Lean Software Development Practices and Principles in Terms of Observations and Evolution Methods to increase work environment productivity

Llahm Omar Faraj Ben Dalla
Software Engineering Department
Sabha Technical University
mohmdaesed@gmail.com

Abstract
Lean Software Development (LSD) is one of the influential Agile Software Development (ASD) methods. Furthermore, the main aim and objective of this essential method is creating customer value as well as swift delivery in time within the required budget. Moreover, lean methodology can enhance business domain via adopting the usage of lean principles (LPs) according to the business requirements in diverse domains. This observational paper provides observations on the evolution of lean software development practices as well as principles. This study is a significant in terms of three important contributions: the first stage of the contribution is defined as lean as well as lean principles in terms of powerful as well as weaknesses. In addition, the second contribution studied the relationship between ASD as well as LSD. Further, the study contributes the comprehensive understanding of LSD principles and practices during the recent decade. Additionally, the results of this beneficial study are important for several domains such as the industrial world, the educational world, the manufacturing world as well as the scientific world in addition to researchers who aimed for some investigations outcome based on LSD practices besides principles in manufacturing domain.

Keywords: *Lean Software Development (LSD); Lean Principles (LPs); Lean Practices; Software requirements; Software Development Life Cycle (SWDLC).*

I. Introduction

After the Second World War in the 1940s, Japanese industries began to introduce a new approach to software development methodology called “Lean software development” method (LSD). In the mid-1980s, the term "lean" was combined with the product management process and, after that, utilized in product development at MIT. Lean Methodology (LM) life cycle can be found in the book entitled "The machine that changed the world" (Yadav et al., 2020); (Kalaria et al., 2020). The primary goals of Lean are to reduce support terms, increase customer value and time-to-market (Laanti, 2016); (Llahm et al., 2017); (Gutbrod and Münch, 2018); (Yadav et al., 2020). Furthermore, lean software development is one of the agile methodologies (AMs) and, thus, depends on a flexible method to distinguish the utilize of LSD depending on the ventures and their quality while operating under an agile umbrella (Janes, 2015). The success of LSD is to have support from the top management level. Since this observational study could not find an article on the evolution of lean principles and practices and their benefits to lean product development (LPD). This paper will advance the knowledge in this direction. All stakeholders will also know how to determine the venture life cycle (related to sustainable development). The product should be built from high quality-component (HQC). The remainder of this paper is organized as follows. Section 1: Introduction, Section 2 presents the comprehensive review of LSD in last decade. Section 3 and 5 presents the discussion and limitations. Finally it concludes with future research direction. The benefits of a structured method according to (Saltz and Sutherland, 2020), is to build software with human effort, development hours and investment as compared to what CMMI level 3 organizations would achieve.

1.2. Aim and objective of the study

This study aimed for the following points

- To understand lean software development principles and practices which were utilized as well as why the utilizes of lean software development industrialized over time.
- To understand the usage according to differ period such as software manufacturing amongst modern backgrounds.
- To know the common practices and principles of lean software development and it is important in software industries.
- To observe some practices and principles of this methodology based on the objective effect of software development lifecycle in the production domain.

1.3.2. Change tolerance

When an organization could continue improving despite high uncertainties as well as fluctuations in the market, this property is referred as ‘change tolerance’ linked with the adaptation (Saltz and Sutherland, 2020). Lean software development incorporates the concept of dynamic stability with the ability to be adaptive to change once many customer requirements change to continuously improve the quality of internal processes. LSD is the process of creating change-tolerant software with the help of individuals to reduce development time, proper investment in tools as well as methodologies, which is linked with adapting to the new time-to-market settings. According to Alba-Baena et al., (2020), the main idea here is that potential competitive advantage comes from being more flexible than other competitors to gain access to the market in less time as well as with less efforts. However, unexpected circumstances in the new markets cannot be easily foreseen via many firms, leading to disturbances in the firm plan. Fast maintenance duration can also be another high-risk factor leading to failure of delivery (Alba-Baena et al., 2020).

1.3.3. Leadership barriers and risk

LSD provides on-time as well as within-budget delivery via implementing "risk entrepreneurship", which is a component of risk leadership/management. In traditional software development, the risk is resolved to avoid loss. In LSD, (Baumer-Cardoso et al., 2020)., change is viewed as an opportunity. Risk leadership involves risk management which is linked with entrepreneurship requires the ability to detect opportunities derived via cost (Alba-Baena et al., 2020). Baumer-Cardoso et al., (2020) has declared that via managing risk management to get opportunities from the risks, is called ‘risk entrepreneurship’.

1.3.4. Lean principles(LPs)

Lean software development is focused on eliminating waste as well as create customer value (Sutherland et al., 2020). There are 7 principles in lean software development (Sutherland et al., 2020).

- **LP1:** Eliminate waste (Sutherland et al., 2020); (Kalaria et al., 2020).
- **LP2:** Amplify learning related to apply training (Kalaria et al., 2020).
- **LP3:** Delay Commitment (Kalaria et al., 2020). Decide as late as possible derive the stakeholder to have a good decision.
- **LP4:** Empower the team at any organization (related to sponsorship) (Sutherland et al., 2020); (Kalaria et al., 2020).
- **LP5:** deliver fast (related to early and frequent Releases) (Chen, 2015).
- **LP6:** Build Integrity related to optimizing the whole (Kalaria et al., 2020).
- **LP7:** Optimize the whole (Kalaria et al., 2020).

1.3.5. The disadvantage of utilizing LSD

According to (Sutherland et al., 2020) ; (Secor et al., 2014) there are several disadvantages of LSD which presented in Table.A-1.

1.3.6. LSD practices

Kalaria et al., (2020) and several authors have introduced lean practices as tools as reported in Table A-2. Lean practices relevant to software development. The remainder of this paper is structured as follows: In section 2 relevant background research is described, while section 3 provides an evolution of LSD. In addition, section 4 refer to the evolution scenario of lean software development. Section 5 illustares discussion along with required tables. Section 6 is about study limitations as well as finally it concludes with summary linked with future research directions.

1.4. Lean team members characteristics

Lean team members "work cell" (Kalaria et al., 2020) have the ability to solve problems as well as find optimal solutions. Internal which is related to external expertise according to the contract reviews as presented in figure.1.3.below.

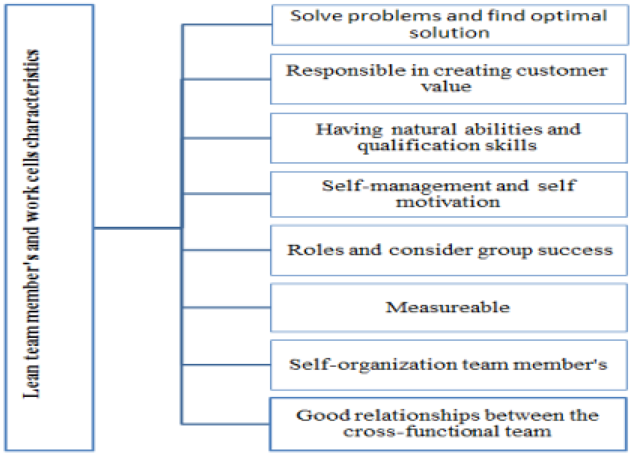
Lean team member's and work cells characteristics
Solve problems and find optimal solution
Responsible in creating customer value
Having natural abilities and qualification skills
Self-management and self motivation
Roles and consider group success
Measureable
Self-organization team member's
Good relationships between the cross-functional team

Figure 1-1. Lean team member's characteristics.

1.5.Lean documentation

In LSD the team member tries to minimize all the documentation (related to lightweight method as well as reduce waste consequently during the venture life cycle to saving time as well as reducing waste (Yang et al., 2020). Excessive documentation does not add value, however, only consumes resources and time (related to eliminating Waste) (Yang et al., 2020). Investigations of documentation created amid ordinary programming advancement demonstrated that for a normal 1,000-work point programming venture (around 125,000 lines of code) is as following (Yang et al., 2020):

- Requirements regards the documentations around three-hundred pages.
- Plans documentations average one-hundred pages.
- Design Archives documentation average more than 1,500 pages.
- Utilizer manuals: documentation average which are more than six-hundred pages.
- Test reports: average more than five-thousands pages.

Few of clients read six-hundred-page manuals as well as AMs advocate lessening the above midpoints.

1.6.LSD process

- Start-up phase is the initial part of the venture, which leads to reduce risk as well as the capability (Kalaria et al., 2020).
- Steady-state phase iterative part in LSD to build a little process carried out via a decision as well as iterative development.
- Transition as well as renewal phase is the last part of the venture life cycle (Yang et al., 2020).
- Self-learning, as well as self-adopting organizations can utilize LPs according to their CMMI Capability Maturity Model Integrated (Yang et al., 2020).

1.7. The research methodology

The used methodology in this paper can be summarized in the steps that presented in Figure.2.below.

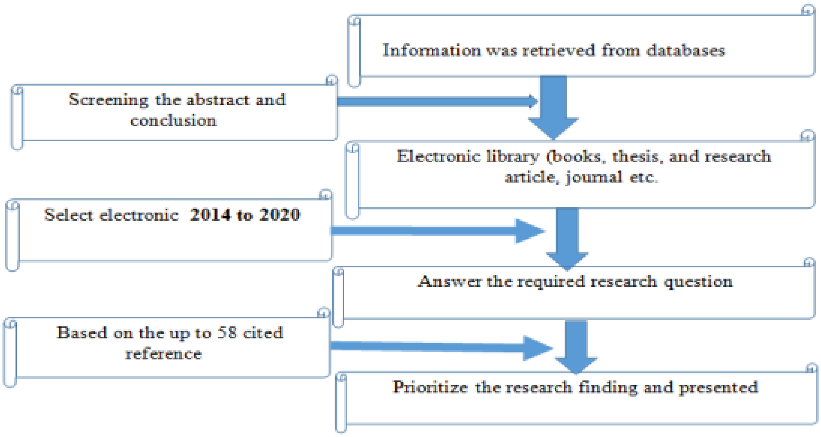
Information was retrieved from databases
Screening the abstract and conclusion
Electronic library (books, thesis, and research article, journal etc.
Select electronic 2014 to 2020
Answer the required research question
Based on the up to 58 cited reference
Prioritize the research finding and presented

Figure.2. The research methodology.

2. Literature review

Lean Software development is an emerging paradigm in software development. Also its significance is growing among practitioners belongs to researchers due to the popularity of Agile Methods as well as DevOPs in software development (Yang et al., 2020). Therefore, its evolution trend remain an interesting area. To observe Lean Software Development evolution most of the major digital libraries for instance, Scopus, Web of Science, IEEE, etc. are reconnoitered to find articles related to LSD during last one as well as half decade in which it has been evolved (Yang et al., 2020). Here main objective is to focus on LSD evolution, its analysis, pattern to assist future researchers which is linked with software professionals. This work is organized in form of tables to provide summary on different attributes at a glance. There are very few related studies for instance (Kupiainen et al., 2015) studied application of metrics in Agile as well as Lean Software Development in industrial studies. We could not find such study belongs to therefore it motivates authors to investigate further in this area. To the best of our knowledge this is first such study in area of LSD. Here main objective is to study on LSD evolution, its analysis, advantages belongs to the limitations, Lean practices, project types etc., scenario to assist future researchers as well as software professionals. Also we have noted important gist of different crucial papers in this area. As Agile Methods belongs its associated methodologies like LSD also DevOps will play a major role in coming future thus, this study advances body of knowledge in this direction. This study focuses on observations regarding the evolution of LSD as well as lean core principles, practices within the last twelve years (till, 2019). The first contribution is defined as LSD as well as LPs besides when these can be influential along with limitations of LSD as reviewed via some well-known authors in this field. The second contribution studies the relationship between ASD as well as LSD as well as also the main variances between LSD as well as ASD besides how to be combined to any work together as lean thinking. Furthermore, LSD principles have been put into utilizing, as identified via (Yang et al., 2020). In 2020, Yang et al., introduced seven principles to help software development communities (Janes, 2015). According to Janes, (2015) transformation in LSD is comparison to the approach designed via Ericsson, termed as “lean Amplifier”. He had introduced 12 lean core principles to be applied in the evaluation. Lean appears in general manufacturing, from software development to production practice (Janes, 2015). In 2016, some large manufacturing organizations has started to utilize the KLSS model to enhance product quality as well as increase efficiency to ensure customer requirements as well as satisfaction (Arun Kumar and Dillibabu, 2016). Similarly, Saboo et al. (2014) has noticed that some of the manufacturing firms as well as organizations in India approved actually utilizing lean software development.

3. The Evolution of LSD

3.1. Lean approach as frameworks advantages as well as disadvantages

Lean approach as described in Table A-1 which demonstrates Lean according to each author as well as which kind of frameworks have been mentioned via the author besides LPs the author described along with the advantages of utilizing such kind of LM, also disadvantages are presented in this section.

3.2. Lean practices

This section is regarding the Lean practices relevant to software development and LSD practices in published papers such as conferences, books, thesis, journals etc. during 2014-2020. The Lean practices in Table.A-2 has been presented according to each author and years.

3.3. Lean practices as well as venture domains

This section detailed lean approach as well as the Lean practices relevant to software development belongs to LSD practices in published papers (conferences, books, thesis, journals etc.) during 2014-2020. In addition, Table.A.3 has presented LSD practices according to each author, as well as when can be utilized according to the evidence of each author in any venture domain or business domain.

3.4. Lean software practices as well as SWDLC

This section describes Lean practices relevant to software development as well as LSD practices in published papers during the last decade besides where these can be applied during SDLC. These practices are illustrated in Table.A-4 according to applications in SDLC applications, each author, as well as when can be applied according to the evidence of each author in SDLC phases.

4. The obserational evolution scenario of lean software development

Lean came into several software development practices for the first time in Oct (1992) in a conference organized in Stuttgart Germany by ESPRIT initiative of the European Union. The origin was from the book “The Machine That Changed the World: The Story of Lean Production” authored via James Womack, Daniel Jones, in addition Daniel Roos illustrated the approach in management at Toyota. The idea to apply lean methods in software development was later developed after it was observed to be effective in manufacturing and industrial engineering. Initially in 1995, Womack as well as Jones had defined five key pillars of lean thinking i.e. value, value stream, flow, pull as well as perfection (attained via elimination of waste). The next decade observed the term associated more with the manufacturing industry rather than software development; this later changed in the 21st century when Poppendieck and Poppendieck revamped the idea. The 5 pillars were reviewed to be seven principles of LSD besides twenty-two (22) Tools.

According to ArunKumar and Dillibabu (2016), the kind of LM described in LSD was KLSS model this was through the LSS Kano approach. Furthermore, this model is aimed at developing and enhancing software quality without increased costs, effort and time. The KLSS model identifies the exact requirements that the customer requires in the software besides the utility of the software so as to ensure they are strictly addressed. The model is also utilized to categorize the requirements to identify the nature of the defect, eliminate the requirements of no value processes also to ensure that the main functionality is implemented so that the expectations of the customers are met. ArunKumar and Dillibabu (2016) indicate that the model is tested and has implemented in a leading IT firm. When the model is applied, the advantage is that the outcomes are greater improved software in terms of quality, effort, and costs The KLSS operations have the disadvantage of weak operations; additionally, most firms utilize KLSS in their ventures.

Jadhav et al. (2014) has reported that any organization’s survival depends upon its competitive edge; the top management is highly tasked to identify, recognize which is linked with the implementation of lean practices such as Kanban, quality circle, as well as just-in-time purchasing.

ISM is an interactive learning process that ensures human beings are assisted to understand better, what they believe as well as recognize undoubtedly, what they know (Attri et al., 2013). The characteristics of the ISM process are that the methodology is interpretive as the judgment group makes the decisions on whether also how the diverse elements are connected. The model is also structural based on mutual relationship.

It also helps impose direction and order where there are complex relationships in the elements of the framework. The model is principally developed as a group learning process; however, individuals can also apply it. It has six steps, for instance, Structural Self-Interaction Matrix (SSIM); Step 2: Reachability Matrix; Step 3: Level partitions; Step 4: Conical matrix; Step 5: Digraph; as well as Step 6: ISM Model (Attri et al., 2013). Digalwar et al., (2020) has identified the need to apply lean thinking in the health sector; however, this is restricted via the limited knowledge on the success as well as interventions needed to succeed. The study reviews the

Swedish pediatric accident also emergency department to able to understand the lean process. The optimal solution identified is the redevelopment of lean LPs that are in line with each section. In adapting LPs, there is a possibility of developing knowledge in the organization. The success of any method in an organization brings certainty in the requirements as well as a better way is known to adopt the framework. The only disadvantage is that implementing the process can be compared to journeying a mysterious route (Digalwar et al., 2020). In addition, the advantage of the model is the increased product quality as well as the increase of the level of customer satisfaction besides timely delivery also work within the budget.

LSD is a set of tools and principles resulting from Lean manufacturing that focuses on removing waste, delaying the decisions, enhancing the quality of the product at the earliest time. Nurdiani et al. (2016) state that “the growth of interest in Agile as well as LSD is reflected via a large number of research papers published between 2001 besides 2010”. The study in 2020 by via Wińska and Dąbrowski is on the applicability of the Agile practices in Global software development. The number of studies in Frameworkatic Literature Review (SLR) has continued to rise since 2008 reaching a record high in 2014 as well as later plummeting in (2015). The studies with Literature Review (LR) have mainly been featured in 2014, while those in Frameworkatic Mapping (SM) are seen in 2010 which are rise dramatically in (2013). Furthermore, the advantages of utilizing the combination of Agile as well as LM is that there is improved productivity, it will enable learning; it ensures the product is of quality, has a better walkthrough besides it enhances the software development process. The disadvantage is that the state cannot be generalized the external validity due to the limitations in research. Jonsson et al. (2013) propose a framework that is structured according to the authors and they are for utilizing on lean principles.

Wińska and Dąbrowski, (2020) reviewed the management principles in Toyota to build software; he identified eight major seminal sources for LSD. Furthermore, the software industry is a vibrant business that must always look for new ways to develop competitive products; this is for the reason that new paradigms succeed each other fast. The changing patterns in lean management have been rigorous over decades, for instance, sometimes it is dominated via object-oriented programming then it is overtaken via 4G languages. Additionally, in terms of method is that there is a sequential “waterfall’ methods then they are replaced via iterative methods such as spiral also unified process). The AMs is the current hot topic and no one knows what will be the next. The need to eliminate the crisis in the software industry and put in the past the reputation of bad quality and costly delays in ventures there was the need for a new evolution. To attain this success there have been many initiatives such as the Capability Maturity Model that was created to assist organizations to have the ability to deliver quality software efficiently.

The evolution in the LSD was via the need to scale up production from small craftsmanship; this however needed more disciplined. The plan-driven methods have dominated large-scale production of software for a long time. Researchers from Massachusetts Institute of Techy (MIT) of the phenomenon that Toyota had much more efficient production as well as with high quality than other American firms brought forth the Lean production philosophy. The major challenge to apply this philosophy was that software had unique features as opposed to physical products; they include intangible design, a complex logic, high design costs as well as low production cost. Jonsson et al. (2012) identified a number of seminal sources for lean in addition the basic concepts. Sodhi et al., (2020) discussed the roots of lean models and dates it back to 1950s; this is based on the variance in the Japanese and Western cultures of management. Edgeman, (2019) in his study identified more than one-hundred hits when looking for a database; of these 30 peers reviewed journals and magazines were left. The 30 primary papers were then checked on the references they had and hence several seminal sources were identified. They were divided into those that referred to lean production in general and those that were specific on LSD.

Andersson's books were the most cited in terms of Kanban venture management method as well as the agile method Scrum (Wińska and Dąbrowski, 2020). Middleton and Sutton had fewer citations , however, had more scientific evidence besides more concrete guidance on the application of lean concepts in software industry. Coplein and Bjornvig also provided solid guidance to software though the focus was on lean architecture (Goienetxea Uriarte and Urenda Moris, 2020). The 1998 Morgan's thesis found to be the oldest reference seminal paper in addition the primary paper by Ayayoma was identified and showed that LPs in software has older roots from the Japanese industry (Goienetxea Uriarte and Urenda Moris, 2020). In terms of basic lean concepts, Aitken, (2014); Antinyan, (2014); ArunKumar and Dillibabu, (2016); Cawley et al., (2015) has identifies fourteen principles that are most utilized in the management sector; they are Chen and Power, (2015); Colazo, (2016); Edison et al., (2015); Fagerholm et al., (2015); Llahm et al., (2017). Standardized responsibilities as well as processes are the groundwork for incessant development also empowerment of employees (Goienetxea Uriarte and Urenda Moris, 2020). Utilize visual control to ensure that there are no hidden problems. Utilize only dependable, frameworkatically tested technology that provides people and processes. Develop leaders who are aware of their duties, make sure they implement the philosophy and spread it to others.

Womark and Jones (2015) provide five important concepts; Value, Value Stream, Flow, Pull as well as Perfection. The concepts can be identified as a summary of lean thinking. Indeed, Womack's definitions are more concise and technical, for instance, these are process oriented, however, likers are the same principles with more details focusing on the human side. In the line of LSD, Poppendiecks' principles as given earlier are in line with the principles given via Liker though they are not concretely articulated. The principles remain the same, only that Liker emphasizes the importance of standards of working methods while Poppendieck's focuses more on self-determination. The major issue with Poppendieck's principle is that there is no description of how the work should be documented as well as when the issues should be discussed.

Andersson defines Kanban software principles as Jonsson et al. (2013)

- Visualize the workflow (L7)
- Limit work in progress (L3, L4, P5)
- Manage flow (L2, L4)
- Make process policies explicit (L6)
- Increase collaboratively (via utilizing models as well as the scientific method) (L14, P7).

Andersson has gone a step ahead in establishing the Lean Software as well as framework Consortium (www.leanframeworksociety.org) (Bamana et al., 2019); (Salleh and Nohuddin, 2019); (Schonberger, 2019) that is charged with the promotion of lean in many areas including software. The preliminary principles that guide the consortium include Jonsson et al. (2012)

Follow at thinking as well as blueprint Approach frameworks. In addition, sprouting outcomes can be informed via reviewing the framework of a framework that is multifaceted and adaptive, respect people.

- Apply methods that are scientific to drive improvements.
- Promote Leadership.
- Enhance Visibility (into work, workflow, and framework operation).
- Decrease Flow Time.
- Trim down on waste so as to enhance efficiency.

Chen (2015) advocates for the continuous delivery optimal solution. Continuous delivery is a software engineering principle where groups keep producing valuable software in short cycles as well as make certain that there will be a reliable release of software at any time. In addition, the continuous delivery six steps. The benefits of continuous delivery include an accelerated time to market; improved customer satisfaction; improved quality; reliable releases; enhanced productivity as well as efficiency; and developing the right product.

5. Discussions

Over the span of the latest decade, there have been various changes in LSD; the lean practices continue evolving to suit new trends as well as most of all to keep firms competitive. Object-oriented programming was replaced via 4G languages; in terms of methodology, the sequential "waterfall methods were replaced via iterative methods such as spiral and unified process. Currently, the agile methods are more practiced waiting for the next generation of invention (Wińska and Dąbrowski, 2020). In the course of the last ten years, there have been various lean practices published in papers (In conferences and journals) i.e. see Table A-3. "Lean practices relevant to software development". These include continuous improvement; this has been discussed via authors such as ArunKumar and Dillibabu (2016); Jadhav et al. (2014); Rodríguez et al. (2014) and Jonsson et al. (2013). ArunKumar and Dillibabu (2016) also reviewed the Kano analysis lean practice that focuses on the requirements of the customers. There is also the lean practice that makes all items transparent; i.e. ensuring the venture is highly visible, as well as visualizing all the elements needed in the work as presented in Table A-3. ArunKumar and Dillibabu (2016) discussed this practice. The lean practice of measuring and managing is also reviewed; this includes the employ queuing theory and exact items measurements. Authors, for instance, ArunKumar and Dillibabu (2016) reviewed this aspect in Table.A-3; Table.A-1.

The Plane-do-check-act (PCDA) cycle is also reviewed by Jadhav et al. (2014). Other lean practices include Poka-yoke: defect detection as well as prevention; Quality function deployment;Reduce slack; Root cause analysis; Batch control processing; Avoid too much local optimization; Defer decision making; Developing appropriate incentives/rewards; Hide individual performance; and Kaikaku: radical improvement within a limited time as supported by Pernstål et al. (2013).

The Jidoka: intelligent automation while Jonsson et al. (2013) discussed the Heijunka: eliminate Muda. Laanti, (2016) discusses the two-level planning lean practice. Laanti, (2016) discusses VSM; Rodríguez et al. (2014); Jonsson et al. (2013); Chen, (2015) as presented in figure 4.2; Jadhav et al. (2014) review the JIT Lean processes. The need for lean practices as demonstrated by the researchers is paramount; software developers must ensure they implement the most appropriate method that makes their products as well as services competitive. Undeniably, the evolution of LSD has been majorly focused on making the process effective, efficient and less costly.

In evidence, change has been the only constant aspect advocated for in the lean development process; the change is from a vaguer process of production of a lean objective and customer-focused model (Colazo, 2016); (Chaple et al., 2018). As Ebert et al. (2012) in Table.A-3 as well as Figure.4.2. has stated that the process is a paradigm shift of product development with a close focus on customer satisfaction, optimizing value, eliminate waste, continuous improvements as well as empowering people. Authors continue to base their arguments on the initial model , however, try to modify it to suit the diverse sectors of the economy. Indeed, over the decade, it has seen more as well as more application of the concept by authors each based on the diverse principle or concept that shape their view as well as function. Jadhav et al. (2014) recommend the interpretive structural modeling (ISM) Figure.1-4, the authors has presented that it as an update to the JIT version as well as share a similar approach in a change of the organization. The model focuses on value addition and eliminating waste; this is in a bid to keep the organization competitive than others. The researchers noted that the implementation of the lean strategies must be done accordingly otherwise they will lead to failure; this is for the reason that the competition in the market demands very cost-efficient products and qualitative on-time delivery of products at the right place.

The lean practice bundles utilized by the authors include Waste elimination practice, volume flexibility practice, delivery reliability practice, low cost practice, health as well as safety practice, human resource management practice, conformance quality practice and creativity and innovation practice as illustrated in Table.A-3. belongs to Lean principles and practices relevant to ventures domain (Włodarski and Poniszewska-Marańda, 2020); (Correia et al., 2018). In the study of the lean development process over the decade, one aspect certainly stands out; this is the need to address current changes while still utilizing the best as well as most efficient process in delivering quality to customers. The aspect is still common to the need at the inception of the Toyota production framework (TPS). Chen (2015), Rodríguez et al. (2014) discussed that utilizing LPs can Increase productivity in Table A-3., customer value, product quality, while Jonsson et al. (2013) discussed that LPs helps practitioners as well as researchers on how to apply lean thinking, detect the variance amongst AMs and LM (Edgeman, 2019); (Chong and Perumal, 2019).

To discuss LSD concepts there will be a need to define the meaning and context of the applications so that the process is well understood. The aim is to define the software development process as well as its attributes also more importantly defining the evolution the sector has undergone over the last decade. In understanding, the relationships between lean that provide the framework for principles such as agile and the methodologies applied such as Kanban and scrum then the study will provide adequate knowledge of the evolution process.

6. Limitations

- Time limits: this research present LSD throughout LM based on evolution in lean software development during last decade.
- The objective limits: this research study identifies venture domains where LPs as well as lean practices can be utilized and how these principles can improve and enhance SWDLC, along with the impact of lean practices and stakeholders as cross-functional team members who derived from SWDLC by utilizing LSD to improve software development process phases.
- This paper directly addresses LPs as they are mostly followed by many organizations as well as industries of any size such as small, medium also large all over the world. On the other hand, LSD as a method which can empower the organizations via applying LPs and

practices as supported by.

- For the purpose here in, literature on LPs has been collected from diverse sources dating back a decade, and studied as to how LPs can be beneficial to enterprises. There is no empirical data (questionnaire, surveys etc.) in this research despite the fact that the author believes that there needs to be such data in order to further strengthen the outcomes.

7. Conclusions and Future Research

This study aimed to introduce LSD, as an optimal solution to emphasize software development process, also the study has defined Lean from the past to the recent also the evolution during the life cycle of this methodology. The main idea of this study is to outline challenges between Lean practices and LPs. Also to introduce many kinds of Lean methodology throughout utilizing this methodology in many diverse phases in diverse ventures. Further, this study is significant in three important contributions in business are, the first contribution remains defined as LSD and LPs and when these can be influential along with limitations of LSD as reviewed via some well-known authors in this field.

The second contribution studies the relationship between ASD as well as LSD and also the main variances between LSD, ASD as well as how to be combined to any work together as lean thinking. The benefit of utilizing both in the same (organization, firm, etc.) also also how LSD and ASD can improve software development process. The study contributes the whole understanding of all the changes of LSD principles and practices throughout the LSD journey during the last ten years. This paper provides theoretical evidences on the topic of the study via observing all the collected papers in the context of LSD and ASD. This argument defines LSD as a method, which can be helpful in the business domains. Software development principles as well as practices can improve organizations, firms' business area, also qualification scales. In addition, this study aims for future study which will move towards in-depth of LSD in a wide area of practicing LPs or practices, moreover, a comparative study of real firms that have been already successful via utilizing one of LSD (lean thinking, lean practices, lean principles…etc.) is kept into the next future study.

8. REFERENCES

Aitken, A. (2014). Dual Application Model for Agile Software Engineering. 2014 47th Hawaii International Conference on Framework Science. 4789 - 4798

Alba-Baena, N., Estrada, F. J., & Torres, O. O. S. (2020). Using lean-sigma for the integration of two products during a ramp-up event. In Sustainable Business: Concepts, Methodologies, Tools, and Applications (pp. 954-976). IGI Global.

Andersson, D.J. (2010). Kanban, Chicago: Blue Hole Press.

Antinyan, V. et al. (2014). Identifying Risky Areas of Software Code in Agile/Lean Software Development: An Industrial Experience Report. Software Maintenance, Reengineering and Reverse Engineering (CSMR-WCRE), 2014 Software Evolution Week - IEEE Conference.

ArunKumar, G. & Dillibabu, R. (2016). Design and Application of New Quality Improvement Model: Kano Lean Six Sigma for Software Maintenance Project. Arab J Sci Eng. 41:997–1014

Attri, R., Grover, S., Dev, N., & Kumar, D. (2013). An ISM approach for modelling the enablers in the implementation of Total Productive Maintenance (TPM). International Journal of Framework Assurance Engineering and Management, 4(4), 313-326.

Bamana, F., Lehoux, N., & Cloutier, C. (2019). Simulation of a Construction Project: Assessing

Impact of Just-in-Time and Lean Principles. Journal of Construction Engineering and Management, 145(5), 05019005.

Baumer-Cardoso, M. I., Campos, L. M., Santos, P. P. P., & Frazzon, E. M. (2020). Simulation based analysis of catalyzers and trade-offs in Lean & Green manufacturing. Journal of Cleaner Production, 242, 118411.

Cawley, O., Richardson, I., Wang, X., Kuhrmann, M., (2015). A Conceptual Framework for Lean Regulated Software Development. ACM. 978-1-4503-3346-7/15/08

Chaple, A. P., Narkhede, B. E., Akarte, M. M., & Raut, R. (2018). Interpretive framework for analyzing lean implementation using ISM and IRP modeling. Benchmarking: An International Journal, 25(9), 3406-3442.

Chen, L. (2015) "Continuous Delivery: Huge Benefits, but Challenges Too," IEEE Software, 32: 23 – 45

Chen, L., & Power, P. (2015). Continuous Delivery Huge Benefits, but Challenges Too. The IEEE Computer Society. 50 – 54

Chong, J. Y., & Perumal, P. A. (2019, July). Conceptual Framework for Lean Manufacturing Implementation in SMEs with PDCA Approach. In Symposium on Intelligent Manufacturing and Mechatronics (pp. 410-418). Springer, Singapore.

Colazo, J. (2016). Team Network Evolution while Undergoing a Lean Transformation: A Missing Success Indicator? 2016 49th Hawaii International Conference on Framework Sciences

Colazo, J. (2016). Team Network Evolution while Undergoing a Lean Transformation: A Missing Success Indicator? 2016 49th Hawaii International Conference on Framework Sciences

Correia, D., Silva, F. J. G., Gouveia, R. M., Pereira, T., & Ferreira, L. P. (2018). Improving manual assembly lines devoted to complex electronic devices by applying Lean tools. Procedia Manufacturing, 17, 663-671.

Digalwar, A., Raut, R. D., Yadav, V. S., Narkhede, B., Gardas, B. B., & Gotmare, A. (2020). Evaluation of critical constructs for measurement of sustainable supply chain practices in lean-agile firms of Indian origin: A hybrid ISM-ANP approach. Business Strategy and the Environment.

Ebert, C., Abrahamsson, P. & Oza, N. (2012). Lean Software Development. IEEE computer society. 22 – 25

Edgeman, R. (2019). Complex Management Frameworks and the Shingo Model: Foundations of Operational Excellence and Supporting Tools. Productivity Press.

Edison, H. Wang, X., & Abrahamsson, P. (2015).Lean Startup: Why Large Software Firms Should Care. Workshops May 25 - 29, 2015, Helsinki, Finland. ACM. ISBN 978-1-4503-3409-9/15/05

Fagerholm, F., Ikonen, M., Kettunen, P., Münch, J., Roto, V. & Abrahamsson, P. (2015). Performance Alignment Work: How software developers experience the continuous adaptation of team performance in Lean and Agile environments. Information and Software Technology. 64: 132 – 147

Goienetxea Uriarte, A., Ng, A. H., & Urenda Moris, M. (2020). Bringing together Lean and simulation: a comprehensive review. International Journal of Production Research, 58(1), 87-117.

Goienetxea Uriarte, A., Ng, A. H., & Urenda Moris, M. (2020). Bringing together Lean and simulation: a comprehensive review. International Journal of Production Research, 58(1), 87-117.

http://www.shmula.com/lean-for-software-interview-with-mary-poppendieck/340/

Jadhav, J. R., Mantha, S. S. & Rane, S. B. (2014). Development of framework for sustainable Lean implementation: an ISM approach. J Ind Eng Int. 10:72

Janes, A. (2015). A Guide to Lean Software Development in Action. IEEE Eighth International Conference on Software Testing, Verification and Validation Workshops (ICSTW). IEEE, 13th utilizer Symposium on Software Quality, Test and Innovation (ASQT 2015)

Jonsson, H. Larsson, S. & Punnekkat, S. (2013). Synthesizing a Comprehensive Framework for Lean Software Development. 2013 39th Euromicro Conference Series on Software Engineering and Advanced Applications

Kalaria, D. R., Parker, K., Reynolds, G. K., & Laru, J. (2020). An industrial approach towards solid dosage development for first-in-human studies: application of predictive science and lean principles. Drug Discovery Today.

Kupiainen, E., Mäntylä, M. V. & Itkonen, J. (2015). Using metrics in Agile and Lean Software Development – A frameworkatic literature review of industrial studies. Information and Software Technology. 62: 143–163

Laanti, M. (2016). Piloting Lean-Agile Hardware Development. XP '16 Workshops. May 24-24, 2016, Edinburgh, Scotland UK. ACM. 978-1-4503-4134-9/16/05

Llahm F, Mishra, A., Yazici, A. (2017). Observations on Evolution of Lean Software Development Thesis, Atilim University

Mishra, A., Garbajosa, J., Wang, X., Bosch, J., & Abrahamsson, P. (2017). Future directions in Agile research: Alignment and divergence between research and practice. Journal of Software: Evolution and Process, 29(6), e1884.

Nurdiani, I., Börstler, J., & Fricker, S.A. (2016). The impacts of agile and lean practices on project constraints: A tertiary study. The Journal of Frameworks and Software. 119: 162 – 183

Ramdass, K. (2015). Integrating 5S Principles with Process Improvement: A Case Study. 2015 Proceedings of PICMET '15: Management of the Technology Age. 1908 – 1917
Rauch, E., Dallasega, P. & Matt, D.T. (2016). The way from Lean Product Development (LPD) to Smart Product Development (SPD): 26th CIRP Design Conference. Procedia CIRP. 50: 26 – 31

Razzak,M. A. (2016). An Empirical Study on Lean and Agile Methods in Global Software Development. 2016 IEEE, 11th International Conference on Global Software Engineering Workshops

Rich, N., & Shararah, M. A. (2020). Frameworks for Manufacturing Excellence: Generating efficient and reliable manufacturing operations. Kogan Page Publishers.

Rodríguez, P., Partanen, J., Kuvaja, P., & Oivo, M. (2014). Combining Lean Thinking and Agile Methods for Software Development: A Case Study of a Finnish Provider of Wireless Embedded Frameworks. IEEE, 2014 47th Hawaii International Conference on Framework Science.

Saboo, A., Garza-Reyes, J.A., Er, A. and Kumar, V. (2014) 'A VSM improvement-based approach for lean operations in an Indian manufacturing SME', Int. J. Lean Enterprise Research. 1(1):41–58

Sagnak, M., & Kazancoglu, Y. (2020). Integration of green lean approach with six sigma: an application for flue gas emissions. Journal of Cleaner Production. xxx: 1 – 7

Salleh, N. M., & Nohuddin, P. N. (2019). Optimization of software requirement process: An integrated conceptual model of lean six sigma and requirement planning. International Review of Applied Sciences and Engineering, 10(2), 125-133.

Saltz, J., & Sutherland, A. (2020, January). SKI: A New Agile Framework that Supports DevOps, Continuous Delivery, and Lean Hypothesis Testing. In Proceedings of the 53rd Hawaii International Conference on Framework Sciences.

Schonberger, R. J. (2019). The disintegration of lean manufacturing and lean management. Business Horizons, 62(3), 359-371.

Secor, D., Lucae, S. & Rebentisch, E. (2014). Planning for Resilient Lean Programs. Conference on Frameworks Engineering Research (CSER 2014). Procedia Computer Science. 28 (2014) 138 – 145

Silva, F. S. et al. (2015). Using CMMI together with agile software development: A frameworkatic review. Information and Software Technology. 58: 20 – 43

Sodhi, H. S., Singh, D., & Singh, B. J. (2020). A conceptual examination of Lean, Six Sigma and Lean Six Sigma models for managing waste in manufacturing SMEs. World Journal of Science, Technology and Sustainable Development.

Sutherland, A., Saltz, J., & Anderson, E. (2020, January). Introduction to the Minitrack on Agile and Lean: Organizations, Products and Development. In Proceedings of the 53rd Hawaii International Conference on Framework Sciences.

Tenera, A., & Pinto, L. C. (2014). A Lean Six Sigma (LSS) project management improvement model. Procedia-Social and Behavioral Sciences, 119, 912-920.

Wang, T. K., Yang, T., Yang, C. Y., & Chan, F. T. (2015). Lean principles and simulation optimization for emergency department layout design. Industrial Management & Data Frameworks, 115(4), 678-699.

Wińska, E., & Dąbrowski, W. (2020). Software Development Artifacts in Large Agile Organizations: A Comparison of Scaling Agile Methods. In Data-Centric Business and Applications (pp. 101-116). Springer, Cham.

Włodarski, R., & Poniszewska-Marańda, A. (2020). Development Method of Innovative Projects in Higher Education Based on Traditional Software Building Process. In Data-Centric Business and Applications (pp. 225-243). Springer, Cham.

Womack, J. P., & Jones, D. T. (2015). Lean solutions: how firms and customers can create value and wealth together. Simon and Schuster.

Wu, Y., Sato, K., Jing, L., Wang, J., & Cheng, Z. (2015). The Lean Awareness in Software intensive Engineering: Experience from One Project. 2015 IEEE 7th International Conference on Awareness Science and Technology (iCAST 2015)

Yadav, G., Luthra, S., Huisingh, D., Mangla, S. K., Narkhede, B. E., & Liu, Y. (2020). Development of a lean manufacturing framework to enhance its adoption within manufacturing firms in developing economies. Journal of Cleaner Production, 245, 118726.

Yang, T., Wen, Y. F., Hsieh, Z. R., & Zhang, J. (2020). A lean production framework design for semiconductor crystal-ingot pulling manufacturing using hybrid Taguchi method and simulation optimization. Assembly Automation.

Zhu, K. & Lv, J. (2014). Research on the Improvement Effect of Lean Thinking to Construction Safety Management. Control and Decision Conference (2014 CCDC), The 26th Chinese. IEEE. pp. 1428 – 1433

9. APPENDIX- A

Table A-1. LM advantages and limitations as frameworks

Authors name	Which kind of LM the author describes in SD	Optimal solutions	The advantages	Disadvantages
(Kumar and Dillibabu, 2016) (Edgeman, 2019) (Salleh and Nohuddin, 2019) (Yadav et al., 2020) (Llahm et al., 2017)	Propose a model called (KLSS model) by combining LSS with Kano model	(KLSS) to determine as well as prioritize the PREQ from the customer domain. which will help SD to remove EW as well as implement the main functions	Increase the product quality within less cost, EW,	1.LSS operations rather weak 2.The firm most already 3.Utilize LSS in their projects.
(Jadhav, 2014) (Llahm et al., 2017) (Patri and Suresh, 2018) (Chaple et al., 2018)	Identify ISM model which contain 4 group of lean practices	(ISM) FW	Help leaders as well as managers besides practitioners to reduce cost	1.Non-expertise LEAM 2.Misunderstanding by the stakeholder 3.the relationships amongst lean practices
(Huan and ZhanWen, 2018) (Psomas et al., 2018) (Llahm et al., 2017)	Apply lean thinking as well as try to adopt LPs	Redevelopment LPs linked with the requirements of each section	Adapt LPs is possible that the development of knowledge in organizations and the work of other sites as well as become the method of work that organization in accordance with the requirements as well as reach for success certainly. solve Lean problems	Trying to cut down the mysterious processes
(Huan and ZhanWen, 2018) (Yadav et al., 2020) (Llahm et al., 2017)	Utilize Poppendieck's LPs	Utilize Poppendieck's 7 LPs EW, BQ, CKG ,DC,DF,OTW,RESP	Increase product quality also customer satisfaction deliver in time within budget	

(Rodríguez, 2014) (Correia et al., 2018) (Llahm et al., 2017)	Combine LM as well as AM , Utilize five core LPs of lean thinking ,LPs	Utilize the combination LM as well as AM	IMPRP ,Learning ,Ensure the product quality as well as better understand of walkthrough, Enhance SD process	This state cannot be generalized (external validity) for the reason that of the research limitation
(Jonsson, 2013) (Llahm et al., 2017)	propose a FW structured according to the authors as well as their focusing on LPs	Utilize LPs according to each author	Helps practitioners and researchers on how to apply lean thinking, detect the difference between AM and LM	
(Chen, 2015) (Llahm et al., 2017)	Utilize CDY	CDY	Increase productivity, customer satisfactions, product quality,	sudden change in the organizational structure
(Laanti, 2016) (Correia et al., 2018)	Utilize lean thinking as well as agile	FW can be utilized in phase-gate model as well as two-level planning besides multiple solution also DC as well as self-management	Reusability (the proposed FW can be utilized in another projects), reduce iteration errors by monitoring errors in each iteration so this errors can be inputs in next iteration, increase time to market, reach a good CMMI level	Peer feedback, people demand, hand off code or design,
(Wang, 2015) (Correia et al., 2018) (Rich and Shararah, 2020) (Llahm et al., 2017)	Lean thinking as well as regulated SD FW	Lean 7 LPs	introduce lean thinking in everyday life requirements, as well as the formation of independent developments teams, organize SWDPS as well as secure it	
(Abrahamsson, 2015)	Lean startup	Lean startup 5 LPs as an analytical FW	doorway to tangible as well as immaterial assets, increase time to market, improve innovativeness	1.too much Cost, 2.brand and reputation can be destroyed, 3.Culture change, managers cannot run it for the reason that it is new
(Secor, 2014) (Llahm et al., 2017)	LEAP method, workshop	Increase customer satisfaction, reduce risk (waste)	Help to include the key program to help stakeholders in up-front planning, increase team performance, reduce cost, increase ORGs income, improve leadership skills, increase team	

			satisfaction	
(Correia et al., 2018) (Rich and Shararah, 2020)	Platform of adapt LPs	13 LPs	Utilize LPs to achieve LPD as well as enterprise level	
(Rich and Shararah, 2020)	A3 problem solving during PDCA lifecycle.	EW,DF,OTW, RESP	Long fix-times ,which will not paid via the customer	many feedbacks can find optimal solution, the organization can implement CMMI disciplines within lean culture.
(Nord, 2012) (Llahm et al., 2017)	LSD flow management FW, architecture-related with lean thinking	EW,DC,DF,RP	1.Impove flow via managing overproduction waste with over architect Turing, eliminate rework cost, acceptance test units can quid the project(work), scheduling in Kanban can JIT delivery via emphasizing pulling. via make compatibility between LPs as well as lean awareness 4 principles. lean awareness 4 LPs are (EW=Cost effectiveness, AL =Accumulative effort	If the developers could not determine the size increment of an architecture for each iteration they will not manage the development flow. , Monitoring is a helpful way to detect errors as well as increase productivity, better outcomes via keeping away from conditions that wasted time and exertion and cost, avoid rework cost and delay cost.
(Wang, 2015) (Llahm et al., 2017)	Lean awareness	(FW)	Decide as late as possible=Accumulative effort, DF=Accumulative effort ETT= Feasibility, BQ= Manageability, OTW =Accumulative effort improve the manageability, cost effectiveness as well as Accumulative effort with the work cells, empower	New developers join the team when needs come. crowded data,

			the innovation via improving the manageability	

Table A-2.Lean practices relevant to SD and projects types.

Authors and year	Lean principles	Lean practices	Projects domain
(Laanti, 2016) (Llahm et al., 2017) (Gutbrod and Münch, 2018) (Yadav et al., 2020)	DC,ETT,OTW Self-management	Two-level planning	pilot Electrical engineering, hardware development, SWDPS ,software in intensive healthcare
(Wang, 2015) (Llahm et al., 2017) (Yadav et al., 2020) (Patri and Suresh, 2018) (Psomas et al., 2018)	7 LPs	Rapid prototyping, object oriented as well as component based development, quality function deployment, continuous integration	Automotive SW, Robotics, medical Devices, financial Management frameworks
(Abrahamsson, 2015) (Llahm et al., 2017) (Gutbrod and Münch, 2018)	LPs 5 startup	Continuous innovation	Large software firms, internal projects
(Secor, 2014) (Llahm et al., 2017) (Patri and Suresh, 2018) (Yadav et al., 2020)	increase customer satisfaction Reduce waste as well as risk	VSM	large international firm head quartered in the US. government programs. Utilized in (Rock Well Collins).
(Llahm et al., 2017) (Huan and ZhanWen, 2018) (Yadav et al., 2020)	Lean 13 LPs to improve LPD as well as TPDS as well as PD enterprise level	VSM pull flow	large-scale development projects automotive industry. Volvo corporation (VCC)Volvo truck corporation (VTC)
(Llahm et al., 2017) (Yadav et al., 2020)	A3 problem solving during PDCA lifecycle.	PDCA, Continuous integration	independent software as well as frameworks firm (IT)
(Wang, 2015) (Llahm et al., 2017) (Correia et al., 2018)	LSD flow management FW, architecture-related with lean thinking	WIP, kanban	building internal functions.a heating as well as ventilation.air conditioning access as well as safety hardware functions for high level of severity.
(Llahm et al., 2017) (Wang, 2015) (Correia et al., 2018)	Manageability Cost effectiveness Feasibility	Lean awareness	successfully developing a ESR monitoring framework developing a novel mobile cloud framework for earth surface radiation (ESR) monitoring

(Yadav et al., 2020)	Accumulative effort		
(Kumar and Dillibabu, 2016) (Llahm et al., 2017)	integrate LPs with LSS to make alogical approach to do continuous improvement	Continuous improvement, Kano analysis focuses in the customer requirements	producing high-quality software in the software industry software industry for process improvement. for embedded software as well as application projects in software development.
(Jadhav, 2014) (Llahm et al., 2017) (Bamana et al., 2019) (Schonberger, 2019)	After combining LPs with lean practices as	Lean, JIT practices	Help leaders as well as managers as well as practitioners to reduce cost building project management ICT project management Roadmap for sustainable lean implementation.
(Llahm et al., 2017) (Gutbrod and Münch, 2018)	Redevelopment LPs according to the requirements of each section	Root cause analysis	Healthcare, emergency care
(Llahm et al., 2017) (Correia et al., 2018) (Bamana et al., 2019) (Schonberger, 2019)	Utilize 7 LPs	JIT, flow commitment , Stop-the Line quality, continuous integration development	Several venture sizes
(Llahm et al., 2017) (Rodríguez, 2014) (Correia et al., 2018)	(5) LPs ,LPs	Continuous improvement, pull	large projects, wireless embedded framework
(Chen, 2015) (Llahm et al., 2017) (Gutbrod and Münch, 2018)	CDY utilization	eliminate Muda (waste types), hide individual performance, making improvement, avoid too much local optimization, WIP	Large applications, monolithic applications
(Llahm et al., 2017) (Correia et al., 2018)	Apply LPs	Comparing Value, Value stream, Flow, Pull as well as Perfection.	as framework to guide researchers as well as practitioners, in many project sizes

Table A-3. Lean practices and when can be used during SWDLC .

Lean practices	Requirements	Analysis	Design	Implementation	Testing	Authors and year
Continuous improvement	✓	✓	✓	✓	✓	• (Jadhav, 2014) • (Kumar and Dillibabu, 2016) • (Rodríguez, 2014) • (Llahm et al., 2017)
Kano analysis focuses in the customer requirements	✓	✓	✓			• (Llahm et al., 2017) • (Kumar and Dillibabu, 2016) • (Edgeman, 2019) • (Salleh and Nohuddin, 2019)
Make all items transparent Make project status highly visible Visualize all work elements	✓	✓	✓	✓	✓	• (Llahm et al., 2017) • (Kumar and Dillibabu, 2016) • (Gutbrod and Münch, 2018)
Measure and manage Employ queuing theory Measure the exact things	✓	✓				• (Llahm et al., 2017) • (Kumar and Dillibabu, 2016) • (Psomas et al., 2018)
Plane-do-check-act(PDCA) cycle						• (Llahm et al., 2017) • (Jadhav and Rane, 2014) • (Chong and Perumal, 2019)
Poka-yoke: defect detection as well as prevention						• (Llahm et al., 2017)
Move variability downstream						• (Correia et al., 2018)
Pull	✓	✓	✓	✓	✓	• (Rodríguez, 2014) • (Llahm et al., 2017)
Quality function deployment	✓					• (Llahm et al., 2017)
Reduce slack						• (Correia et al., 2018) • (Llahm et al., 2017)

	✓	✓	✓	✓	✓	
Root cause analysis		✓				• (Kumar and Dillibabu, 2016) • (Jadhav and Mantha, 2014) • (Llahm et al., 2017) • (Yadav et al., 2020)
Batch control processing		✓	✓	✓	✓	• (Rodríguez, 2014) • (Kumar and Dillibabu, 2016) • (Llahm et al., 2017)
Address bottlenecks Cumulative flow diagram(CFD)						• (Kumar and Dillibabu, 2016) • (Jadhav and Mantha, 2014) • (Llahm et al., 2017)
Avoid too much local optimization				✓	✓	• (Rodríguez, 2014) • (Kumar and Dillibabu, 2016) • (Chen, 2015) • (Llahm et al., 2017)
Defer decision making	✓	✓	✓			• (Rane, 2014) • (Llahm et al., 2017) • (Chong and Perumal, 2019) • (Yadav et al., 2020)
Develop appropriate incentives/rewards	✓	✓	✓	✓	✓	• (Llahm et al., 2017) • (Correia et al., 2018) • (Yadav et al., 2020)
Hansei: self-reflection, add commit to making improvement, relentless, acknowledge one's, own mistakes			✓	✓	✓	• (Llahm et al., 2017) • (Chen, 2015) • (Yadav et al., 2020)
Hide individual performance		✓	✓	✓	✓	• (Chen, 2015) • (Llahm et al., 2017) • (Correia et al., 2018)

Kaikaku: radical improvement within a limited time		✓	✓	✓		• (Jadhav, 2014) • (Rodríguez, 2014) • (Llahm et al., 2017) • (Yadav et al., 2020)
Jidoka: intelligent automation				✓	✓	• (Llahm et al., 2017) • (Yadav et al., 2020)
Heijunka: eliminate Muda	✓	✓	✓	✓	✓	• (Chen, 2015) • (Rodríguez, 2014) • (Llahm et al., 2017)
Kanban	✓	✓	✓	✓	✓	• (Rodríguez, 2014) • (Chen, 2015) • (Llahm et al., 2017) • (Yadav et al., 2020)
WIP			✓	✓	✓	• (Rodríguez, 2014) • (Chen, 2015) • (Llahm et al., 2017)
Two-level planning		✓	✓			• (Laanti, 2016) • (Llahm et al., 2017)
Continuous innovation		✓	✓			• (Abrahamsson, 2015) • (Llahm et al., 2017)
VSM		✓	✓	✓		• (Secor, 2014) • (Llahm et al., 2017)
Continuous integration						• (Correia et al., 2018)

			✓	✓	✓	• (Llahm et al., 2017)
JIT	✓	✓	✓	✓	✓	• (Jadhav, 2014) • (Llahm et al., 2017) • (Bamana et al., 2019) • (Schonberger, 2019)