CHAPTER – 1
INTRODUCTION
CHAPTER - 1

INTRODUCTION

1.1 Research Topic

Starting in the late 1990s massive change has occurred in all walks of life as a result of information technology revolution. The business environment has irrevocably impacted by the advent of personal computing technologies, coupled with high speed networking systems, Internet infrastructure, the Web, and sophisticated software tools for inexpensive information processing and transmission capabilities (Hsieh et al., 2007). Accurate information characterized by accessibility and speed have became the norm. The internet and the Web have become a common and very economical way to carry out business transactions. There is a fundamental shift in the business world. Driven by this revolution in the business environments brought about by information technology, a global economy has emerged (Bowersox et al., 2008). In today’s global business environment and economy, software applications and systems have become the driving force of modern business operations (Jones, 2006). These changes in business environment have matured to the point where the demand for more complex, integrated, robust, and resilient software applications and systems is placing enormous demands on more effective and efficient software development processes. Thus, software industry has emerged as one of the most strategic industry in today’s global economy (Albanna and Osterhaus, 1998).

The rapid globalization of software industry, with constant technological advances, competition gets more intense day-by-day. Customers are demanding higher quality, lower cost and shorter lead-time for software services and products. Software companies are facing tremendous pressure to improve their business and stay competitive in the global market by continuous process improvement and cost reduction (Pillai et al., 2003).

In recent years, Lean and Six Sigma approaches for continuous improvement have attracted the software industry. In today’s recessions and difficult economic times, scientists, researchers, and process experts believe that integrating Lean method and Six Sigma approach can bring large benefits to software organizations by uncovering
process waste, reducing non-value-added activity, and increasing efficiency and effectiveness of the software development processes. Furthermore, few software companies have already started experimenting on Lean Sigma approach with their regular project management practices. Lean Sigma evolved from experience in manufacturing industry where the process has inherent visibility. By contrast, software development is creative and intellectual process that must be made visible to measure and manage it (Card, 2000). Therefore, there are some issues and problems needs to be addressed, and some misperception needs to be cleared about the applicability of Lean Sigma approach to improve the capability of software development process. Lean Sigma to software development is in its nascent stage.

The main objective of this research work is to explore the advantages, opportunities, challenges, knowledge and attitude in the software community toward Lean Sigma approach to improve the capability of software development process. A conceptual model/framework is proposed to initiate Lean Sigma approach to improve the software development process continually that helps achieve competitive advantage in the global software business.

1.2 Landscape of Indian Software Industry

The IT boom in India started around the late 1990s. Today the IT industry is one of the most dynamic and vibrant sectors in India’s economy. Its contribution to the country’s Gross Domestic Product (GDP) is about 8% for the year 2013. Today, Indian IT industry directly employs over 3.1 million professionals. The industry indirectly employs another 10 million people. The industry today is the largest private sector employer in India. According to the industry body, the National Association of Software and Services Companies (NASSCOM), this sector is estimated to generate revenues of USD 225 billion, provide direct employment to 10 million people and indirect employment to 20 million people by the year 2020.

Indian IT industry comprises domestic firms, and captive centers of multinationals providing IT services, business process management (BPM), engineering and research and development (ER&D), and software products. Further, the industry is expanding its offering to emerging opportunities in social media, mobility, analytics, and cloud (SMAC), and new verticals like healthcare and medical devices.
Despite global economic uncertainty, Indian IT industry sustained growth trajectory for the year 2013. Industry export reached USD 76 billion with a year-on-year growth rate of 10.2 per cent. Domestic market also witnessed year-on-year growth rate of 14.1 per cent taking the domestic revenue to INR 1,047 billion.

For the financial year 2014, the NASSCOM is expecting the industry to clock export revenues to USD 84-87 billion maintaining a growth rate of 12-14 per cent. Domestic revenues are expected to reach INR 1,180-1,200 billion with the growth rate of 13-15 per cent.

1.2.1 Indian IT Industry Value Proposition

India is the global hub for the sourcing industry with presence in over 78 countries, and 600+ offshore delivery centers. It has 25 years of outsourcing experience in developing best practices and helping customers adopt new technologies to improve their business and operational performance. According to NASSCOM, key to India’s dominance in global sourcing is the fact that the industry is constantly expanding its service offerings and adding capabilities, evolving its business models, exerting tight control over operational parameters to ensure high customer satisfaction and high stockholder returns. Today Indian IT industry has moved from just a pure technology service provider to a strategic business partner.

Indian IT industry comprises over 16,000 firms, including 3,000+ software product firms, with the ability to provide coverage of all technology services. This unique diversity gives ample opportunities for clients to choose what size they want to be, what model they want to adopt, and what partnerships to create. This reinforces India’s unique position as the only country from where you can do everything (NASSCOM Strategic Review, 2014).

Furthermore, India is home to a new breed of startup firms focused on high growth areas such as e-commerce and SMAC (social media, mobility, analytics, and cloud), and vertical-specific solutions – these firms are creating new markets and driving innovation.

According to NASSCOM, India continues to lead in cost competitiveness. Flat entry level salaries, flattening employee pyramid, Tier II/III service delivery, alternate talent
pool hiring and fast career growth helping India stay 7-8 times cheaper than source locations and 30 per cent cheaper than the next nearest low-cost country.

1.2.2 Bangalore – The Silicon Valley of India

Bangalore is where the Indian IT industry began its dramatic success story, often referred as the Silicon Valley of India. It is now home to some of the best international and domestic software companies. Nearly 40% of the country’s IT industry is concentrated in Bangalore (BBC News, 2013). Bangalore has been the hub for software and services in India, with 950 product firms (out of an overall count of 3000 in India). It has been ranked fourth as a global hub of technological innovation, behind San Francisco and Austin of the US, and Taipei in Taiwan (VERGE – A NASSCOM Regional Newsletter – South, 2012).

1.2.3 Prospect of Mysore for IT Industry

Mysore is only 150 kms from Bangalore with well connected transportation system. Infosys has already set up a Leadership Training Institute in Mysore. The Government of Karnataka has decided to launch a program to promote Mysore as next leading destination for IT sector. It has joined hands with NASSCOM, forming a core team that will focus on promoting Mysore as an IT hub. Guided by IT secretary, the core team will set up a task force to take the IT sector in Mysore to its next phase of growth and evolution. As a part of this agenda, the Karnataka Government successfully rolled out Mysore 2.0 Conference. N. R. Narayana Murthy was the key speaker for this event attended by over 300 delegates.

1.3 Overview of Current Software Development Models

Software development can be broadly categorized into custom developed software and generic software products. Customized software development involves close interaction between the development team and the end-user. Software product may be targeted to a vertical segment or many cut across segments.

In software industry, currently, two models dominating the software development activities – waterfall model and agile/scrum model. Waterfall model is sequential development model. It is the predominant model in the software development community. It is there since 1970s. Agile/scrum model is iterative development model.
And, it is direct response to traditional waterfall model. According to a 2010 survey by Forrester, only about 35% of software companies surveyed states that their development process is based on agile methods. Therefore, waterfall model is still a dominant software development model in the industry today.

1.3.1 Waterfall Model – Sequential Approach

The sequential approach to software development process is termed as waterfall model, after a model described by Dr. Winston W. Royce in the mid 1970s. In waterfall model the work progresses in sequential phases. These phases typically include requirement analysis, high level design, low level design, coding and unit testing, integration testing, system testing, and release. The documentation involved in this model is quite a lot, as documents developed in the previous phase needs to be signed off before proceeds to the next phase. A typical waterfall model is shown in Figure 1-1.

Figure 1-1. A typical waterfall model for software development process

![Waterfall Model Diagram](Image)

(Source: Adapted from Royce, 1970)

In this model, the project team do all the requirement analysis in one go, sign off, then all the design in one go, sign off, then all the coding in one go, sign off, and then finally, the complete system testing in one go. Then release. The major problem with the waterfall model is its inflexibility. It is especially ineffective and inefficient in responding to changing customer requirements. In water fall model, testing normally takes place only after coding is complete. Unfortunately, most of the time the development work continues throughout the cycle while testing starts only late and
never seems to get enough time to complete all testing activities. According to Royce, this model is risky and invites failure.

The waterfall model rarely works because it is predicted on stable requirements. In today’s marketplace, the customer requirements and priorities are rapidly changing. This approach takes several months to years to deliver a project.

In order to address some of the issues in waterfall model, the software industry adopted V-Model. This model is an extension of waterfall model and is based on association of a testing phase for each corresponding development stage. This means that for every single phase in the development cycle there is a directly associated testing phase. On the left side, development activities including requirement analysis, feature specifications, high-level design, and low-level design proceed from top to bottom. On the right side, testing activities, including, unit testing, integration testing, system testing, and acceptance testing are completed in a bottom-up fashion. In this model, test plans are developed along with each development activities, and the tests will executed in reverse order after coding is completed. Coding phase joins the two sides of the V – Model. A typical V – model is shown in Figure 1-2. The advantage of this model is that the test plans can be completed earlier in the process, resulting in shortened overall development and testing time.

*Figure 1-2. A typical V – model for software development process*

(Source: Zhang et al., 2010)
1.3.2 Agile Model – Iterative Approach

Agile method is basically an iterative approach to software development. It represents a major departure from traditional waterfall model. Agile manifesto was written by experienced practitioners and was formalized in 2001. According to the agile manifesto, agile software development is based on four core values:

- **Individuals and interactions** over processes and tools
- **Working software** over comprehensive documentation
- **Customer collaboration** over contract negotiation
- **Responding to change** over following a plan

There are many agile software development methods. These include – Dynamic System Development Methods (DSDM), Extreme Programming, Crystal Clear, Feature Driven Development, Scrum, and others. Of all the agile methods, Scrum is very popular these days.

Scrum method was developed by Ken Schwaber and Jeff Sutherland in 1990s. Scrum is unique because it uses the real-time progress of a project, where feedback loops constitute the core element, to plan and deliver the application. In Scrum, unlike waterfall model, we use smaller batch sizes and short delivery cycles called *sprints*. These sprints are typically one to four weeks in duration. In each sprint, the project team members do a little bit of requirement analysis, a little bit of design, a little bit of development, and a little bit of testing in order to deliver a handful of prioritized features every few weeks. The team members coordinate their work in daily stand-up meetings, which normally lasts for about 15 minutes. At the end of each sprint, product owner and team members meet to assess the progress of a project and plan its next sprint release. This allows the project manager to adjust and revise the project plan based on completed work, not predictions. This results in enormous improvements in quality work, delivery time, and customer satisfaction. A typical scrum model is shown in Figure 1-3.

Though scrum is a better fit than waterfall model to software development, it has pitfalls when it comes to reality. The quality of requirement analysis suffers as team members rush to prepare for upcoming sprints, the quality of current development
activities suffer when busy team members are unable to respond to issues during the iteration, and finally the quality of the product itself suffers as the development work continuous throughout a iteration while testing starts late at the very end of the iteration and never seems to get enough time.

Figure 1-3. A typical scrum model for software development process

(Source: Adapted from Schwaber and Beedle, 2002)

1.4 The Evolution of Lean Sigma Approach

The Lean Sigma is a combination of the two most popular management practices to enhance competitiveness – Lean methods and Six Sigma approaches.

1.4.1 Lean

The term "Lean" was coined to describe Toyota's business model during the late 1980s by a research team headed by Jim Womack at MIT's International Motor Vehicle Program (IMVP). Lean is a business transformation methodology that derived from the Toyota Production System (TPS). Lean focus on increasing customer value by reducing the cycle time of product or service delivery through the elimination of all forms of muda (a Japanese term for waste) and mura (a Japanese term for unevenness in the workflow).

1.4.2 Six Sigma

Six Sigma was a concept developed in 1985 by Bill Smith of Motorola, who is known as “the Father of Six Sigma”. Six Sigma is a business transformation methodology that maximizes profits and delivers value to customers by focusing on the reduction of
variation and elimination of defects by using various statistical, data-based tools and techniques.

1.4.3 The Synergy of Lean and Six Sigma

When companies adopt either Lean method or Six Sigma approaches alone, they might reach a point of diminishing returns. Further improvements in their business processes are not easy. In this situation, Lean organizations can make use of statistical techniques for more scientific approach to quality. On the other hand, Six Sigma companies can eliminate all forms of waste to reduce lead time for their operations. When companies use both Lean and Six Sigma simultaneously, dramatic improvements across the organization can be achieved much more rapidly. The integration of Lean and Six Sigma improvement methods is required because Lean

<table>
<thead>
<tr>
<th>Process Improvement:</th>
<th>Lean</th>
<th>Six Sigma</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concept:</td>
<td>Eliminate wastes</td>
<td>Reduce variations</td>
</tr>
<tr>
<td>Implementation steps:</td>
<td>Create customer value</td>
<td>Define opportunity</td>
</tr>
<tr>
<td></td>
<td>Define value stream</td>
<td>Measure performance</td>
</tr>
<tr>
<td></td>
<td>Create flow</td>
<td>Analyze opportunity</td>
</tr>
<tr>
<td></td>
<td>Pull from customer</td>
<td>Implement improvement</td>
</tr>
<tr>
<td></td>
<td>Strive for excellence</td>
<td>Control performance</td>
</tr>
<tr>
<td>Focus:</td>
<td>Flow focused</td>
<td>Problem focused</td>
</tr>
<tr>
<td>Performance indicator:</td>
<td>Flow</td>
<td>Fluctuation</td>
</tr>
<tr>
<td>Primary effect:</td>
<td>Reduced flow time</td>
<td>Uniform process output</td>
</tr>
<tr>
<td>Secondary effects:</td>
<td>Less variation</td>
<td>Less waste</td>
</tr>
<tr>
<td></td>
<td>Uniform output</td>
<td>Fast throughput</td>
</tr>
<tr>
<td></td>
<td>Less inventory</td>
<td>Less inventory</td>
</tr>
<tr>
<td></td>
<td>Improved quality</td>
<td>Improved quality</td>
</tr>
<tr>
<td>Criticism:</td>
<td>Statistical analysis not considered</td>
<td>System interaction not considered (processes improved independently)</td>
</tr>
</tbody>
</table>

(Source: Adapted from Nave, 2002)
cannot bring a process under statistical control, and Six Sigma alone cannot dramatically improve process speed (George, 2002). Table 1-1 presents the comparison of Lean and Six Sigma.

Figure 1-4 shows how Lean Sigma combines the speed and efficiency of Lean method with the effectiveness of Six Sigma approach to achieve process excellence, customer satisfaction and growth.

*Figure 1-4. Synergy of Lean and Six Sigma*

1.4.4 Software Project Management and Lean Sigma Approach

A software project has two main activity dimensions – engineering and project management. The engineering dimension deals with building the system and focuses on design, code, test, and so on. The project management dimension deals with properly planning and controlling the engineering activities to meet projects goals and objectives for cost, schedule, and quality (Jalote, 2005). In contrast, Lean Sigma framework provides a structured data-driven methodology. It is designed to uncover the root cause of process-related problems. It is not just another project management initiative to replace the current practices. It is a complementary project management methodology that is integrated into the existing software project management practices and replaces the existing ways of determining, analyzing, and resolving process-related problems, as well as achieving project goals and customer requirements objectively and
methodically. Non-value-added activities are eliminated, data is collected; processes are measured, and analyzed. Problems are identified. Innovative solutions are designed and deployed. Lean Sigma is a strategy deployment framework that includes statistical methodologies.

Lean Sigma framework allows software professionals to take their projects to new levels of discipline. Lean Sigma framework perfectly gel with current software project management practices and leads to increased quality and productivity in software development process (Peterka, 2007).

1.5 Need for the Research

In today’s marketplace, the rapid globalization of software industry with constant technological advances, the competition gets more intense. Customers are demanding higher quality, lower cost, and shorter lead-time for software services and products. In this situation the traditional approach and mindset to software development and project management are not enough. Most effective and efficient processes are needed.

Research analysis shows that top manufacturing companies with successful track records have discovered one possible solution. Lean Sigma. Using Lean Sigma, they discovered opportunities, and produced breakthrough innovations that had profound impacts on their business processes. Therefore, scientists, researchers and process experts believe that Lean Sigma can bring similar benefits to software organizations also. Furthermore, few software companies have already started experimenting on Lean Sigma approach with their regular project management practices. But it is in its early stages.

Lean and Six Sigma have their roots in manufacturing. Compared to manufacturing process, software development is very different as it is not rolling out a physical product. Instead, it is creating an intellectual property, which is highly dependent on the developers’ creativity and efficiency. Moreover, software development process has more uncertainties and complexities than manufacturing process. Therefore, there is a need to explore opportunities and challenges in adopting Lean Sigma to improve the capability of software development process. The result will ultimately benefit the Software industry as a whole, which is one of the most dynamic and vibrant sectors in
India’s economy contributing significantly to the country’s Gross Domestic Product (GDP).

1.6 Problem Statement

The problem statement is:

“To explore the opportunities and challenges in implementing Lean Sigma to improve the software development process for three purposes: (1) Lean and Six Sigma has dramatically increased the efficiency of product development and quality of products in manufacturing industry. They are very promising for software development process. There are huge opportunities to contribute in this field; (2) Unlike manufacturing, the very nature of software development process is intangible and not repeatable, the challenges in implementing Lean Sigma needs to be explored and (3) As no framework available to implement Lean Sigma, there is a need to construct a conceptual model for implementing Lean Sigma to improve software development process.”

1.7 Research Objectives

The main aim of the research is to explore the various opportunities and challenges in implementing Lean Sigma to improve the software development process. In line with this, the following are the specific objectives of the research:

1. To explore the advantages of implementing Lean Sigma to software development process
2. To examine the knowledge of Lean and Six Sigma among software professionals
3. To examine the attitude of software professionals in adopting Lean Sigma to software development process
4. To investigate the opportunities for implementing Lean Sigma to software development process
5. To investigate the challenges in implementing Lean Sigma to software development process
6. To propose a conceptual model for implementing Lean Sigma to improve the software development process.
1.8 Research Hypotheses

The following null-hypotheses have been formulated in the present study:

To explore whether there are significant advantages in implementing Lean Sigma to improve the capability of the software development process in both service and product oriented software companies, the following null-hypotheses have been formulated:

\[ H_{0,1a} \]: There are no significant advantages in adopting Lean Sigma approach to improve the capability of the software development process.

\[ H_{0,1b} \]: There is no significant difference in advantages between service and product oriented software companies in adopting Lean Sigma to improve the capability of the software development process.

To examine whether there is significant level of knowledge in Lean and Six Sigma among software professionals to improve the software development process in both service and product oriented software companies, the following null-hypotheses have been formulated:

\[ H_{0,2a} \]: The knowledge level of Lean and Six Sigma among software professionals is not sufficient to improve the capability of the software development process.

\[ H_{0,2b} \]: There is no significant difference in the knowledge level among software professionals between service and product oriented software companies to improve the capability of the software development process.

To examine whether there is significant attitude among software professionals toward implementing Lean Sigma to the software development process in both service and product segments, the following null-hypotheses have been formulated:

\[ H_{0,3a} \]: There is no sufficient attitude among software professionals towards implementing Lean Sigma to improve the capability of the software development process.

\[ H_{0,3b} \]: There is no significant difference in attitude among software professionals between service and product oriented software companies towards implementing Lean Sigma to improve the capability of the software development process.
To investigate whether there are significant opportunities exist in implementing Lean Sigma framework to improve the capability of the software development process in both service and product oriented software companies, the following null-hypotheses have been formulated:

H<sub>0.4a</sub>: There are no significant opportunities for implementing Lean Sigma to improve the capability of the software development process.

H<sub>0.4b</sub>: There is no significant difference in opportunities between service and product oriented software companies in implementing Lean Sigma to improve the capability of the software development process.

To investigate whether there are significant challenges in implementing Lean Sigma framework to improve the capability of software development process in both service and product oriented software companies, the following null-hypotheses have been formulated:

H<sub>0.5a</sub>: There are no challenges in implementing Lean Sigma to improve the capability of the software development process.

H<sub>0.5b</sub>: There is no significant difference in challenges between service and product oriented software companies in implementing Lean Sigma framework to improve the capability of the software development process.

To validate the proposed conceptual model for Lean Sigma to improve the software development process for both service and product oriented software companies, the following null-hypotheses have been formulated:

H<sub>0.6a</sub>: The implementation of the proposed conceptual model for Lean Sigma could not improve the capability of the software development process.

H<sub>0.6b</sub>: There is no significant difference in opinion between service and product oriented software companies in the implementation of the proposed conceptual model for Lean Sigma to improve the capability of the software development process.
1.9 Research Methodology

This section presents the research design and procedures that have been followed in conducting this research work.

A research design is a framework or blue print for conducting the investigation to obtain empirical evidence to address the research problem. It details the procedure necessary for the collection, measurement, and analysis of data (Cooper et al., 2012). Research may be broadly classified, depending on the objectives of the study, as exploratory, descriptive, or causal (Zikmund, 2003).

Exploratory research is conducted to get more insight and better understanding of the dimensions of the research problem. It is particularly suitable in two circumstances – when the topic of research is in the early stages, and when the researcher intends to extend the applicability of a theory in a new context (Voss et al., 2002).

The main purpose of descriptive research is to describe the characteristics of a population or phenomenon. Descriptive design seeks to determine the answers to who, what, when, where, why, and how of the research topic (Zikmund, 2003).

Causal research is used to identify cause-and-effect relationships among variables. It attempts to establish that when we do one thing, there will be an effect on another thing (Cooper et al., 2012).

The main objective of this research is to investigate the opportunities and challenges in implementing Lean Sigma approach to improve the capability of software development process. This study will help in collective understanding on the effectiveness of Lean Sigma approach to improve the software development process and will ultimately benefit the software industry as a whole.

In order to achieve the research objectives, exploratory and descriptive research designs have been chosen. In this case, Lean and Six Sigma method have their roots in manufacturing industry and later became popular in the service sector. Researcher intends to extend this approach to improve the capability of software development process. The implementation of Lean Sigma to improve software development is in its nascent stage. Therefore, exploratory and descriptive research designs are suitable as the research topic is in the early stages, and the researcher intends to extend the
applicability of a theory from the manufacturing process to a new context – software development process.

**Selection of Respondents:**

Snowball sampling technique was adopted for the selection of respondents for three reasons:

1. Lean Sigma approach is in its early stage.
2. Very few companies have started experimenting with Lean Sigma approach to software development activities.
3. It is very difficult to identify software professionals with a blend of project management experience, knowledge on different maturity models, agile/scrum, Lean and Six Sigma, and had an opportunity to work on such strategically important projects from service and/or product oriented software companies.

Therefore, snowball sampling technique was found to be appropriate. In snowball sampling initial respondents were selected and after being taken input, these respondents were requested to identify and refer who belong to the target population of interest. Subsequent respondents were selected based on the referrals.

From the sample it was found that 23.5% of respondents are senior executives, 31.6% of them are project managers, 27.5% of the professionals are project/team leaders, and 17.3% of them are software engineers. Majority of the respondents have more than 10 to 15 years of experience in handling software development projects in India and abroad. 93.9% of the respondents expressed that they have knowledge on different maturity models for improving software development processes.

This indicates that the respondents had very good knowledge on the software development life cycle and the company’s processes and various process improvement programs to improve the capability of software development process. They were very familiar with different maturity models. 43.9% of the respondents had already handled agile models. Overall, the result of the demographic questionnaire showed good knowledge and awareness of the processes, metrics, and continuous improvement initiatives like CMM, Lean and Six Sigma, which is also evident from their responses and throughout the discussion on various topics of software development process.
Development of Questionnaire:

A questionnaire was used to elicit required information from the software professionals from service and product oriented software companies. The questionnaire was developed based on the literature review and thorough discussions held with software professionals involved with process improvement programs. The questionnaire contained six sections with different number of statements to extract the perception and understanding of the professionals towards the Lean Sigma approach to software development activities. The prepared questioner was reviewed by the practitioners from the industry and academician to assess the ability of the research instrument. Further, the Cronbach’s alpha has been used to estimate the reliability of the questionnaire. Cronbach’s alpha is commonly used for computing the reliability coefficient of a test. The different sections in the questionnaire, number of statements in each section, and the computed reliability coefficients are presented in the Table 1-2 below:

Table 1-2. Different sections in the questionnaire, number of statements, and the reliability coefficients

<table>
<thead>
<tr>
<th>No.</th>
<th>Different Sections in the Questionnaire</th>
<th>No. of Statements</th>
<th>Cronbach’s Alpha</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Advantages of implementing Lean Sigma to software development process</td>
<td>11</td>
<td>.984</td>
</tr>
<tr>
<td>2.</td>
<td>Knowledge of Lean and Six Sigma among software professionals</td>
<td>15</td>
<td>.869</td>
</tr>
<tr>
<td>3.</td>
<td>Attitude of software professionals towards implementing Lean Sigma to software development process</td>
<td>10</td>
<td>.876</td>
</tr>
<tr>
<td>4.</td>
<td>Opportunities in implementing Lean Sigma to software development process</td>
<td>12</td>
<td>.981</td>
</tr>
<tr>
<td>5.</td>
<td>Challenges in implementing Lean Sigma to software development process</td>
<td>15</td>
<td>.870</td>
</tr>
<tr>
<td>6.</td>
<td>A proposed conceptual model for implementing Lean Sigma to software development process</td>
<td>15</td>
<td>.796</td>
</tr>
<tr>
<td></td>
<td><strong>Total number of statements</strong></td>
<td><strong>78</strong></td>
<td></td>
</tr>
</tbody>
</table>


Data Collection:
This research is based on a survey addressed to software professionals from service and product segments who were involved with process improvement programs to their software development processes.

A questionnaire survey has been conducted among software professionals in software development centers in Bangalore. These software professionals are working in software companies that are member of the National Association of Software and Service Companies (NASSCOM). We also approached Project Management Institute (PMI, USA) Bangalore Chapter and appealed to the project management community to help networking the software professionals involved with Lean and Six Sigma background. A total of 150 questionnaires were sent to software professionals. 98 responses have been obtained from 38 software companies in India (Bangalore, Chennai, and Hyderabad) and their strategic business partners abroad (Singapore, Canada, USA, and UK).

The data collection process took around 6 months from July to December 2013. During this period, the researcher stayed in Bangalore. Visited leading software companies in Bangalore, and held thorough discussions with software professionals and process practitioners. These were supplemented by numerous phone calls, conference calls, text messages, and e-mails. All the data collected through the survey is consolidated and analyzed with the SPSS (Statistical Package for Social Sciences) tool.

Procedure:
- Studying methodologies of current software development processes
- Collecting observations from detailed literature review
- Developing a conceptual model for implementing Lean Sigma to improve the software development process
- Conducting opinion survey on advantages, opportunities, challenges, knowledge, attitude, and the practicality of proposed conceptual model for Lean Sigma through questionnaire
- Improving the model based on the contributions of the domain experts from the software community.
Statistical Methods:

The collected data was analyzed with following statistical methods:

- Descriptive statistics: The Descriptive statistics procedure usually displays univariate summary statistics for several variables in a single table and calculates standardized values. Descriptive statistics like frequencies, percentages, mean, S.D, and hierarchical ranking were utilized.

- One-sample t-test: One sample t-test is a statistical procedure for testing the mean value of a distribution to determine whether the population mean confirms to a given hypothesis.

- Independent samples t-test: The independent-samples t-test procedure compares means for two groups of cases. Independent samples t-tests were applied to verify the significance of difference between respondents in service and product oriented software companies.

- Chi-square test: The Chi-Square test procedure tabulates a variable into categories and computes a chi-square statistic. This goodness-of-fit test compares the observed and expected frequencies in each category to test either that all categories contain the same proportion of values or that each category contains a user-specified proportion of values. In the present study chi-square test was applied to find out the significance of difference between various groups of frequencies of advantages, opportunities, challenges, knowledge level and attitude – with 5 point Likert scale.

- Contingency table analysis or cross tabulation procedure: The Crosstabs procedure forms two-way and multi-way tables and provides a variety of tests and measures of association for two-way tables. Contingency table analysis or cross tabulation procedure was applied to find out the association between service and product segments.
1.10 Scope of the Research

Software development companies in India are still in primitive stage to adopt Lean Six Sigma practices to achieve higher quality, lower cost and shorter lead-time. There are many opportunities here to improve software development process for meeting global standards to satisfy various stakeholders.

This research is an initiative to adopt one of the best global practices for process improvement that is Lean Six Sigma approach which is found huge success in many other business sectors.

- The research will be conducted in software companies that are adopting or willing to adopt Lean and/or Six Sigma approaches to improve software development process.
- The research result will be based on the survey addressed to software professionals from reputed software companies located in Bangalore, India and its branches in other cities or countries (based on referrals).

1.11 Limitations of the Research

The researcher has identified some limitation to this study.

- Very difficult to identify the companies that are executing projects with Lean and Six Sigma approaches for process improvement.
- It is very difficult to get software professionals with a blend of project management experience, knowledge on different maturity models, agile/scrum, Lean and Six Sigma, and had an opportunity to work on such strategically important projects from service and/or product oriented software companies.
- The researcher had found difficulties in getting responses from some of the software professionals because they are very busy with current project, or they were on business travel.
- Some of the software professionals did not participate in the survey because of the non-disclosure-agreement (NDA) with the company.
1.12 Thesis Outline

This section describes the layout of the thesis.

CHAPTER – 1 provides an overview of the research topic. A brief landscape of the software industry in India and an overview of current software development models are discussed. The evolution of Lean Sigma approach is introduced. The research objectives and hypotheses are stated. The research methodology followed in this research is explained. Finally, the scope of the research and limitations are discussed.

In CHAPTER – 2, a systematic literature review was carried out on Lean and Six Sigma in the manufacturing sector followed by that in the service sector. Current research in implementing Lean and Six Sigma to software development process is summarized. Highlight the research gaps relating to the adoption of Lean Sigma approaches to software development is discussed. Finally, the literature review is used to formulate hypotheses to explore the research objectives stated in CHAPTER – 1.

CHAPTER – 3 introduces a conceptual model/framework for implementing Lean Sigma to improve the capability software development process.

In CHAPTER – 4, the data analysis from the survey responses is reported. A demographic profile of respondents is presented, followed by the analysis of the survey data and interpretations. Finally, the results of the hypotheses tests relating to the research objectives are presented.

CHAPTER – 5 provides the summary of the overall findings of the research work.

CHAPTER – 6 presents the conclusions on the research work. Suggestions are provided to initiate Lean Sigma to software development projects followed by the recommendations for future research work.

This is followed by appendices and bibliography.