean adoption module can be created for value and information flow management in SD projects. Value creation module in SD projects could be developed by fulfilling customer-requirements, creative involvement of the customer in deliverables, and collecting feedback from the customers**.** Further, team-related practices could enhance the value through adoption of continuous integration, team-based problem solving, knowledge acquisition through KIVP (knowledge innovation visible planning), picking mature framework or platforms to expedite the work, and continuous improvement to enhance the quality of software. It is recommended for project managers to adopt task oriented practices such as standardized work to set SOP (standard operating procedures), mistake proofing for developing mistake proof code and VSM to reduce undesired iterations as well as for determining waste/ rework. Task-oriented emerging practices could be used to enhance value in software development. While developing software some process improvement practices such as bug fixing, periodic code review, pre-build solutions, third-party modules, and standard code libraries, a standard error code and testing of new features could be adopted.

Project managers are suggested to adopt some task-oriented activities for Information flow management. These activities include lean practices such as visual management for task prioritization, and design structure matrix for optimization of information. Likewise, cause and effect matrix to know the root causes of the problem and current state of the process, and time based management of task can also be used. Process-based management practices can be adopted to manage the information flow. Process-oriented practices such as KANBAN, knowledge stream mapping, and set-based concurrent development are also useful to improve the flow of information. Team oriented practices such as creating cross-functional teams and aligning the organization through communication and coordination of the development team, are two important practices to enhance the flow of information. Project management and testing practices are other methods to improve the flow of information. Project management tools and automation of testing, prototype program/navigational prototype/ functional prototype)/Virtual prototype; pulling of customized interim deliverables are useful in the adoption of flow principle.

Lean can be adopted in a holistic way by adopting various steps. The steps are covered by considering various measures such as hiring a lean team in SD organization, lean training to project manager/project leaders/B.A, define lean assessment metrics (quality, cost and delivery), develop current value stream map, identify non value activities(NVA) /defects /error , correlate waste with lean practices to eliminate NVA, identify major lean practices; identify major lean principles ,develop future value stream map, arrange training of employee, lean implementation in SD project by removing barriers, lean assessment/evaluation and performance measurement

7.9 Research Limitations and Scope for future research

Data for this research work was primarily collected from the project manager and senior management dealing with the planning, monitoring, and control of projects, presuming that their judgment concerning lean adoption is objective. There may be over-enthusiastic responses or under-describing of few lean principles/practices by respondents to justify their job area. Within the extent of this research, existing inner resources were emphasized, while external parameters viz. social and economic performances were not studied. It is hard to attain an entirely random sample; it may possess a few flaws. The research may suffer from possible prejudices connected with case studies with a single respondent like a project manager.

A framework may be developed for lean implementation in SD projects, which may incorporate lean adoption stages. Further research is needed to scientifically explore the impact of lean adoption in SD projects and to accurately find out a particular set of lean practices/principles for SDP improvement. Effect of responsiveness on individual parameters of operational performance can be evaluated.

References

References

Abreu-Led_on, R., Luján-García, D.E., Garrido-Vega, P. and Escobar-Pérez, B. (2018),“AMeta-analytic study of the impact of lean production on business performance”, International Journal of Production Economics, Vol. 200, pp. 83-102.

Achanga, P., Shehab, E., Roy, R. and Nelder, G. (2006) ‘Critical success factors for lean implementation within SMEs’, *Journal of Manufacturing Technology Management*, 17(4), pp. 460–471.

Achanga, P., Shehab, E., Roy, R. and Nelder, G. (2012), “A fuzzy-logic advisory system for lean manufacturing within SMEs”, International Journal of Computer Integrated Manufacturing, Vol. 25 No. 9, pp. 839-852.

Anand, G. and Kodali, R. (2010a),“Development of a framework for implementation of lean manufacturing systems”, International Journal of Management Practice, Vol. 4 No. 1, pp. 95-116.

Anand, G. and Kodali, R. (2010b), “Analysis of Lean manufacturing frameworks”, Journal of Advanced Manufacturing Systems,Vol. 9 No. 1, pp. 1-30.

Anand, G., and R. Kodali. 2009. “Development of a Frameworkfor Lean Manufacturing Systems.” International Journal of Services and Operations Management 5 (5): 687–716.

Anand, G., Arpitha, C. and Narayanmurthy, G. (2014), “Business process reengineering through leanthinking: a case study”, Journal of Enterprize Transformation, Vol. 4 No. 2, pp. 123-150.

Anderson, D.J. (2003), “Agile management for software engineering: applying the theory of constraints for business results”, Prentice Hall Professional.

Anderson, D.J. (2010), “Kanban: successful evolutionary change for your technology business”, Blue Hole Press.

Arlbjørn, J. S., and P. V. Freytag(2013), “Evidence of Lean: A Review of International Peer-reviewed Journal Articles.” European Business Review vol. 25 No.2 pp.174–205.

Arlbjørn, J. S., P. V. Freytag, and T. Damgaard (2008),“The Beauty of Measurements.” European Business Review Vol. 20 No.2 pp. 112–127.

Atkinson, R., Crawford, L., & Ward, S. (2006),“Fundamental uncertainties in projects and the scope of project management”, *International journal of project management*, Vol.*24* No.8, pp.687-698.

Baines, T., Lightfoot, H., Williams, G. M., & Greenough, R.(2006). “State-of-the-art in lean design engineering: a literature review on white collar lean”. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, Vol. *220 No.*9, pp.1539-1547.

Ballard, G. (2008), “The Lean Project Delivery System: An Update”. *Lean Construction Journal*.

Bateman, N. and Rich, N. (2003),“Companies perceptions of inhibitors and enablers for process improvement activities”, *International Journal of Operations and Production Management*, Vol. 23 No. 2, pp. 185-199.

Bernardes, E.S. and Hanna, M.D., (2009), “A theoretical review of flexibility, agility and responsiveness in the operations management literature: Toward a conceptual definition of customer responsiveness”, *International Journal of Operations & Production Management*, Vo l*29 No.*1, pp.30-53.

Bhasin, S. (2008). “Lean and performance measurement”, *Journal of Manufacturing Technology Management*, *19*(5), pp.670-684.

Bhasin, S. (2012),“An appropriate change strategy for lean success”, *Management Decision*, *50*(3), pp.439-458.

Bhatnagar, S. (2006), “India’s software industry”, Technology, Adaptation, and Exports, Vol. 49

Bocock, L., & Martin, A. (2011), “There's something about lean: A case study”, In *2011 Agile Conference* (pp. 10-19). IEEE.

Brhel, M., Meth, H., Maedche, A. and Werder, K. (2015), “Exploring principles of user centered agile software development: a literature review”, Information and Software Technology, Vol. 61, pp. 163-181.

Bryde, D. J., & Schulmeister, R. (2012). “Applying Lean principles to a building refurbishment project: experiences of key stakeholders”. *Construction Management and Economics*, Vol.*30* No.9, pp.777-794.

Buesa, R.J. (2009), “Adapting lean to histology laboratories”, *Annals of Diagnostic Pathology*, Vol. 13 No. 5, pp. 322-333.

Coakes et al., (2005), “SPSS version12.0 for window Analysis without Anguish”, Willey India Ltd.

Cook, J. (2004), “Lean object-oriented software development”, Advanced Management Journal, Vol. 69, No. 2.

Corona, E. and Pani, F.E. (2012), ““An investigation of approaches to set up a Kanban board, and of tools to manage it”, In Proceedings of the 11th International Conference on Telecommunications and Informatics (TELEINFO’12) and the 11th International Conference on Signal Processing (SIP’12), Saint Malo, France, pp. 7-9.

Crute, V., Ward, Y., Brown, S. and Graves, A. (2003), “Implementing lean in aerospace: challenging the assumptions and understanding the challenges”, Technovation, Vol. 23 No. 12, pp. 917-928.

Cudney, E. and Elrod, C. (2011), “A comparative analysis of integrating lean concepts into supplychain management in manufacturing and service industries”, International Journal of LeanSix Sigma, Vol. 2 No. 1, pp. 5-22.

Dingsøyr, T., Moe, N.B., Fægri, T.E. and Seim, E.A. (2018), “Exploring software development at the very large-scale: a revelatory case study and research agenda for agile method adaptation”,b Empirical Software Engineering, Vol. 23 No. 1, pp. 490-520.

Doolen, T.L. and Hacker, M.E., (2005). A review of lean assessment in organizations: an exploratory study of lean practices by electronics manufacturers. *Journal of Manufacturing systems*, *24*(1), pp.55-67.

Dora, M., Kumar, M., & Gellynck, X. (2016), “Determinants and barriers to lean implementation in food-processing SMEs–a multiple case analysis”, *Production Planning & Control*, *27*(1), 1-23.

Dora, M., Kumar, M., Van Goubergen, D., Molnar, A., & Gellynck, X. (2013). “Operational performance and critical success factors of lean manufacturing in European food processing SMEs” , *Trends in Food Science & Technology, 31*(2), 156-164.

Ebert, C., Abrahamsson, P., & Oza, N. (2012). Lean software development. *IEEE Software*, (5), 22-25.

Eisenhardt, K.M. (1989), “Building theories from case study research”, Academy of Management Review, Vol. 14 No. 4, pp. 532-550.

Emiliani, M.L. (2006), “Origins of lean management in America-the role of Connecticut businesses”, Journal of Management History, Vol. 12 No. 2, pp. 167-184.

Esain, A., Williams, S. and Massey, L. (2008), “Combining planned and emergent change in a healthcare lean transformation”, *Public Money and Management*, Vol. 28 No. 1, pp. 21-26.

Gautam, N., Singh, N., 2008. “Lean product development: Maximizing the customer perceived value through design change (redesign)”. International J. Prod. Econ. 114, pp. 313–332.

Gopinath, S.and Freiheit, T.I. (2012), “A waste relationship model and the center point tracking metric for lean manufacturing systems”, IIE Transactions, Vol. 44 No. 2, pp. 136-154.

Gustavsson, H. and Axelsson, J. (2010), “Improving the System Architecting Process through the Use of Lean Tools”, Technology Management for Global Economic Growth (PICMET), 2010 Proceedings of PICMET’10, IEEE, pp. 1-7.

Hallam, C.R. (2003), “Lean enterprise self-assessment as a leading indicator for accelerating transformation in the aerospace industry”, Doctoral dissertation, Massachusetts Institute of Technology

Henrik Jonsson (2012), “Lean software development: a systematic review”, Research Methodology Course Mini-Conference.

Hines, P. and Lethbridge, S., (2008), “New development: creating a lean university”, Public Money Management, Vol. 28, pp. 53-56.

Hines, P., Francis, M. and Found, P. (2006), "Towards lean product life cycle management: a framework for new product development", Journal of Manufacturing Technology Management, Vol. 17 No. 7, pp. 866-887.

Hines, P., Holweg, M. and Rich, N. (2004), "Learning to evolve", International Journal of Operations and Production Management, Vol. 24 No. 10, pp. 994-1011.

Holweg, M. (2005),"The three dimensions of responsiveness", International Journal of Operations& Production Management, Vol. 25 Nos 7/8, pp. 603-22.

Holweg, M. (2007), "The genealogy of lean production", Journal of Operations Management,Vol. 25 No. 2, pp. 420-37.

Holweg, M., Pil, F.K., 2001. "Successful build-to-order strategies start with the customer". MIT Sloan Manag. Rev. 43, pp.74–83.

Houshmand, M., & Jamshidnezhad, B.(2006), "An extended model of design process of lean production systems by means of process variables". *Robotics and Computer-Integrated Manufacturing*, *22* pp. 1-16.

Iberle, K. (2010), "Lean system integration at Hewlett-Packard", Proceedings of the 28th Annual Pacific Northwest Software Quality Conference, Portland, OR October 18-19, pp. 187-204.

Ikonen, M., Kettunen, P., Oza, N. and Abrahamsson, P. (2010), "Exploring the sources of waste in kanban software development projects", Proceedings of the 36th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA 2010), Lille, France, pp. 376-384, September 1-3.

Jadhav, R. J., Mantha, S., and Rane B., S. (2014). "Exploring barriers in lean implementation". *International Journal of Lean Six Sigma*, Vol.*5* No.2, pp. 122-148.

Joseph Hair, J. F.Joseph F. Hair Jr. William C. Black Barry J. Babin Rolph E. Anderson, "Multivariate Data Analysis", Pearson Education India .Seventh Edition

Karvonen, T., Rodriguez, P., Kuvaja, P., Mikkonen, K. and Oivo, M. (2012), "Adapting the lean enterprise self-assessment tool for the software development domain", in Software

Engineering and Advanced Applications (SEAA), 2012 38th EUROMICRO Conference on IEEE, pp. 266-273.

Khan, M. S., Al-Ashaab, A., Shehab, E., Haque, B., Ewers, P., Sorli, M., & Sopelana, A. (2013). Towards lean product and process development. *International Journal of Computer Integrated Manufacturing*, *26*(12), 1105-1116.

Kline, R. B. (2015). ”*Principles and practice of structural equation modeling”*. Guilford publications.

Kobus, J. (2016). Demystifying lean IT: conceptualization and definition. *Materiały konferencyjne: Multikonferenz Wirtschaftsinformatik, Technische Universität Ilmenau, Ilmenau*.

Kundu, G. K., Murali Manohar, B., & Bairi, J. (2011). A comparison of lean and CMMI for services (CMMI-SVC v1. 2) best practices. *Asian Journal on Quality*, *12*(2), 144-166.

Kundu, G.K. and Manohar, M. (2016), “Prioritizing lean practices for implementation in IT support services”, VINE Journal of Information and Knowledge Management Systems, Vol. 46 No. 1, pp. 104-122.

Kundu, G.K., Manohar, B.M. and Bairi, J.,(2011), “IT support service: identification and categorisation of wastes”. *International Journal of Value Chain Management*, *5*(1), pp.68-91.

Kundu, G.K., Murali Manohar, B. and Bairi, J.,(2011), “A comparison of lean and CMMI for services (CMMI-SVC v1. 2) best practices”. *Asian Journal on Quality*, *12*(2), pp.144-166.

Kupiainen, E., Mäntylä, M.V. and Itkonen, J. (2015), “Using metrics in agile and lean software development – a systematic literature review of industrial studies”, Information and Software Technology, Vol. 62, pp. 143-163.

Kuusela, R. and Koivuluoma, M. (2011), “Lean transformation framework for software intensive companies: responding to challenges created by the cloud”, In Software Engineering and Advanced Applications (SEAA), 2011 37th EUROMICRO Conference on, IEEE, pp. 378-382.

Lambert, D.M. and Harrington, T.C., (1990), "Measuring nonresponse bias in customer service mail surveys". *Journal of Business Logistics*, Vol. *11* No.2, p.5.

Lewis, M. A. (2000), "Lean production and sustainable competitive advantage", InternationalJournal of Operations and Production Management, Vol. 20 No. 8, pp. 959-978.

Liker, J.K. (2004), "The 14 principles of the Toyota way: an executive summary of the culture behind TPS", The Toyota Way, Vol. 14, pp. 35-41.

Liker, J.K. and Morgan, J.M. (2006), "The Toyota way in services: the case of lean product development", Academy of Management Perspectives, Vol. 20 No. 2, pp. 5-20.

Lo, C.P. and Liu, B.J. (2009), "Why India is mainly engaged in offshore service activities, while China is disproportionately engaged in manufacturing?", China Economic Review, Vol. 20 No. 2, pp. 236-245.

Lyons, A. C., K. Vidamour, R. Jain, and M. Sutherland,(2011),"Developing an Understanding of Lean Thinking in Process Industries." Production Planning and Control 24 (6): 475–494.

Malladi, S., Dominic, P. D. D., and Kamil, A. (2011). Lean principles in IT services: a case study onimplementation and best practices, *International Journal of Business Information Systems*, 8(3), 247–268

Malmbrandt, M. and Åhlström, P. (2013),"An instrument for assessing lean service adoption", International Journal of Operations and Production Management, Vol. 33 No. 9, pp. 1131-1165.

Mandic, V., Oivo, M., Rodriguez, P., Kuvaja, P., Kaikkonen, H., and Turhan, B. (2010).What is flowing inlean software development? In P. Abrahamsson and N. Oza (Eds.), *Proceedings of the 1st InternationalConference on Lean Enterprise Software and Systems (LESS 2010)* (pp. 72–84), Helsinki, Finland, October 17–20.

Mehri, D., 2006. "The darker side of lean: an insider's perspective on the realities of the Toyota production system". Acad. Manag. Perspect. No. 20, pp.21–42.

Mehta, M., Anderson, D. and Raffo, D. (2008), “Providing value to customers in software development through lean principles”, Software Process: Improvement and Practice, Vol. 13No. 1, pp. 101-109.

Melton, T. (2005), “The benefits of lean manufacturing: what lean thinking has to offer the process industries”, Chemical Engineering Research and Design, Vol. 83, pp. 662-673.

Middleton, P. (2001), “Lean software development: two case studies”, Software Quality Journal, Vol. 9 No. 4, pp. 241-252.

Middleton, P. and Joyce, D. (2010), “Lean software development: BBC worldwide case study”, In Lean Software and Systems Conference, Atlanta, pp. 24-44.

Middleton, P. and Joyce, D. (2012), “Lean software management: BBC worldwide case study”, IEEE Transactions on Engineering Management, Vol. 59 No. 1.

Middleton, P., Flaxel, A. and Cookson, A. (2005), “Lean software management case study: timberline inc”, International Conference on Extreme Programming and Agile Processes in Software Engineering, Springer, Berlin, Heidelberg, pp. 1-9.

Mujtaba, S., Feldt, R. and Petersen, K. (2010),“Waste and lead time reduction in a software product customization process with value stream maps”, in 21st Australian Software Engineering Conference (ASWEC), IEEE Computer Society.

Musat, D. and Rodríguez, P. (2010), “Value stream mapping integration in software product lines”, In Proceedings of the 11th International Conference on Product Focused Software, ACM, pp. 110-111.

Nepal, B.P., Yadav, O.P. and Solanki, R. (2011), “Improving the NPD process by applying lean principles: a case study”, Engineering Management Journal, Vol. 23 No. 3, pp. 65-81.

Nord, R.L., Ozkaya, I. and Sangwan, R.S. (2012),“Making architecture visible to improve flow management in lean software development”, IEEE Software, Vol. 29 No. 5, pp. 33-39.

Nunnally, J. C. (1994). Bernstein. IH (1994). *Psychometric theory, 3.*

Ohno, T. 1988. Toyota Production System – Beyond Large Scale Production. Cambridge, MA: Productivity Press.

Oliveira, C. S. D., and E. B. Pinto. 2008. "Lean ManufacturingParadigm in the Foundry Industry." Estudos Tecnológicos Vol.4 no.3 pp. 218–230.

Paez, O., Salem, S., Solomon, J. and Genaidy, A. (2005), "Moving from lean manufacturing to lean construction: toward a common sociotechnological framework", *Human Factors and Ergonomics in Manufacturing*, Vol. 15 No. 2, pp. 233-245.

Panwar, A., Jain, R., Rathore, A.P.S., Nepal, B. and Lyons, A.C. (2018),"The impact of lean practices on operational performance – an empirical investigation of Indian process industries", Production Planning and Control, Vol. 29 No. 2, pp. 158-169.

Panwar, A., Nepal, B.P., Jain, R. and Rathore, A.P.S. (2015),"On the adoption of lean manufacturing principles in process industries", Production Planning and Control, Vol.26No. 7, pp. 564-587.

Pedersen, E.R.G. and Huniche, M. (2011a), "Determinants of lean success and failure in the Danish public sector: a negotiated order perspective", *International Journal of Public Sector Management*, Vol. 24 No. 5, pp. 403-420.

Perez, C., de Castro, R., Simons, D., Gimenez, G., 2010. "Development of lean supply chains: a case study of the Catalan pork sector". Supply Chain Manag. Int. J. 15, pp.55–68.

Pernstål, J., Feldt, R. and Gorschek, T. (2013), "The lean gap: a review of lean approaches to large-scale software systems development", Journal of Systems and Software, Vol. 86 No. 11, pp. 2797-2821.

Petersen, K. (2012), "A palette of lean indicators to detect waste in software maintenance: a case study", International Conference on Agile Software Development, Springer, Berlin, Heidelberg pp. 108-122

Petersen, K. and Wohlin, C. (2011), "Measuring the flow in lean software development", Journal of Software: Practice and Experience, Vol. 41 No. 9, pp. 975-996.

Petersen, K., and Wohlin, C. (2010),"Software process improvement through the lean measurement (SPI-LEAM) method", Journal of Systems and Software, Vol. 83 No. 7, pp. 1275-1287.

Pettersen, J. (2009), "Defining lean production: some conceptual and practical issues", The TQM Journal, Vol. 21 No. 2, pp. 127-142.

Piercy, N. and Rich, N. (2009a), "High quality and low cost: the lean service centre", *European Journal of Marketing*, Vol. 43 Nos 11/12, pp. 1477-1497.

Piercy, N. and Rich, N. (2009b), "Lean transformation in the pure service environment: the case of the call service centre", *International Journal of Operations and Production Management*, Vol. 29 No. 1, pp. 54-76.

Poppendieck, M. and Cusumano, M.A. (2012), "Lean software development: atutorial",IEEE Software, Vol. 29, pp. 26-32.

Poppendieck, M. and Poppendieck, T. (2005), "Introduction to lean software development", in: Extreme Programming and Agile Processes in Software Engineering, Springer, pp. 280-280.

Poppendieck, M. and Poppendieck, T. (2006), Implementing lean software development: from concept to cash, Pearson Education.

Poppendieck, M., & Poppendieck, T. (2007), Implementing lean software development: From concept to cash, Pearson Education.

Poppendieck, M., and Poppendieck, T. (2003). *Lean software development: An agile toolkit*. Boston: Addison-Wesley.

Psomas, E., Antony, J. and Bouranta, N. (2018), "Assessing lean adoption in food SMEs: evidence from Greece", International Journal of Quality and Reliability Management, Vol. 35 No. 1, pp. 64-81.

Psomas, E., Antony, J. and Bouranta, N. (2018), "Assessing lean adoption in food SMEs: evidence from Greece", International Journal of Quality and Reliability Management, Vol. 35 No. 1, pp. 64-81.

Reinertsen, D., (2005), "Let it flow", Ind. Eng, Vol. 37, pp. 40-45.

Rodríguez, P., Mäntylä, M., Oivo, M., Lwakatare, L.E., Seppänen, P. and Kuvaja, P., (2019), "Advances in using agile and lean processes for software development".,In *Advances in Computers* (Vol. 113, pp. 135-224). Elsevier.

Rodríguez, P., Mikkonen, K., Kuvaja, P., Oivo, M. and Garbajosa, J. (2013), “Building lean thinking in a telecom software development organization: strengths and challenges”, In Proceedings of the 2013 International Conference on Software and System Process, ACM, (pp. 98-107).

S.C.L. Koh, A. Gunasekaran, T. Goodman."Drivers, barriers and critical success factorsfor ERPII implementation in supply chains: Acritical analysis" The Journal of Strategic Information Systems, 2011

Saieg, P., Sotelino, E.D., Nascimento, D. and Caiado, R.G.G. (2018), “Interactions of building information modelling, lean and sustainability on the architectural, engineering and construction industry: a systematic review”, Journal of Cleaner Production, Vol. 174, pp. 788-806.

Samanta, U. and Mani, V.S. (2015),“Successfully transforming to lean by changing the mindset in a global product development team”, In Global Software Engineering (ICGSE), IEEE 10th International Conference on, pp. 135-139.

Sanjay Bhasin. "Lean Management Beyond Manufacturing", Springer Nature, 2015

Satolo, E.G., Hiraga, L.E.S., Goes, G.A. and Lourenzani, W.L. (2017), “Lean production in agribusiness organizations: multiple case studies in a developing country”, International Journal of Lean Six Sigma, Vol. 8 No. 3, pp. 335-358.

Saurin, T.A., Marodin, G.A. and Ribeiro, J.L.D. (2011),“A framework for assessing the use of lean production practices in manufacturing cells”, International Journal of Production Research, Vol. 49 No. 11, pp. 3211-3230.

Scherrer-Rathje, M., Todd, A.B. and Patricia, D. (2009), “Lean, take two! reflections from the second attempt at lean implementation”, *Business Horizons,* Vol. 52 No. 1, pp. 79-88.

Shah, R. and Ward, P.T. (2003), “Lean manufacturing: context, practice bundles, and performance”, Journal of Operations Management, Vol. 21 No. 2, pp. 129-149.

Shah, R. and Ward, P.T. (2007),“Defining and developing measures of lean production”, Journal of Operations Management, Vol. 25 No. 4, pp. 785-805.

Shapiro, R. (2014), “The US software industry as an engine for economic growth and employment”, University McDonough School of Business Research Paper, Georgetown.

Shingo, S. (1989), *A Study of Toyota Production System from an Industrial Engineering Viewpoint*, Japan Management Association, Tokyo.

Singh, I. and Kaur, N. (2017), “Contribution of information technology in growth of Indian economy”, International Journal of Research Granthaalayah, Vol. 5 No. 6, pp. 2394-3629, available at: https://doi.org/10.5281/zenodo.805859

Singh, R.K., Choudhury, A.K., Tiwari, M.K. and Maull, R.S. (2006), “An integrated fuzzy-based decision support system for the selection of lean tools: a case study from the steel industry”, Proc. Inst. Mech. Eng. Part B J. Eng. Manuf, Vol. 220 No. 10, pp. 1735-1749.

Staats, B.R., Brunner, D.J. and Upton, D.M. (2011), “Lean principles, learning, and knowledge work: evidence from a software services provider”, Journal of Operations Management, Vol. 29 No. 5, pp. 376-390.

Stone, K.B. (2012), “Four decades of lean: a systematic literature review”, International Journal of Lean Six Sigma, Vol. 3 No. 2, pp. 112-132.

Storch, R.L. and Lim, S. (1999), “Improving flow to achieve lean manufacturing in shipbuilding”, Production Planning and Control, Vol. 10 No. 2, pp. 127-137.

Tracy, R. (2007), “Driving lean through your supply chain: the cultivation of a more efficient supplier network”, White Paper, Intek Plastics. available at:www.werc.org/assets/1/workflow_staging/Publications/726.PDF (accessed 24 August 2013)

Ullman, J. B., and Bentler, P. M. (2003), Structural equation modelling. *John Willey and Sons. Inc.*.

Vinodh, S. and Joy, D., (2012),“Structural equation modelling of lean manufacturing practices”, *International Journal of Production Research, 50*(6), pp.1598-1607.

Voss, C. (2001),“Case research in operations management”, *Researching operations management*, pp.176-209.

Widman, J., Hua, S.Y. and Ross, S.C. (2010), “Applying lean principles in software developmentprocess – a case study”, Issues in Information Systems, Vol. 9 No. 1, pp. 635-639.

Wiengarten,F., Gimenez C., Fynes B., Ferdows K.(2015), "Exploring the importance of cultural collectivism on the efficacy of lean practices", International Journal of Operations & Production Management

Winter, M., & Szczepanek, T. (2008), “Projects and programmes as value creation processes: A new perspective and some practical implications”. *International Journal of Project Management*, *26*(1), 95-103.

Winter, M., (2006), "Directions for future research in project management: The main findings of a UK government-funded research network", International Journal of Project Management.

Womack, J.P. and Jones, D.T. (1994),“From lean production to the lean enterprise”, Harvard Business Review, Vol. 72 No. 2, pp. 93-103.

Womack, J.P. and Jones, D.T. (2003), *Lean Thinking*, Free press, New York, NY.Womack, J.P. and Jones, D.T. (1996), Lean Thinking, Banish Waste and Create Wealth in Your Corporation, Simon and Schuster, New York, NY.

Womack, J.P., Jones, D.T. and Roos, D. (1990), The Machine That Changed the World, Rawson Associates, New York, NY.

Wong, Y.C., Wong, K.Y. and Ali, A. (2009), “A study on lean manufacturing implementation in the Malaysian electrical and electronics industry” , *European Journal of Scientific Research*, Vol. 38 No. 4, pp.521-535.

Yadav, O.P., Nepal, B., Goel, P.S., Jain, R. and Mohanty, R.P., (2010), “Insights and learnings from lean manufacturing implementation practices”, *International Journal of Services and Operations Management*, *6*(4), pp.398-422.

Yadav, O.P., Nepal, B.P. and Jain, R., (2007), “Managing product development process complexity and challenges: a state-of-the art review”. *Journal of design research*, Vol. *6* No. 4, pp.487-508.

Yadav, R., Mittal, M.L. and Jain, R., 2018,“Adoption of lean principles in software development projects”. *International Journal of Lean Six Sigma*, https://doi.org/10.1108/ IJLSS-03-2018-0031

Yadav, V., Jain, R., Mittal, M.L., Panwar, A. and Lyons, A., (2019), “The impact of lean practices on the operational performance of SMEs in India”, *Industrial Management & Data Systems*, *119*(2), pp.317-330.

Yin, R. (2003), Case Study Research: Design and Methods, 3rd ed., Sage Publications, Thousand Oaks, CA

Ziskovsky, B. and Ziskovsky, J. (2007), “Doing more with less-Going lean in education”, available at: www.leaneducation.com/resources.html (accessed 24 August 2013).

APPENDICES

Appendix I

Questionnaire: Case Studies

Q. No.	Description of SD project/SD Organisation
1.	What are the different stages of software development?
2.	What percentage of total time is taken by each stage?
3.	What is generally the approximate duration of prototype programme of projects?
4.	When do you freeze your software development process (if dynamicity is present)?
5.	What problems are generally encountered in each stage of a project and how do you solve them?
6.	How do you handle changes in customer requirements?
7.	During which stages of the development process do the customers participate?
8.	What measures do you take to reduce cost of development?
9.	What measures do you take to reduce development time?
10.	What measures do you take for process improvement?
11.	What are different types of non-value activities (waste) in projects?
12.	How do you identify non-value activities in projects?
13.	How do you reduce waste in your SD projects?
14.	How do you prioritize between adding new technical features and removing defects (debugging)?
15.	How do you assess and create value in your SD product/project?
16.	How do you facilitate uninterrupted flow of information?
17.	How do you achieve flow in your SD organization?
18.	How is work pulled in your SD organization?
19.	How do you prioritize cost, delivery and quality of projects?
20.	In which stages of the projects is lean being implemented?
21.	How long has your firm been involved in implementing lean practices?
22.	Why do you want to apply lean principles in projects?
23.	What are the issues related to customer, process and people in adoption of lean principles in SD organization?
24.	Which lean practices do you use for software Development? **(Please fill up Table 1 as the answer to this query)**

Table 1: Lean Practices

S. No.	Lean practices used	1	2	3	4	5	Remarks
1.	Value Stream Mapping (VSM)						
2.	KANBAN/KANBAN BOARD						
3.	Visual Control						
4.	Standardized Work						
5.	Continuous Improvement						
6.	Visual Management						
7.	Flexible and cross functional teams						
8.	Team Based Problem solving						
9.	Set base Concurrent Development						
10.	JIT Information						
11.	Visual Management						
12.	Knowledge Innovation Visible Planning (KIVP)						
13.	Visual control Board						
14.	Design Structure Matrix						
15.	Cause and Effect Matrix						
16.	Obeya						
17.	Reverse Phase Scheduling						
18.	Heinjunka (Work load levelling)						
19.	SPC Techniques						
20.	Pull and Sequencing						
21.	Collocation (Concurrent and integrated product development practices)						
22.	Mistake-Proof-Processes						

Appendix-II

Lean Assessment Questionnaire

Position: Senior Management/Business Leader/Project Leads/Project Manager/Account Manager/ Team Leads/Team Member/Product Owner/B.A/analyst/designer/developer/tester……….…………………………………………………

Company's name & address: Cognizant Technology solutions Pune…………………..

Group I: General information

Q1. How do you consider your organization?

- Project based SD (Software Development) organization Yes [ ] No [ ]
- Product based SD organization Yes [ ] No [ ]
- Both product and project type organization Yes [ ] No [ ]

Q2. Kindly specify the type of your company according to the project/product:

i) Software services

E-commerce [ ] Web services [ ] Consulting [ ] other ………

ii) Domain Specific

Banking Software Solution [ ] Insurance [ ] Enterprise Application [ ] Other ……………

iii) IT support Services

Knowledge Management [ ] Information Security [ ] Business Processes [ ] Other ….

iv) Other than above ……..

Q3.What is the deliverable of your project?

Complete, Stand-alone software product/system [ ] Partial software product/system [ ] other ….

Q4.What is the annual turnover (in rupees) of your organization?

(i) 5-10 million [ ] (ii) 10 million- 1 billion [ ] (iii) Over 1 billion [ ]

Q5. How many people does your organization employ?

< 10 [ ] 10-50 [ ] 51-250 [ ] 251-500 [ ] >500 [ ]

Group II: Awareness about lean in SD Projects

LEAN ADOPTION STATUS

Q6. Kindly specify: Indicate your preferences

Please indicate the extent of implementation of Lean practices (Consider 1= low implementation, 7= complete implementation)

Item	Description	1	2	3	4	5	6	7
VL1	We use standardized tasks/parts clearly defining substance, order, timing, and desired result and processes for continuous improvement.							
VL2	Whenever defects are identified, it is logged and immediate actions are taken for fixing the problem and to obtain the desired quality on the first try/pass.							
VL3	We update, prioritize the backlog and incorporate new features using non-functional prototypes							
VL4	We frequently remain in close contact with our customer to know about their exact requirements							
VL5	Our customers give feedback on quality and delivery performance of the software							
VS1	We determine those activities that do not contribute value to the product or service and eliminate them							
VS2	We frame current and future development process diagram capturing processing time, delay time (DT), number of people involved							
VS3	We identify all the sources of waste in SD such as delays, unnecessary rework due to errors, defects, extra features and partial work done and track them.							
VS4	We analyze and design the workflow required to bring a software to a customer							
FL1	We have developed a system for planning to handle dependencies among tasks and ordering them							
FL2	We consider sets of conceptual solutions in parallel and narrow to develop single focused idea for achieving the final specifications							
FL3	We have developed direct pathway and connection among people and tasks in order to have smooth and timely flow of information							

FL4	Workload is leveled and balanced according to available resource to meet sprint demand							
FL5	Team members collaborate with each other for achieving goal							
FL6	Process parameters are displayed on workplace on visual dashboards/electronic dashboards to improve the process and progress of work							
PL1	Development of software at workplace is pulled by the demand of the next stakeholder in the organization							
PL2	We use signal cards which contains specification of job, detailed designs, verifications to do next task within stipulated time							
PL3	Work is assigned according to available capacity in my area of operations							
PL4	Team members pull the item/task from the backlog according to top priority							
PF1	Information for the execution of my work is displayed visually at my workstation and visible for every team member							
PF2	We capture online/documented feedback from customers							
PF3	Our organization adopt continuous improvement and continuous learning techniques inside and outside							
PF4	Employees are actively involved in problem solving in Product and process improvement							
RS1	We are highly capable of responding to customized requirements to handle dynamicity							
RS2	We use practices such as iterative, standardized error code, continuous integration, test driven development, and daily stand up meetings for expediting the SD project							
RS3	We use scrum in SD project for flow of story							
RS4	We use periodic builds and periodic code reviews , TFS, Docker and Base Camp and Jira							

Group III

Q7. The reason for adopting waste reduction methods and process improvement strategies –

Kindly specify the reasons for implementing in your software project: (1= not important, 7= highly important)

S. No.	Reasons of implementing lean	1	2	3	4	5	6	7
1.	To eliminate waste (rework/delays/ WIP inventory)							
2	To minimize project cost							
3	To increase customer satisfaction							
4	To build quality into the product							
5.	Rapid delivery of software with coordination and communication of team							
6.	To get solution of problem of customers							
7.	Other							

Group IV

Q8. What are the major challenges you have faced while implementing waste reduction methods and process improvement strategies:

(1= Least important; 7= Most important)

	Challenges: Barriers	1	2	3	4	5	6	7
CH1	Uneven requirement flow and change requests.							
CH2	Lack of techniques to recognize waste in SD Projects.							
CH3	Lack of information sharing or communication between organization and customers							
CH4	Lack of lean implementation experts with good leadership and coaching skills							
CH5	Lack of top management commitment							
CH6	Lack of communication and coordination within the team and among cross functional team							
CH7	Difficult to create a cross functional focus.							

CH8	Lack of coordination for parallel and sequential tasks.							
CH9	Lack of prioritization of bug fixing and adding new features							
CH10	Company policy and structure creates obstruction							
CH11.	Lack of consultants and trainers in the company and outside							
CH12.	Lack of proper requirement handover /We have unnecessary handovers							
CH13.	Lack of transparency							
CH14.	We face functional silos and Employees' resistance inside the organization							
CH15.	Lack of availability of required and allocation of resources							

Group V

Q9. Kindly specify the following operational characteristics in your organization/project

(1= strongly disagree, 7= strongly agree)

	Operational characteristics SD project	1	2	3	4	5	6	7
	Operational performance							
OP1	We generally complete the project within its estimated cost.							
OP2	Generally, we supply product/deliverables on the agreed upon time.							
OP3	We have fewer complaints from customers regarding desired quality and service.							
OP4	We have less work in progress (WIP) when a set of features are designed, coded and tested. (WIP:Requirements, specification, design documents, code fragments & partially done work)							

Appendix-III

List of Publications

International Journal Publications

Papers published in International Journal and proceedings in International Conferences

- Yadav, R., Mittal, M.L., & Jain, R. (2018). Adoption of lean principles in Software development projects. *International Journal of Lean Six Sigma*, DOI 10.1108/IJLSS-03-2018-0031
- Yadav, R., Mittal, M.L., Jain, R. & Yadav, V. (2019). Empirical analysis of lean adoption in Software Development Projects. *International Journal of enterprises information Management* (Communicated)
- Yadav, R., Mittal M.L, Jain R. Yadav, V. (2019). Drivers and Barriers of lean implementation in Software development projects. (To be communicated)

International/ National Conference

- Yadav, R., Mittal, M. L., and Jain, R. (2019, September). Lean practices in software development projects: A literature review. In *AIP Conference Proceedings* (Vol. 2148, No. 1, p. 030044). AIP Publishing., International conference on advances of mechanical engineering and Nano Technology (ICAMEN,8-9 March 2019) organized by Manipal University, Jaipur and University Malasia PAHANG (Scopus Indexed)
- Yadav, R., Mittal M.L, and Jain, R. (2018). Adoption of Lean Principles in New Product Development Projects: Case Study in Bearing Organization. Considered in an editor book chapter to be published by Elsevier. Role of Industrial Engineering in Industry 4.0 Paradigm (ICIEIND,27-30 September, 2018) Bhubaneswar, Odisha (Scopus Indexed)
- Yadav, R., & Mittal M.L. (2016). Presented on Adoption of lean Practices in New Product Development projects, All India Conference on Emerging trends in Automobile Engineering eld under the Institution of Engineers (India) and organizes by local chapter of *I E I, Alwar*
- Yadav, R., Mittal, M.L, Jain R. (2013). Presented on "Lean Project Management and Industrial Prospectus", during the National Conference on Manufacturing And Logistics Management (NCMLM 2013), held on March 8-9,2013 at MNIT Jaipur

Appendix-IV

Author's Biographical Profile (As on December, 2019)

Name: RamKaran Yadav

Date of Birth: 15 August 1970 (Born in Kotputli, Jaipur, India)

Address: Department of Mechanical Engineering
Govt. Ram Chandra Polytechnic College, Jaipur
Email: ramkaranyadav102@gmail.com
Contact No. : 9414422074

Educational Qualifications:

B.E: Mechanical Engineering (1994)
M.B.M. Engineering College, J.N.V. University, Jodhpur, Rajasthan (India)
M.E: Manufacturing Technology (2010)
NITTTR, Chandigarh (India).

Experience: Assistant Director/ Lecturer selection grade (Mechanical Department)
Government Ram Chandra Khaitan Polytechnic College, Jaipur, India

Area of Interest: Lean Manufacturing
Lean Software Development
Lean New Product Development Projects
Lean Thinking

Association with Professional Bodies:

Life membership of ISTE, IEI, and IIIE
Secretary of ISTE faculty chapter
(Govt. R.C Khaitan Polytechnic College, Jaipur)